Robust Causal Inference using Double/Debiased Machine Learning: A Guide for Empirical Research

Achim Ahrens (ETH Zurich)

with Victor Chernozhukov Christian Hansen Damian Kozbur
Mark Schaffer Thomas Wiemann

MZES Social Science Data Lab

September 18, 2024

Based on two papers:

"Model Averaging and Double Machine Learning" with Christian Hansen, Mark Schaffer, Thomas Wiemann. Forthcoming in *Journal of Applied Econometrics*. https://arxiv.org/abs/2401.01645

"Robust Causal Inference using Double/Debiased Machine Learning: A Guide for Empirical Research" with Victor Chernozhukov, Christian Hansen, Damian Kozbur, Mark Schaffer, Thomas Wiemann.

Motivation

Researchers are often interested in ***one or a few (causal) target parameters*** summarizing the relationship between key variables — e.g., the average treatment effect of a language training program or a minimum wage rise.

Motivation

Researchers are often interested in ***one or a few (causal) target parameters*** summarizing the relationship between key variables — e.g., the average treatment effect of a language training program or a minimum wage rise.

Identification of these parameters often hinges on ***high-dimensional nuisance parameters*** arising from

- ▷ Many confounding variables or many instrumental variables
- ▷ Non-linear effects

Motivation

Example 1 (Text controls)

Dube et al. (2020) ask whether employers have market power, allowing them to pay workers below their marginal productivity. Using data from MTurk, they estimate

$$\ln(\text{duration}_i) = -\eta \ln(\text{reward}_i) + g(X_i) + \varepsilon_i$$
$$\ln(\text{reward}_i) = m(X_i) + \mu_i$$

where duration = time it takes for a posted job to be filled, and reward = payment for the job.

Under some assumptions, η is a measure of market power (labor supply elasticity). However, to properly estimate η, we should adjust for job characteristics (X), which come in text format (title, description, keywords).

Motivation

Example 2 (Rainfall instruments)

Beraja et al. (2023) ask whether political unrests increase local demand for AI technology (e.g., facial recognition) in China.

$$\begin{aligned} \mathrm{AI}_{i,t+1} &= \beta \mathrm{Unrest}_{it} + \alpha_t + \gamma_i + f(X_i, Z_i) + \varepsilon_{it} \\ \mathrm{Unrest}_{it} &= \alpha'_t + \gamma'_i + h(X_i, Z_i) + \nu_{it} \end{aligned}$$

To estimate the parameter of interest, β, they adjust for local economic and political conditions (X). They leverage weather variables (rain, wind, thunder; Z) to instrument for unrests.

Motivation

Example 3 (Gender gap in wages)

A policy-relevant parameter is the ***unexplained*** gender gap in wages

$$\theta_0 \equiv E\left[E\left[Y|D=1,X\right]-E\left[Y|D=0,X\right]|D=1\right],$$

where Y denotes the logarithm of wages, D is an indicator equal to one for women, and X is a vector of potentially many individual characteristics (such as skills, education, industry affiliation, and experience) that may account for part of the ***observed*** gender wage gap.

Motivation

All these examples have two aspects in common:

1. there is a low-dimensional parameter of interest (e.g., labor supply elasticity, unexplained wage gap),
2. estimation of this parameter depends on a high-dimensional nuisance component (e.g., text controls, weather events).

Motivation

Recently, methods have been suggested that ***leverage supervised machine learning*** to aid causal effect estimation (e.g., Belloni et al., 2014). We focus on one popular method:

Double/debiased machine learning (DDML) (Chernozhukov et al., 2018)

Motivation

Recently, methods have been suggested that ***leverage supervised machine learning*** to aid causal effect estimation (e.g., Belloni et al., 2014). We focus on one popular method:

Double/debiased machine learning (DDML) (Chernozhukov et al., 2018)

The use of machine learning promises to select predictive features and capture non-linear effects in a data-driven way.

Motivation

Recently, methods have been suggested that ***leverage supervised machine learning*** to aid causal effect estimation (e.g., Belloni et al., 2014). We focus on one popular method:

Double/debiased machine learning (DDML) (Chernozhukov et al., 2018)

The use of machine learning promises to select predictive features and capture non-linear effects in a data-driven way.

Recent literature also ***raises concerns*** about practical advantages of relying on ***ML for causal inference***:

- Goller et al. (2020): Random forests + matching "might lead to misleading results."
- Wüthrich and Zhu (2023): Lasso selection of controls can introduce OVB in small samples.
- Angrist and Frandsen (2022): "ML seems ill-suited to IV applications in labor economics."

Overview

In this talk, I...

- ▷ review DML,
- ▷ highlight the importance of selecting & validating machine learners,
- ▷ discuss pairing DML with stacking (a.k.a. model averaging, 'super learning'),
- ▷ illustrate DML using multiple applications,
- ▷ formulate recommended practices,
- ▷ point you to our complementary Stata and R software to implement our recommendations.

Partially linear model

For simplicity, we focus (for now) on a commonly encountered model.

Assumption 1

$$Y = \tau D + m(X) + U, \quad E[U|D,X] = 0 \tag{1}$$

where Y=outcome, D=treatment, τ=target parameter, X=controls.

Partially linear model

For simplicity, we focus (for now) on a commonly encountered model.

Assumption 1

$$Y = \tau D + m(X) + U, \quad E[U|D, X] = 0 \tag{1}$$

where Y=outcome, D=treatment, τ=target parameter, X=controls.

Typical approach: we assume $m(X) = X'\beta$ and apply least squares.

Partially linear model

For simplicity, we focus (for now) on a commonly encountered model.

Assumption 1

$$Y = \tau D + m(X) + U, \quad E[U|D,X] = 0 \tag{1}$$

where Y=outcome, D=treatment, τ=target parameter, X=controls.

Typical approach: we assume $m(X) = X'\beta$ and apply least squares.

Important distinction:

- $E[U|D,X] = 0$ is an ***identifying assumption*** that is fundamental for the identification of τ.
- $m(X) = X'\beta$ is (usually) an ***assumption of convenience***.

Partially linear model

For simplicity, we focus (for now) on a commonly encountered model.

