An Introduction to Double/Debiased Machine Learning

Achim Ahrens Victor Chernozhukov Christian Hansen
Damian Kozbur Mark Schaffer Thomas Wiemann

CERGE-EI

November 18, 2025

Motivation

Researchers are often interested in ***one or a few (causal) target parameters*** summarizing the relationship between key variables.

Motivation

Researchers are often interested in ***one or a few (causal) target parameters*** summarizing the relationship between key variables.

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

- ▷ confounders
- ▷ instrumental variables

Motivation

Example 1 (Monopsony power)

Dube et al. (2020) ask whether employers have market power. Using data from MTurk, they estimate

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

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

θ is a proxy for the negative labor supply elasticity (a measure of market power). For θ to have a meaninful interpretation, we should adjust for job characteristics (X_i), some of which come in text format (title, description, keywords).

Motivation

Example 2 (Difference-in-differences with staggered adoption)

In DiD designs with staggered adoption, we target the group-time average treatment effect on the treated (GT-ATT) (e.g., Callaway and Sant'Anna, 2021):

$$\theta_0^{ATT(g,t)} \equiv E[Y_{it}(1) - Y_{it}(0)|D_{it} = 1, G_i = g],$$

where G_i is the time of adoption, $D_{it} \equiv \mathbb{1}\{G_i \leq t\}$ is the treatment indicator, and $(Y_{it}(1), Y_{it}(0))$ are the potential outcomes.

The parallel trends assumption in a difference-in-differences design may only hold *conditional* on additional pre-treatment covariates X_i.

Motivation

These examples have two aspects in common:

1. there is a ***low-dimensional parameter of interest*** (e.g., labor supply elasticity),
2. estimation of this parameter potentially depends on a ***high-dimensional nuisance object*** (e.g., associated with text controls).

Motivation

These examples have two aspects in common:

1. there is a ***low-dimensional parameter of interest*** (e.g., labor supply elasticity),
2. estimation of this parameter potentially depends on a ***high-dimensional nuisance object*** (e.g., associated with text controls).

High-dimensionality can arise

- ▷ with few covariates if there is no known parametric functional form,
- ▷ with many covariates if researchers are willing to assume a known parametric functional form,
- ▷ with a combination of both (e.g., text or image data).

Motivation

Formally, in all these examples, there is a ***user-specified moment condition*** such that

$$E[m(W; \theta_0, \eta_0)] = 0$$

- W is the "data,"
- θ_0 is the unknown parameter of interest,
- η_0 is a so-called nuisance (or auxiliary) parameter; potentially high-dimensional.

In many interesting examples (but not always), η_0 will be a vector of conditional expectation functions.

Motivation

The ***semi-parametric problem*** has a long history in econometrics and statistics (e.g., Bickel, 1982; Robinson, 1988a; Newey, 1994).

Recently, methods have been suggested that ***leverage supervised machine learning*** to aid causal effect estimation; e.g., the PDS-lasso estimator of Belloni et al. (2014).

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

Overview

A central method that has received much attention is

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

Overview

A central method that has received much attention is

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

In this talk, we...

- ▷ review DML and its two central "ingredients,"
- ▷ illustrates its usage in two applications,
- ▷ formulate recommended practices.

Partially linear regression

Let's return to the monopsony example of Dube et al. (2020).

Model 1 (Partially Linear Regression; PLR)

$$Y = \theta D + g(X) + U, \quad E[U|D, X] = 0 \tag{2.1}$$

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

Typical approach: we assume $g(X) = X'\beta$ and apply least squares. Here, β is the *nuisance parameter*.

Partially linear regression

Let's return to the monopsony example of Dube et al. (2020).

Model 1 (Partially Linear Regression; PLR)

$$Y = \theta D + g(X) + U, \quad E[U|D, X] = 0 \tag{2.1}$$

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

Typical approach: we assume $g(X) = X'\beta$ and apply least squares. Here, β is the *nuisance parameter*.

Problem: some of the confounders include text data. It is difficult to find a credible *ad hoc* linear representation.