Assumption 1

$$Y = \tau D + m(X) + U, \quad E[U|D,X] = 0 \tag{2}$$

where Y=outcome, D=treatment, τ=target parameter, X=controls.

Typical approach: we assume $m(X) = X'\beta$ and apply least squares.

Why use something else than least squares?

- We have many controls (relative to the sample size), but do not know which to include.
- We suspect non-linear effects.

Partially linear model

For simplicity, we focus (for now) on a commonly encountered model.

Assumption 1

$$Y = \tau D + m(X) + U, \quad E[U|D,X] = 0 \tag{3}$$

where Y=outcome, D=treatment, τ=target parameter, X=controls.

Constructed moment condition gives familiar expression (Robinson, 1988):

$$E\left[(Y - E[Y|X] - \tau(D - E[D|X]))(D - E[D|X])\right] = 0$$

$$\Rightarrow \qquad \tau = \frac{E\left[(Y - E[Y|X])(D - E[D|X])\right]}{E\left[(D - E[D|X])^2\right]}.$$

Partially linear model

For simplicity, we focus (for now) on a commonly encountered model.

Assumption 1

$$Y = \tau D + m(X) + U, \quad E[U|D, X] = 0 \tag{3}$$

where Y=outcome, D=treatment, τ=target parameter, X=controls.

Constructed moment condition gives familiar expression (Robinson, 1988):

$$E\left[(Y - E[Y|X] - \tau (D - E[D|X]))(D - E[D|X])\right] = 0$$

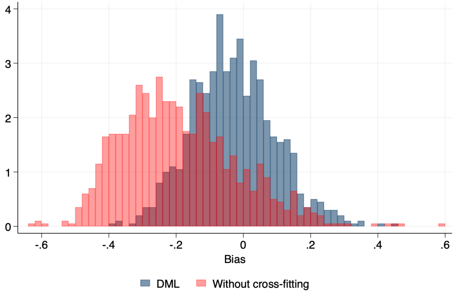
$$\Rightarrow \qquad \tau = \frac{E\left[(Y - E[Y|X])(D - E[D|X])\right]}{E\left[(D - E[D|X])^2\right]}.$$

Idea: Use ML to estimate conditional expectation functions (CEFs).

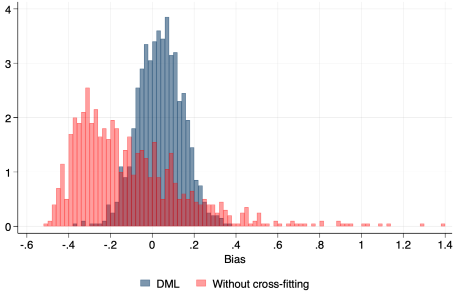
Over-fitting bias

Problem: plugging in ML estimates of CEFs generally induces an ***over-fitting bias***.

Figure: Estimating the PLR Coefficient with and without Cross-Fitting



(a) Gradient-boosted trees



(b) Feed-forward neural net

Double/Debiased Machine Learning

Double/Debiased Machine Learning (Chernozhukov et al., 2018)

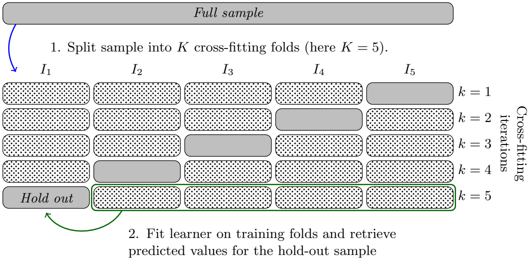
- two central ingredients:
  - ***sample-splitting/cross-fitting***
  - ***Neyman-orthogonal moment conditions***
- can be combined with a ***general class of ML methods***,
- requires only relatively mild rate requirements for asymptotic normality,
- can be used to estimate various target parameters (beyond the partially linear model coefficient).

Double/Debiased Machine Learning

The cross-fitting algorithm

1. splits the sample I randomly into K folds denoted $I_1, \ldots, I_K$,
2. fits CEF estimators iteratively on the sample excluding the hold-out fold, i.e., $I \setminus I_k$,
3. calculates the out-of-sample predicted values for the hold-out fold I_k,
4. and uses these *'cross-fitted'* predicted values to estimate the structural parameters on the full sample I.

Figure: Cross-fitting with one learner. Example: estimation of $\ell_0 = E[Y|X]$.



General framework

DML applies to a much more general framework with pre-specified, low-dimensional target parameter θ_0 that is identified by moment conditions of the form

$$E\left[\psi(W_i; \theta_0, \eta_0)\right] = 0 \tag{4}$$

where $\psi(\cdot)$ is a known score function, W_i is the observed data and η is a possibly high-dimensional nuisance parameter.

General framework

This framework accommodates:

- ***Partially linear model*** where we flexibly adjust for control variables (discussed above)

General framework

This framework accommodates:

- ***Partially linear model*** where we flexibly adjust for control variables (discussed above)
- Estimation of ***Average Treatment Effects*** (ATE), ***Average Treatment Effects on the Treated*** (ATET)

General framework

This framework accommodates:

- ***Partially linear model*** where we flexibly adjust for control variables (discussed above)
- Estimation of ***Average Treatment Effects*** (ATE), ***Average Treatment Effects on the Treated*** (ATET)
- ***Local Average Treatment Effects*** (LATE) with flexible controls

General framework

This framework accommodates:

- ***Partially linear model*** where we flexibly adjust for control variables (discussed above)
- Estimation of ***Average Treatment Effects*** (ATE), ***Average Treatment Effects on the Treated*** (ATET)
- ***Local Average Treatment Effects*** (LATE) with flexible controls
- ***Partially linear IV model*** with flexible controls

General framework

This framework accommodates:

- ***Partially linear model*** where we flexibly adjust for control variables (discussed above)
- Estimation of ***Average Treatment Effects*** (ATE), ***Average Treatment Effects on the Treated*** (ATET)
- ***Local Average Treatment Effects*** (LATE) with flexible controls
- ***Partially linear IV model*** with flexible controls
- ***Fully flexible IV estimation*** allowing for "many" instruments and "many" controls

General framework

This framework accommodates:

- ***Partially linear model*** where we flexibly adjust for control variables (discussed above)
- Estimation of ***Average Treatment Effects*** (ATE), ***Average Treatment Effects on the Treated*** (ATET)
- ***Local Average Treatment Effects*** (LATE) with flexible controls
- ***Partially linear IV model*** with flexible controls
- ***Fully flexible IV estimation*** allowing for "many" instruments and "many" controls
- ***Difference-in-differences estimation*** under conditional parallel trends and heterogeneous group effects

General framework

All these target parameters depend on conditional expectation functions, which constitute (possibly high-dimensional) ***nuisance functions***.

We can estimate these conditional expectation functions using machine learning.

Weakly causal parameters

There is another motivation for DDML:

- Linear residualization does not guarantee "weakly causal" parameters.
- Term coined by Blandhol et al. (2022): Positively weighted average of causal parameters. Viewed as minimum requirement.
- Blandhol et al. (2022) show this for TSLS. Arguments generalize to OLS (Angrist and Krueger, 1999).

Weakly causal parameters

There is another motivation for DDML:

- Linear residualization does not guarantee "weakly causal" parameters.
- Term coined by Blandhol et al. (2022): Positively weighted average of causal parameters. Viewed as minimum requirement.
- Blandhol et al. (2022) show this for TSLS. Arguments generalize to OLS (Angrist and Krueger, 1999).

The good news:

- DDML relies on moment functions that identify "weakly causal" parameters.

The choice of machine learner

Which machine learner should we use?

Which machine learner performs best in a particular application depends crucially on ***match quality of machine learner & structure of the DGP***.

- ▷ There is no general answer to the question of whether lasso or random forests will 'work' or will not 'work' in a given application.
- ▷ No-free lunch theorem in machine learning (Wolpert, 1996; Wolpert and Macready, 1997).
- ▷ Machine learners require 'tuning' (e.g., tree-depth, learning rate).

The choice of machine learner

For example, ***the lasso*** has become a popular tool in empirical economics.

- ▷ intuitive assumption of (approximate) sparsity
- ▷ computationally relatively cheap
- ▷ linearity has its advantages (e.g. extension to panel data; Belloni et al., 2016)

The choice of machine learner

For example, ***the lasso*** has become a popular tool in empirical economics.

- ▷ intuitive assumption of (approximate) sparsity
- ▷ computationally relatively cheap
- ▷ linearity has its advantages (e.g. extension to panel data; Belloni et al., 2016)

But there are also drawbacks:

- ▷ What if the ***sparsity assumption*** is not plausible?
  $\rightarrow$ "Illusion of Sparsity" (Giannone et al., 2021)
- ▷ There is a wide set of machine learners at disposal—lasso might not be the best choice for a particular application.

DDML+stacking

Stacking allows for ***combining multiple*** CEF estimators.

- constructs weighted average of ‘candidate’ learners
- performs asymptotically ***at least as well as the best-performing candidate learner*** if number of candidates grows at most at polynomial rate (der Laan et al., 2007; Polley et al., 2011)
- Model averaging techniques have a long tradition in economics and statistics, especially time-series (Crane and Crotty, 1967; Bates and Granger, 1969).
- Yet, despite its theoretical appeal, stacking is rarely used for the estimation of causal effects in economics or other social sciences.
- One exception: Van der Laan and Rose (2011) advocate for stacking (“super learning”) for Targeted MLE.

DDML+stacking

Stacking allows for ***combining multiple*** CEF estimators.

- constructs weighted average of ‘candidate’ learners
- performs asymptotically ***at least as well as the best-performing candidate learner*** if number of candidates grows at most at polynomial rate (der Laan et al., 2007; Polley et al., 2011)
- Model averaging techniques have a long tradition in economics and statistics, especially time-series (Crane and Crotty, 1967; Bates and Granger, 1969).
- Yet, despite its theoretical appeal, stacking is rarely used for the estimation of causal effects in economics or other social sciences.
- One exception: Van der Laan and Rose (2011) advocate for stacking (“super learning”) for Targeted MLE.

We illustrate: Stacking safeguards against ill-chosen/poorly tuned learners ***provided a generous and diverse*** set of base learners is included.

DDML+stacking

In each cross-fitting step $k = 1, \dots, K$,

$$\min_{w_{k,1},\dots,w_{k,J}} \sum_{i \in T_k} \left(Y_i - \sum_{j=1}^{J} w_{k,j} \hat{\ell}^{(j)}_{T^c_{k,v(i)}}(\boldsymbol{X}_i) \right)^2 \qquad \text{s.t. } w_{k,j} \geq 0,\ \sum_{j=1}^{J} |w_{k,j}| = 1.$$

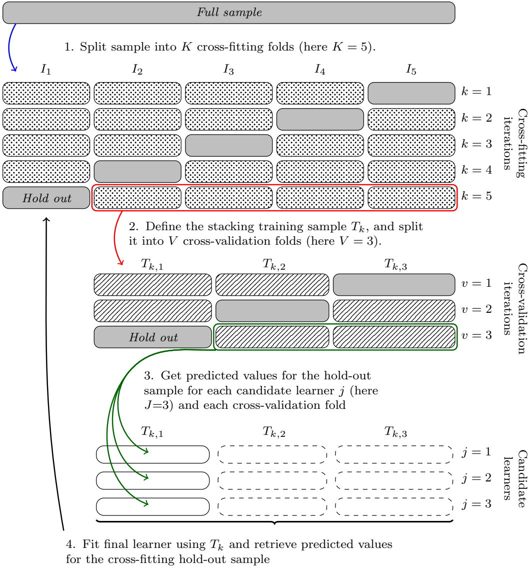
where $\hat{\ell}^{(j)}_{T^c_{k,v(i)}}(\boldsymbol{X}_i) \equiv$ cross-validated predicted values, $J \equiv$ number of candidate learners, $V \equiv \#$ CV folds.

Final stacking estimator: $\hat{\ell} = \sum_{j=1}^{J} \hat{w}_j \hat{\ell}_j$.

Other options: single-best learner, unconstrained OLS, unweighted average, etc.

Result of der Laan et al. (2007) does not require non-negativity or sum-to-one constraint.

Figure: Cross-fitting and stacking. Example: estimation of $\ell_0 = E[Y|X]$.