Partially linear regression

A commonly encountered model:

Model 1 (Partially Linear Regression; PLR)

$$Y = \theta D + g(X) + U, \quad E[U|D, X] = 0 \tag{2.2}$$

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

Linear regression is equivalent to estimation based on the FWL-style moment condition

$$m_{LM}(W; \theta, \eta) = [(Y - \eta_Y X) - \theta(D - \eta_D X)](D - \eta_D X).$$

Partially linear regression

A commonly encountered model:

Model 1 (Partially Linear Regression; PLR)

$$Y = \theta D + g(X) + U, \quad E[U|D, X] = 0 \tag{2.2}$$

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

More generally, we can consider the moment condition

$$m(W; \theta, \eta) = [(Y - \ell(X)) - \theta(D - r(X))]\,(D - r(X)),$$

where the nuisance parameters are $\eta(X) = (\ell(X), r(X))$ with true values $\ell_0(X) = E[Y|X]$ and $r_0(X) = E[D|X]$.

Partially linear regression

A commonly encountered model:

Model 1 (Partially Linear Regression; PLR)

$$Y = \theta D + g(X) + U, \quad E[U|D, X] = 0 \tag{2.2}$$

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

From this, we get the "generalized" FWL-style estimator (Robinson, 1988b):

$$\theta = \frac{E\left[\left(Y - \ell(X)\right)\left(D - r(X)\right)\right]}{E\left[\left(D - r(X)\right)^2\right]}.$$

Partially linear regression

A commonly encountered model:

Model 1 (Partially Linear Regression; PLR)

$$Y = \theta D + g(X) + U, \quad E[U|D, X] = 0 \tag{2.2}$$

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

From this, we get the "generalized" FWL-style estimator (Robinson, 1988b):

$$\theta = \frac{E\left[(Y - \ell(X))\,(D - r(X))\right]}{E\left[(D - r(X))^2\right]}.$$

Note: Target parameter depends on potentially high-dimensional nuisances; these are the conditional expectation functions.

Two biases

Naive approach:

1. Estimate nuisance parameter η_0 using supervised machine learning,
2. Plug $\widehat{\eta}$ into estimator to obtain estimate of θ_0.

Two biases

Naive approach:

1. Estimate nuisance parameter η_0 using supervised machine learning,
2. Plug $\widehat{\eta}$ into estimator to obtain estimate of θ_0.

Two biases:

If the estimator of η_0 is a "flexible," slow-converging estimator ($\rightsquigarrow$ modern supervised ML), the dependence on the first step leads to two sources of bias:

- ▷ ***regularization bias*** arises because the estimation error $(\check{\eta} - \eta_0)$ is "too large,"
- ▷ ***overfitting bias*** appears because $(\check{\eta} - \eta_0)$ is "strongly" correlated with W_i.

Two biases

Naive approach:

1. Estimate nuisance parameter η_0 using supervised machine learning,
2. Plug $\widehat{\eta}$ into estimator to obtain estimate of θ_0.

Two biases:

If the estimator of η_0 is a "flexible," slow-converging estimator ($\rightsquigarrow$ modern supervised ML), the dependence on the first step leads to two sources of bias:

- ▷ ***regularization bias*** arises because the estimation error $(\check{\eta} - \eta_0)$ is "too large,"
- ▷ ***overfitting bias*** appears because $(\check{\eta} - \eta_0)$ is "strongly" correlated with W_i.

These two biases are addressed through ***two essential ingredients*** that define DML.

Ingredient 1: Neyman orthogonality

Neyman orthogonality is a local ***robustness property*** of the score function in that it decreases the sensitivity of the plug-in estimator to the nuisance parameter.

Definition 1 (Neyman orthogonality)

A Neyman orthogonal score ψ satisfies

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

$$\frac{\partial}{\partial \lambda}\{E\left[\psi(W_i; \theta_0, \eta_0 + \lambda(\eta - \eta_0))\right]\}|_{\lambda=0} = 0, \quad \forall \eta \in \mathcal{T},$$

where η and λ can be interpreted as the direction and magnitude of a deviation away from the true nuisance parameter η_0, respectively.

NO alleviates the regularization bias, and ensures that inference about θ_0 do not depend on the precise statistical properties of the nuisance estimator $\check{\eta}$.