DDML+stacking

Two drawbacks of pairing DDML with (regular) stacking:

- computational complexity: $K \times V \times J$ learners are fit where $K =$ cross-fitting folds, $V =$ cross-validation folds, $J =$ number of candidate learners
- possibly sub-optimal performance in small samples given that learners are fit on $(K-1)(V-1)/(KV)\%$ of the sample

DDML+stacking

Two drawbacks of pairing DDML with (regular) stacking:

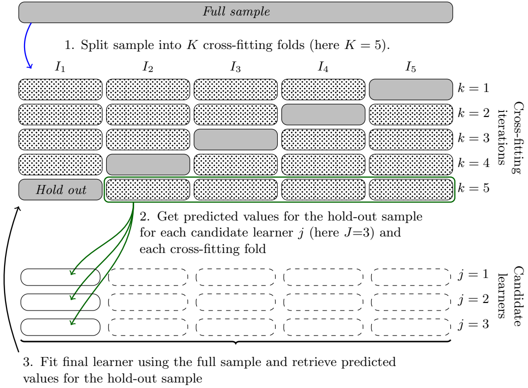
- computational complexity: $K \times V \times J$ learners are fit where $K =$ cross-fitting folds, $V =$ cross-validation folds, $J =$ number of candidate learners
- possibly sub-optimal performance in small samples given that learners are fit on $(K-1)(V-1)/(KV)\%$ of the sample

Short-stacking takes a short-cut by training the final learner on the cross-fitted values using the full sample. The objective function becomes:

$$\min_{w_1,\ldots,w_J} \sum_{i=1}^{n} \left(Y_i - \sum_{j=1}^{J} w_j \hat{\ell}^{(j)}_{I^c_{k(i)}}(\boldsymbol{X}_i) \right)^2 \qquad \text{s.t. } w_j \geq 0, \ \sum_j |w_j| = 1$$

where w_j are the short-stacking weights. Cross-fitting serves a ***double purpose***: addressing own-observation bias and yielding out-of-sample predicted values used for estimating weights.

Figure: Cross-fitting & *short*-stacking. Example: estimation of $\ell_0 = E[Y|X]$.



Advantages of DDML+Stacking

Calibrated simulation based on Poterba et al. (1995), who estimate the causal effect of 401(k) eligibility on wealth.

- outcome: log wealth
- treatment: 401(k) eligibility
- controls: age, income, education in years, family size, two-earner status, home ownership, and participation in two alternative pension schemes.
- $N = 9,915$

Simulation design: Reconstruct CEFs with either OLS (*linear DGP*) or gradient boosting (*nonlinear DGP*) to reinforce the linear/non-linear signal in the data.

Simulation design

- 1,000 bootstrap draws
- 10 candidate learners including OLS, CV-lasso/ridge, random forests, gradient boosting, feed-forward neural net Details

Table: Bias and Coverage Rates in the ***Linear DGP*** Table notes More results

	$n_b = 9{,}915$			$n_b = 99{,}150$		
Panel (A): Linear DGP	Bias	MAB	Rate	Bias	MAB	Rate
Full sample:						
DDML methods:						
OLS	49.9	793.8	0.95	−6.8	281.2	0.95
PDS-Lasso	48.4	787.1	0.95	−4.2	280.8	0.95
OLS	46.2	818.1	0.94	−6.9	283.1	0.95
Base learners						
Lasso with CV (2nd order poly)	50.9	806.6	0.95	−6.2	284.8	0.95
Ridge with CV (2nd order poly)	48.2	806.9	0.94	−6.9	283.7	0.96
Lasso with CV (10th order poly)	248.1	1034.5	0.94	55.9	285.9	0.95
Ridge with CV (10th order poly)	1230.1	1321.9	0.91	31.6	283.0	0.96
Random forest (low regularization)	−74.7	1031.3	0.89	−25.2	344.0	0.88
Random forest (high regularization)	69.1	891.2	0.94	−23.5	287.6	0.93
Gradient boosting (low regularization)	12.1	817.0	0.94	−24.2	285.1	0.96
Gradient boosting (high regularization)	114.8	823.8	0.94	66.9	285.6	0.95
Neural net	394.2	943.6	0.93	9.1	287.5	0.94
Meta learners						
Stacking: CLS	42.8	813.4	0.94	−7.5	282.9	0.96
Stacking: Single-best	43.7	819.8	0.94	−8.6	281.4	0.95
Short-stacking: CLS	45.0	794.9	0.94	−7.0	282.6	0.95
Short-stacking: Single-best	44.4	817.8	0.94	−8.3	281.9	0.95

Table: Bias and Coverage Rates in the ***Non-Linear DGP***

Table notes | More results

	n_b = 9,915			n_b = 99,150		
Panel (B): Non-Linear DGP	Bias	MAB	Rate	Bias	MAB	Rate
Full sample:						
OLS	−2588.9	2576.5	0.58	−2632.3	2611.5	0.
PDS-Lasso	−2598.7	2590.1	0.58	−2631.6	2609.5	0.
OLS	−2613.0	2634.2	0.58	−2635.4	2615.9	0.
DDML methods:						
Base learners						
Lasso with CV (2nd order poly)	703.7	1052.3	0.91	718.5	712.8	0.60
Ridge with CV (2nd order poly)	767.4	1080.8	0.90	729.3	724.0	0.60
Lasso with CV (10th order poly)	−4109.0	1799.9	0.90	7.4	306.5	0.94
Ridge with CV (10th order poly)	−5126.2	2215.7	0.89	9.6	307.8	0.94
Random forest (low regularization)	−96.1	1037.1	0.90	−37.5	328.0	0.87
Random forest (high regularization)	−159.7	904.4	0.94	−4.2	280.4	0.95
Gradient boosting (low regularization)	8.5	866.0	0.94	30.9	275.1	0.96
Gradient boosting (high regularization)	162.0	857.2	0.94	200.1	314.6	0.93
Neural net	−601.3	1063.9	0.93	−131.9	310.0	0.93
Meta learners						
Stacking: CLS	133.9	1049.5	0.94	37.8	271.0	0.95
Stacking: Single-best	−121.9	976.2	0.94	30.9	275.1	0.96
Short-stacking: CLS	162.7	865.1	0.94	33.6	266.3	0.95
Short-stacking: Single-best	71.7	868.4	0.94	30.9	275.1	0.96