Ingredient 1: Neyman orthogonality

Let's take consider as an example:

Model 2 (ATE)

A central policy-relevant parameter is the average treatment effect (ATE) of a binary treatment D on an outcome Y. Consider the parameter

$$\theta_0 \equiv E\left[Y(1) - Y(0)\right]$$

where $Y(d)$ is the potential outcome under treatment status $d \in \{0, 1\}$.

Ingredient 1: Neyman orthogonality*

Two commonly used scores are the inverse propensity weighted (IPW) score and the augmented IPW (AIPW) score:

$$m_{IPW}(W;\theta,\alpha) \equiv \alpha(D,X)Y - \theta,$$
$$m_{AIPW}(W;\theta,\eta) \equiv \alpha(D,X)(Y - \ell(D,X)) + \ell(1,X) - \ell(0,X) - \theta,$$

where $W = (Y, D, X)$. The true value of the nuisance parameters are $\alpha_0(D,X) \equiv \frac{D}{r_0(X)} - \frac{(1-D)}{1-r_0(X)}$, $r_0(X) \equiv E[D|X]$, and $\ell_0(D,X) \equiv E[Y|D,X]$.

Ingredient 1: Neyman orthogonality*

Two commonly used scores are the inverse propensity weighted (IPW) score and the augmented IPW (AIPW) score:

$$m_{IPW}(W;\theta,\alpha) \equiv \alpha(D,X)Y - \theta,$$
$$m_{AIPW}(W;\theta,\eta) \equiv \alpha(D,X)(Y - \ell(D,X)) + \ell(1,X) - \ell(0,X) - \theta,$$

where $W = (Y, D, X)$. The true value of the nuisance parameters are $\alpha_0(D,X) \equiv \frac{D}{r_0(X)} - \frac{(1-D)}{1-r_0(X)}$, $r_0(X) \equiv E[D|X]$, and $\ell_0(D,X) \equiv E[Y|D,X]$.

Remark 1: Target parameter depends again on conditional expectation functions (CEFs).

Ingredient 1: Neyman orthogonality*

Two commonly used scores are the inverse propensity weighted (IPW) score and the augmented IPW (AIPW) score:

$$m_{IPW}(W;\theta,\alpha) \equiv \alpha(D,X)Y - \theta,$$
$$m_{AIPW}(W;\theta,\eta) \equiv \alpha(D,X)(Y - \ell(D,X)) + \ell(1,X) - \ell(0,X) - \theta,$$

where $W = (Y, D, X)$. The true value of the nuisance parameters are $\alpha_0(D,X) \equiv \frac{D}{r_0(X)} - \frac{(1-D)}{1-r_0(X)}$, $r_0(X) \equiv E[D|X]$, and $\ell_0(D,X) \equiv E[Y|D,X]$.

Remark 1: Target parameter depends again on conditional expectation functions (CEFs).

Remark 2: The commonly considered inverse propensity weighted (IPW) score *is not* Neyman orthogonal! The doubly robust score *is* Neyman orthogonal.

Ingredient 2: Cross-fitting

Overfitting bias, also referred to as ***own-observation bias***, arises if the error in the nuisance parameter estimator of η_0 is strongly related to the data used in constructing the plug-in estimator of θ_0.

Sample-splitting idea: If we had access to two independent samples, we could use one for nuisance estimation and one for target parameter estimation. Ensures independence between the estimation error and the data.

Ingredient 2: Cross-fitting

A more general approach that makes efficient use of the data is cross-fitting.

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.

DML in brief

DML = Neyman-orthogonal scores + cross-fitting.

DML in brief

DML = Neyman-orthogonal scores + cross-fitting.

DML only imposes mild convergence rate requirements on the nuisance function estimators, justifying the use of a wide class of machine learners.

DML in brief

DML = Neyman-orthogonal scores + cross-fitting.

DML only imposes mild convergence rate requirements on the nuisance function estimators, justifying the use of a wide class of machine learners.

DML estimators are asymptotically normally distributed.

DML in brief

DML = Neyman-orthogonal scores + cross-fitting.

DML only imposes mild convergence rate requirements on the nuisance function estimators, justifying the use of a wide class of machine learners.

DML estimators are asymptotically normally distributed.

Practical advantages:

- ▷ the DML standard errors are simply the standard errors that would be used if the nuisance parameter estimates were taken as the true parameters,
- ▷ can be used to estimate a wide range of target parameters,
- ▷ can be used with a wide set of nuisance function estimators, incl. popular machine learners,
- ▷ readily available software for Stata, R, Python (Ahrens et al., 2024; Bach et al., 2021, 2022).

Importance of cross-fitting: Many IV simulation

The first simulation revisits the classical many IV setting. This topic has been investigated in many papers since the mid-1990s (e.g., Bekker, 1994).

One of the recommendations to emerge from this literature is to use sample-splitting in the first stage of IV in order to reduce overfitting bias. — In fact, DML with linear regression is essentially just split-sample IV (Angrist and Krueger, 1995; Angrist et al., 1999).

Importance of cross-fitting: Many IV simulation

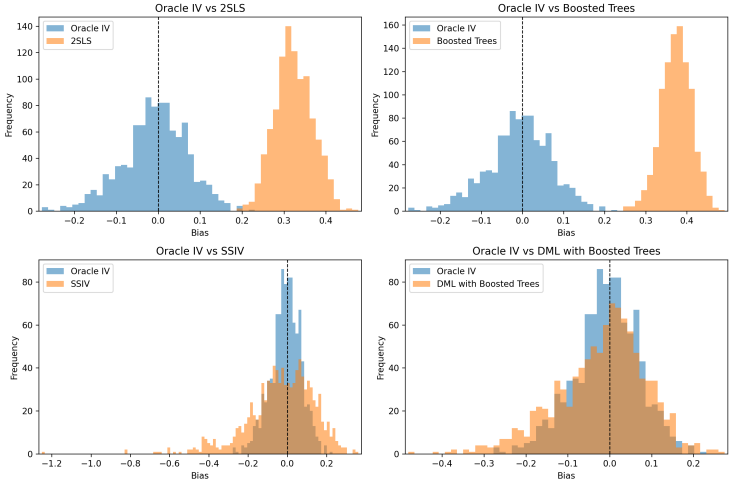
The first simulation revisits the classical many IV setting. This topic has been investigated in many papers since the mid-1990s (e.g., Bekker, 1994).

One of the recommendations to emerge from this literature is to use sample-splitting in the first stage of IV in order to reduce overfitting bias. — In fact, DML with linear regression is essentially just split-sample IV (Angrist and Krueger, 1995; Angrist et al., 1999).

Simulation approach:

- ▷ First stage DGP is sparse linear, with $p = 200$ and only 6 instruments have a signal. $n = 1000$.
- ▷ Nuisance estimator: linear regression and gradient boosting trees (maximum tree depth = 4, a learning rate = 0.1).

Importance of cross-fitting: Many IV simulation



Notes. Each panel compares simulation performance of a feasible IV estimator (orange) to the infeasible oracle IV estimator (blue).

Importance of orthogonality: IPW vs AIPW

Research question: Following Poterba et al. (1995), we estimate the average treatment effect (ATE) of 401(k) eligibility (D_i) on net financial assets (Y_i):

$$\theta_0^{ATE} = E[Y_i(1) - Y_i(0)].$$

Identification strategy: 401(k) eligibility is assumed as good as randomly assigned once we condition on demographic controls X_i (age, income, education, family size, two-earner status, home ownership, alternative pension participation).