Table: Average stacking weights

	Stacking		Short-stacking	
Panel (A): Linear DGP	$E[Y\|X]$	$E[D\|X]$	$E[Y\|X]$	$E[D\|X]$
OLS	0.668	0.501	0.692	0.492
Lasso with CV (2nd order poly)	0.105	0.144	0.118	0.130
Ridge with CV (2nd order poly)	0.068	0.054	0.068	0.063
Lasso with CV (10th order poly)	0.027	0.073	0.020	0.085
Ridge with CV (10th order poly)	0.033	0.043	0.024	0.057
Random forest (low regularization)	0.013	0.011	0.009	0.008
Random forest (high regularization)	0.017	0.024	0.013	0.024
Gradient boosting (low regularization)	0.030	0.043	0.020	0.040
Gradient boosting (high regularization)	0.020	0.060	0.018	0.060
Neural net	0.019	0.049	0.018	0.043
Panel (B): Non-Linear DGP	$E[Y\|X]$	$E[D\|X]$	$E[Y\|X]$	$E[D\|X]$
OLS	0.011	0.015	0.004	0.007
Lasso with CV (2nd order poly)	0.035	0.057	0.019	0.039
Ridge with CV (2nd order poly)	0.161	0.229	0.114	0.237
Lasso with CV (10th order poly)	0.053	0.080	0.048	0.062
Ridge with CV (10th order poly)	0.071	0.064	0.059	0.056
Random forest (low regularization)	0.045	0.011	0.043	0.005
Random forest (high regularization)	0.019	0.069	0.012	0.065
Gradient boosting (low regularization)	0.521	0.233	0.632	0.339
Gradient boosting (high regularization)	0.014	0.191	0.004	0.139
Neural net	0.071	0.051	0.064	0.049

Advantages of DDML+Stacking

As expected, OLS performs best in the fully linear setting and DDML+GB performs best in the when the nuisance function is generated by gradient boosting.

Advantages of DDML+Stacking

As expected, OLS performs best in the fully linear setting and DDML+GB performs best in the when the nuisance function is generated by gradient boosting.

In practice, researchers rarely know the functional structure in economic applications.

- ▷ Stacking & short-stacking assign high weights to the data-generating learner.
- ▷ Stacking reduces the ***burden of choice*** the researcher faces by allowing for the simultaneous consideration of multiple estimators.
- ▷ DDML paired with short-stacking performs very similar to DDML w/ regular stacking, despite lower computational burden (speed gain by factor $1/V$).

Computational time

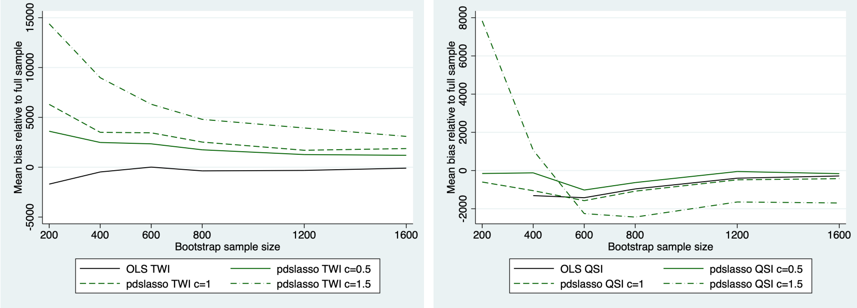
The bias in very small samples

A possible concern for machine learners is that they might not perform well for very small samples given that they are designed for, and typically applied to, large data sets.

Wüthrich and Zhu (2023) use simulations to demonstrate that PDS-Lasso suffers from a significant small sample bias and tends to underselect.

Using the 401(k) data (Poterba et al., 1995), they consider two competing specifications: Two-way interactions (TWI) and Quadratic spline & interactions (QSI).

The bias in very small samples



(a) Bias (TWI) (b) Bias (QSI)

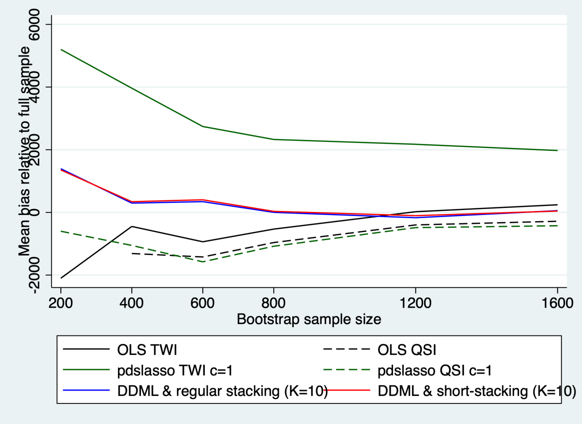
Notes: The figures report the mean bias calculated as the mean difference to the full sample estimates. Following WZ, we draw 600 bootstrap samples of size $n_b = \{200, 400, 600, 800, 1200, 1600\}$. 'TWI' indicates that the predictors have been expanded by two-way interactions. 'QSI' refers to the quadratic spline & interactions specification of Belloni et al. (2017).

Figure: Replication of Figure 8 in Wüthrich and Zhu (2023)

The bias in very small samples

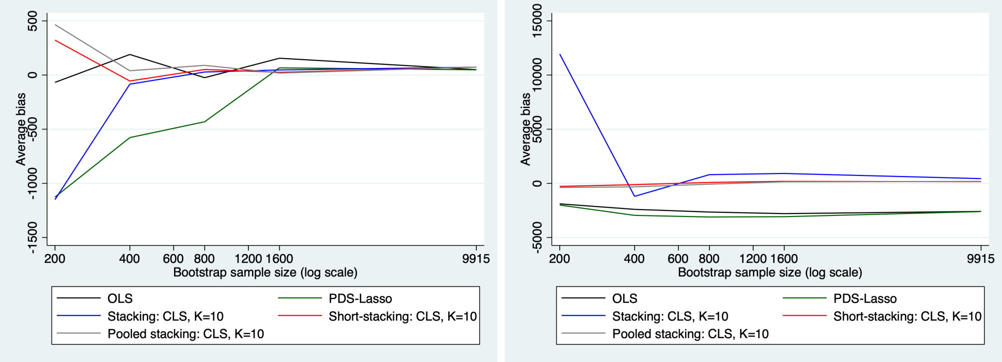
How do DDML paired with stacking or short-stacking perform in comparison?