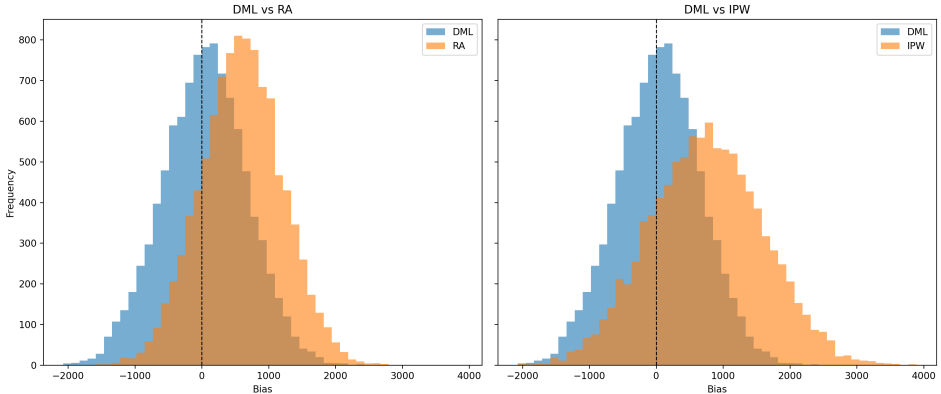
401(k) Eligibility and Net Financial Assets

Estimation approaches: IPW, Regression adjustment (RA), Doubly robust (DR) *with cross-fitting*

Simulation setting: Calibrated simulation where we fix ATE at $\approx$ 6889.

Nuisance function estimator: Random forests with 1000 trees and a maximum depth of 8 and 4 for the outcome and treatment propensity functions, respectively.

Importance of orthogonality: IPW vs AIPW



Notes. Each panel compares simulation performance of the DML estimator (blue) to a cross-fit ATE estimator based on a score that is not Neyman orthogonal (orange). Specifically, the left panel compares DML to a cross-fit regression adjustment (RA) estimator, and the right panel compares DML to a cross-fit IPW estimator.

DML and Staggered Adoption DiD*

Motivation: Recent work on DiD accounts for heterogeneous treatment effects in staggered adoption designs (de Chaisemartin and d'Haultfoeuille, 2023; Roth et al., 2023).

Empirical Illustration:

- ▷ Data: Balanced panel of 656 US households (HRS, waves 7–11).
- ▷ Original study: Effect of hospital admission on out-of-pocket (OOP) spending (Dobkin et al., 2018; Sun and Abraham, 2021).
- ▷ We consider *relaxing the parallel trends assumption* to only hold conditional on pre-treatment controls including age, gender, race, and education.

DML and Staggered Adoption DiD*

A key insight about the estimation of the GT-ATT is that for each group-time pair, the target parameter is equivalent to a conventional ATT identified under conditional unconfoundedness where the outcome is replaced by a first-difference in outcomes.

Two-Step Analysis:

1. For each pair (t, g), estimate change in OOP spending at time t due to hospitalization at time g.
2. Aggregate these estimates to draw inference on the dynamic effects of hospitalization.

DML and Staggered Adoption DiD*

Wave first hospitalized	Wave	*Cross-fitting Repetitions*					Median aggregate	Const.	Doubly robust
		Rep. 1	Rep. 2	Rep. 3	Rep. 4	Rep. 5			
		(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
8	8	2469.2	2458.5	2412.1	2542.4	2524.5	2469.2	3028.6	2200.5
		(792.7)	(787)	(791)	(783.8)	(784.5)	(787.2)	(913.5)	(839)
	9	516.4	512.7	509.8	555.8	554.4	516.4	1247.7	-14.3
		(628.7)	(634.8)	(637)	(597.8)	(597)	(628.7)	(860.7)	(681)
	10	759.8	770.3	774.1	724.3	872.2	770.3	800.1	998.9
		(601)	(578.1)	(583.9)	(579.1)	(578.7)	(583.9)	(1007.5)	(570.2)
9	8	-1334.7	-1221.7	-1227.5	-1405.6	-1248.4	-1248.4	-170	-1717.8
		(936.6)	(951.8)	(922.7)	(1020.3)	(979)	(952.2)	(1128.4)	(1324.8)
	9	3300.3	3180.1	3221.9	3454.7	3244	3244	3324.4	3789.9
		(900.3)	(948.2)	(897)	(996.5)	(981.3)	(950.4)	(958.8)	(1338.6)
	10	1010.3	1071.8	979.8	935.1	1019.1	1010.3	106.8	2533.6
		(441.5)	(437.4)	(439)	(432.2)	(430.3)	(440)	(650.7)	(421.7)
10	8	2850.9	2390.3	2201.4	2187.5	2434.8	2390.3	37.9	710.6
		(2191.2)	(2071.2)	(1959)	(1930.4)	(2102.6)	(2071.2)	(1403.4)	(1530.6)
	9	-287.3	-268.2	-236.8	-354.4	-295.3	-287.3	-410.6	-246.7
		(1022.4)	(1019.1)	(1022.3)	(1018.2)	(1022.1)	(1022.1)	(1027.1)	(1026.2)
	10	2628.4	2880.9	2715.2	2413.2	2585.5	2628.4	3091.5	3796.1
		(1234.7)	(1240.9)	(1234.5)	(1344.5)	(1215.5)	(1237.5)	(995.4)	(931.6)