Figure: Mean bias relative to full-sample estimates Table version



The bias in very small samples

Figure: Bias in ***calibrated simulation*** Coverage Table



(a) Linear DGP (b) Non-linear DGP

DDML with stacking or short-stacking perform well even for moderate sample size. Increasing K improves performance especially for small samples.

Applications

Three applications:

- ▷ Monopsony in the labor market
- ▷ Gender citation gap
- ▷ Gender wage gap

Monopsony in the labor market

Online platforms are an attractive setting for credible research designs using non-experimental data as users interact in a quasi-isolated setting.

$\rightarrow$ many context factors are observed

Monopsony in the labor market

Online platforms are an attractive setting for credible research designs using non-experimental data as users interact in a quasi-isolated setting.

$\rightarrow$ many context factors are observed

One example of a study using digital-trace data:

- Dube et al. (2020) examine monopsony power on the online platform MTurk.
- The parameter of interest is the ***labor supply elasticity***, a measure of market power.
- Outcome = log(duration)
- Treatment = log(reward)
- Controls = mix of textual (the tasks' title, description, and keywords) and non-textual variables (including time allocated for the task and required qualifications) measuring the type, complexity, and attractiveness of tasks.

Monopsony in the labor market

	Candidate learners							
	(1) Short-stacking	(2) OLS	(3) CV-lasso	(4) CV-ridge	(5) RF Low	(6) RF High	(7) XGBoost Low	(8) XGBoost High
Panel A. Median coefficient estimates with outcome log reward								
Log duration	-0.031^{***}	-0.379^{***}	-0.380^{***}	-0.379^{***}	-0.223^{***}	-0.263^{***}	-0.019^{***}	-0.034^{***}
	(0.005)	(0.005)	(0.005)	(0.005)	(0.004)	(0.004)	(0.004)	(0.005)
Panel B. Cross-fitted mean-squared prediction error								
Outcome	2.198	5.018	5.071	5.097	5.812	4.242	2.343	2.485
Treatment	0.398	0.874	0.874	0.881	1.504	0.898	0.556	0.409
Panel C. Short-stacking weights for each candidate learner								
Outcome	n/a	0.008	-0.000	0.000	-0.000	0.000	0.576	0.417
Treatment	n/a	0.000	0.000	-0.000	0.000	0.000	0.209	0.791

Notes: The table reports results from DML estimation with 5 cross-fitting folds and 5 cross-fitting repetitions. We employ median aggregation over the 5 repetitions. The number of observations is 258,352. The data is taken from Dube et al. (2020); the original data is from Ipeirotis (2010). The point estimate (standard error) reported by Dube et al. (2020, Table 1, col. 7) is -0.0299 (0.00402). Columns (2)-(8) pair DML with OLS, CV-lasso, CV-ridge, two types of random forest (400 trees, maximum tree depth of either 4 or 20) and two types of XGBoost (400 trees, maximum tree depth of either 4 or 20, early stopping after 10 iterations). Column (1) employs DML with the short-stacking strategy suggested in Ahrens et al. (2024) which relies on the candidate learners in Columns (2)-(8). Panel A report point estimates and standard errors. Panel B reports the (median) cross-fitted mean-squared prediction errors. Panel C shows the learner weights of the DML and short-stacking estimator. DML estimation uses the R package `ddml` (Wiemann et al., 2023).
$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Gender citation gap

back

It is well-documented that women are under-represented in academia (Ceci et al., 2014; Lundberg and Stearns, 2019).

- ▷ A possible reason is that scholarly work produced by women faces more sceptical scrutiny (Hengel, 2022; Krawczyk and Smyk, 2016).
- ▷ Higher scrutiny could be, for example, reflected in lower citations by other scholars (Card et al., 2020; Roberts et al., 2020; Grossbard et al., 2021).

Gender citation gap

back

It is well-documented that women are under-represented in academia (Ceci et al., 2014; Lundberg and Stearns, 2019).

- A possible reason is that scholarly work produced by women faces more sceptical scrutiny (Hengel, 2022; Krawczyk and Smyk, 2016).
- Higher scrutiny could be, for example, reflected in lower citations by other scholars (Card et al., 2020; Roberts et al., 2020; Grossbard et al., 2021).

We examine average differences in citations of articles published in top-30 economic journals from 1983 to 2020 by the gender composition of the authors ($N = 27\,599$).

Gender citation gap

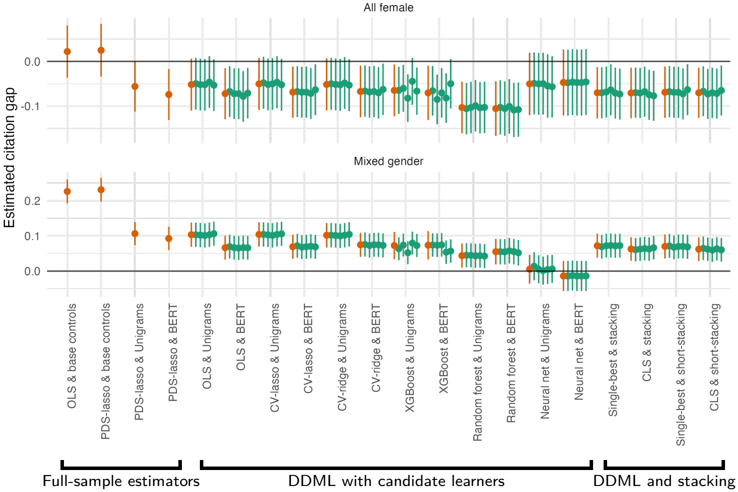
- Outcome = log-citations,
- Treatment = indicator for all-female authorship; mixed-gender authorship; gender is imputed from the author names using `Namsor` (Sebo, 2021; Krstovski et al., 2023).
- Controls = we leverage the abstract text as a proxy for the topic and quality of the article.

Gender citation gap

- Outcome = log-citations,
- Treatment = indicator for all-female authorship; mixed-gender authorship; gender is imputed from the author names using `Namsor` (Sebo, 2021; Krstovski et al., 2023).
- Controls = we leverage the abstract text as a proxy for the topic and quality of the article.

Practical challenge: how to encode text data

Figure: The citation gap by authors' gender composition



Gender citation gap

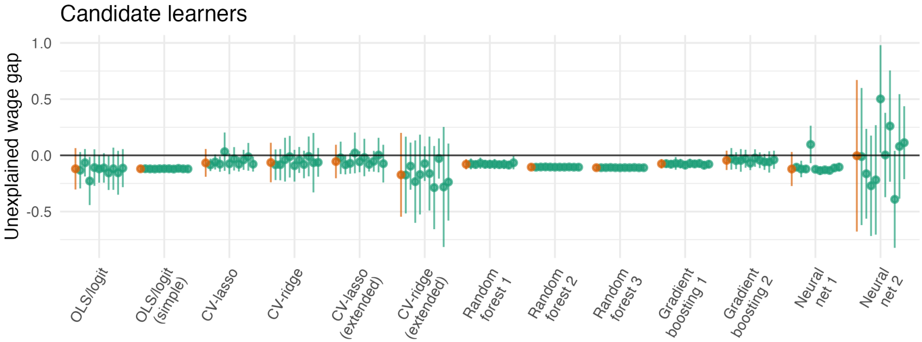
Table: Stacking weights in the gender citation gap application.

	Citations		*All female*		*Mixed gender*	
	Conv.	*Short*	*Conv.*	*Short*	*Conv.*	*Short*
OLS & Unigrams	0.1	0.069	0.113	0.155	0.135	0.114
OLS & BERT	0.35	0.368	0.09	0.106	0.174	0.196
CV-lasso & Unigrams	0.	0.	0.049	0.013	0.	0.
CV-lasso & BERT	0.119	0.102	0.363	0.396	0.129	0.166
CV-ridge & Unigrams	0.	0.	0.	0.	0.	0.
CV-ridge & BERT	0.	0.	0.361	0.31	0.383	0.329
XGBoost & Unigrams	0.217	0.236	0.008	0.008	0.017	0.028
XGBoost & BERT	0.171	0.174	0.016	0.01	0.033	0.035
Random forest & Unigrams	0.047	0.055	0.02	0.026	0.149	0.151
Random forest & BERT	0.	0.	0.001	0.	0.	0.
Neural net & Unigrams	0.	0.	0.	0.	0.	0.
Neural net & BERT	0.	0.	0.	0.	0.	0.

Gender wage gap

- Country: UK
- Data: OECD
- Unconditional wage gap = -.1434 (s.e.=0.017)
- Number of observations = 4,889, $K = 10$
- AIPW estimator
- Covariates:
  - Categorical (21): part-time, industry, education, occupation, health status, management position, number of children, etc.
  - Continuous (5): age, tenure, literacy & numeracy, years of education
- Three sets of control specifications: "reduced" (only age, education, tenure), "baseline" (all variables) and "extended" (full interactions).

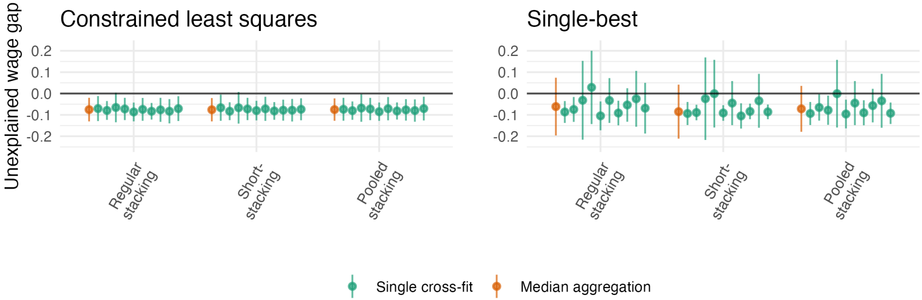
Figure: Unexplained gender wage gap 1/2



Gender wage gap

Details

Figure: Unexplained gender wage gap 2/2



Gender wage gap

Table: Stacking weights in the gender wage gap application.

	Conventional stacking			*Short-stacking*			*Mean-squared error*		
	$g_0(0,X)$	$g_0(1,X)$	$m_0(X)$	$g_0(0,X)$	$g_0(1,X)$	$m_0(X)$	$g_0(0,X)$	$g_0(1,X)$	$m_0(X)$
OLS/logit	0.023	0.012	0.242	0.027	0.013	0.211	0.369	0.347	0.161
OLS/logit (simple)	0.004	0.	0.	0.	0.	0.	0.267	0.204	0.223
CV-lasso	0.103	0.136	0.109	0.03	0.076	0.047	0.236	0.178	0.16
CV-ridge	0.189	0.04	0.064	0.225	0.024	0.108	0.237	0.18	0.161
CV-lasso (extended)	0.041	0.157	0.016	0.035	0.266	0.002	0.238	0.18	0.161
CV-ridge (extended)	0.011	0.04	0.011	0.003	0.024	0.022	0.336	0.194	0.161
Random forest 1	0.435	0.506	0.275	0.483	0.507	0.28	0.23	0.176	0.161
Random forest 2	0.	0.	0.	0.	0.	0.	0.258	0.19	0.171
Random forest 3	0.	0.	0.	0.	0.	0.	0.274	0.199	0.179
Gradient boosting 1	0.025	0.008	0.039	0.011	0.003	0.022	0.239	0.183	0.16
Gradient boosting 2	0.15	0.059	0.216	0.175	0.063	0.285	0.254	0.196	0.161
Neural net 1	0.013	0.022	0.	0.	0.	0.	0.349	0.263	0.241
Neural net 2	0.008	0.02	0.027	0.01	0.023	0.023	0.643	0.357	0.176

Key recommendations

R1. Employ DDML paired with stacking or short-stacking with a diverse and generous set of candidate learners, including OLS.

R2. If the sample size is small, increase the number of folds and repeat the cross-fitting exercise.

R3. Inspect the (short-)stacking weights to adjust and refine learner settings.

Key takeaways

- DDML & stacking approaches ***safeguard against ill-chosen learners*** provided a diverse set of candidate learners is chosen.
- DDML paired with ***short-stacking performs comparably to regular stacking***—and in small samples even better — while being ***computationally cheaper***.
- DDML allows weakening ***assumptions of convenience*** such as linearity, allowing researchers to focus on specifying ***identifying assumptions***.

More info

Software

- Stata — The packages `ddml` and `pystacked` are available on Github/SSC. See `https://statalasso.github.io/` for info.
- R — The package `ddml` is available from CRAN. See `https://thomaswiemann.com/ddml/` for info.

References I

Ahrens, A., Hansen, C. B., Schaffer, M. E., and Wiemann, T. (2024). Model averaging and double machine learning.

Angrist, J. D. and Frandsen, B. (2022). Machine labor. *Journal of Labor Economics*, 40(S1):S97–S140.

Angrist, J. D. and Krueger, A. B. (1999). Empirical strategies in labor economics. In *Handbook of Labor Economics*, volume 3, pages 1277–1366. Elsevier.

Bates, J. and Granger, C. (1969). The combination of forecasts. *Journal of the Operational Research Society*, 20(4):451–468.

Belloni, A., Chernozhukov, V., Fernández-Val, I., and Hansen, C. (2017). Program Evaluation and Causal Inference With High-Dimensional Data. *Econometrica*, 85(1):233–298.

Belloni, A., Chernozhukov, V., and Hansen, C. (2014). Inference on treatment effects after selection among high-dimensional controls. *Review of Economic Studies*, 81:608–650.

References II

Belloni, A., Chernozhukov, V., Hansen, C., and Kozbur, D. (2016). Inference in High Dimensional Panel Models with an Application to Gun Control. *Journal of Business & Economic Statistics*, 34(4):590–605. Genre: Methodology.

Beraja, M., Kao, A., Yang, D. Y., and Yuchtman, N. (2023). AI-tocracy. *The Quarterly Journal of Economics*, 138(3):1349–1402.

Blandhol, C., Bonney, J., Mogstad, M., and Torgovitsky, A. (2022). When is TSLS actually LATE? *BFI Working Paper*, (2022-16).

Card, D., DellaVigna, S., Funk, P., and Iriberri, N. (2020). Are Referees and Editors in Economics Gender Neutral?*. *The Quarterly Journal of Economics*, 135(1):269–327.

Ceci, S. J., Ginther, D. K., Kahn, S., and Williams, W. M. (2014). Women in Academic Science: A Changing Landscape. *Psychological Science in the Public Interest: A Journal of the American Psychological Society*, 15(3):75–141.

References III

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68. tex.ids= Chernozhukov2018a publisher: John Wiley & Sons, Ltd (10.1111).

Crane, D. B. and Crotty, J. R. (1967). A two-stage forecasting model: Exponential smoothing and multiple regression. *Management Science*, 13(8):B–501–B–507.

der Laan, M. J. V., Polley, E. C., and Hubbard, A. E. (2007). Super learner. *Statistical Applications in Genetics and Molecular Biology*, 6(1).

Dube, A., Jacobs, J., Naidu, S., and Suri, S. (2020). Monopsony in Online Labor Markets. *American Economic Review: Insights*, 2(1):33–46.

Giannone, D., Lenza, M., and Primiceri, G. E. (2021). Economic predictions with big data: The illusion of sparsity. *Econometrica*, 89(5):2409–2437.

References IV

Goller, D., Lechner, M., Moczall, A., and Wolff, J. (2020). Does the estimation of the propensity score by machine learning improve matching estimation? The case of Germany's programmes for long term unemployed. *Labour Economics*, 65:101855.

Grossbard, S., Yilmazer, T., and Zhang, L. (2021). The gender gap in citations of articles published in two demographic economics journals. *Review of Economics of the Household*, 19(3):677–697.

Hengel, E. (2022). Publishing While Female: are Women Held to Higher Standards? Evidence from Peer Review. *The Economic Journal*, 132(648):2951–2991.

Ipeirotis, P. G. (2010). Analyzing the amazon mechanical turk marketplace. *XRDS: Crossroads, The ACM magazine for students*, 17(2):16–21.

Krawczyk, M. and Smyk, M. (2016). Author's gender affects rating of academic articles: Evidence from an incentivized, deception-free laboratory experiment. *European Economic Review*, 90:326–335.

References V

Krstovski, K., Lu, Y., and Xu, Y. (2023). Inferring gender from name: a large scale performance evaluation study.

Lundberg, S. and Stearns, J. (2019). Women in Economics: Stalled Progress. *Journal of Economic Perspectives*, 33(1):3–22.

Polley, E. C., Rose, S., and van der Laan, M. J. (2011). Super learning. In *Targeted learning: Causal inference for observational and experimental data*, pages 43–66. Springer New York, New York, NY.

Poterba, J. M., Venti, S. F., and Wise, D. A. (1995). Do 401 (k) contributions crowd out other personal saving? *Journal of Public Economics*, 58(1):1–32.

Roberts, M. E., Stewart, B. M., and Nielsen, R. A. (2020). Adjusting for Confounding with Text Matching. *American Journal of Political Science*, 64(4):887–903.

References VI

Robinson, P. M. (1988). Root-N-Consistent Semiparametric Regression. *Econometrica*, 56(4):931. ISBN: 00129682.

Sebo, P. (2021). Performance of gender detection tools: a comparative study of name-to-gender inference services. *Journal of the Medical Library Association : JMLA*, 109(3):414–421.

Van der Laan, M. J. and Rose, S. (2011). *Targeted learning: causal inference for observational and experimental data*, volume 4. New York: Springer.

Wiemann, T., Ahrens, A., Hansen, C. B., and Schaffer, M. E. (2023). *ddml: Double/Debiased Machine Learning in R*. https://github.com/thomaswiemann/ddml.

Wolpert, D. and Macready, W. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82.

References VII

Wolpert, D. H. (1996). The lack of a priori distinctions between learning algorithms. *Neural computation*, 8(7):1341–1390.

Wüthrich, K. and Zhu, Y. (2023). Omitted variable bias of lasso-based inference methods: A finite sample analysis. *Review of Economics and Statistics*, 105(4):982–997.