Notes: Columns (1)-(5) show DML GT-ATT estimators for randomly generated cross-fitting folds. Column (6) presents the corresponding median aggregated DML estimate. Columns (7) and (8) present the Sant'Anna and Zhao (2020) estimator without and with controls, respectively. Point-wise standard errors are in parentheses.

Practical implementation*

Challenge:

- ▷ Cross-fitting adds an additional layer of randomness.
- ▷ The problem is striking here, since each GT-ATT is estimates on a relatively small sample. For example, the estimated first-period treatment effect for group 9 is based on 176 treated and 404 untreated observations, respectively. Estimates range from 3180 to 3454.

Practical implementation*

Recommendation 1 (Number of folds)

Check sensitivity of DML estimates w.r.t. the number of folds. Velez (2024) shows for a specific setting that $n = K$ is optimal, but simulation evidence suggest moderate K work well for moderate sample size. With very small sample size, it's worthwhile to consider K closer to n.

Practical implementation*

Recommendation 1 (Number of folds)

Check sensitivity of DML estimates w.r.t. the number of folds. Velez (2024) shows for a specific setting that $n = K$ is optimal, but simulation evidence suggest moderate K work well for moderate sample size. With very small sample size, it's worthwhile to consider K closer to n.

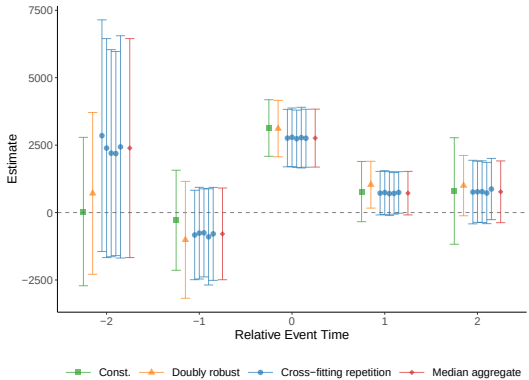
Recommendation 2 (Repeat and aggregate)

Repeating cross-fitting and (median-)aggregation reduced dependence on the random sample split. Letting $\widehat{\theta}_s$ and $\widehat{s.e.}_s$ denote the sth DML estimator and its standard error among a total of S replications, the median-aggregated DML estimator and standard errors are given by

$$\widehat{\theta}^{\text{median}} \equiv \text{median}\left(\{\widehat{\theta}_s\}_{s=1}^{S}\right), \qquad \widehat{s.e.}^{\text{median}} \equiv \sqrt{\text{median}\left(\{\widehat{s.e.}_s^2 + (\widehat{\theta}_s - \widehat{\theta}^{\text{median}})^2\}_{s=1}^{S}\right)}.$$

DML and Staggered Adoption DiD*

Figure: Dynamic Average Treatment Effect Estimates



Notes. The figure displays dynamic treatment effect estimates using different GT-ATT estimators. "Const." and "Doubly robust" refer to the Sant'Anna and Zhao (2020) estimator without and with controls, respectively. "Cross-fitting repetition" shows DML GT-ATT estimates across random cross-fitting folds, and "Median aggregate" presents the corresponding median aggregated DML estimate. Bars indicate point-wise 95% confidence intervals.

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.

The choice of machine learner

So far, we have taken the choice of the machine learners as given. ML toolbox includes a wide set of methods — but ***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?
  → "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.

Monopsony in the labor market

Because there are no obvious choices of nuisance estimators for the inclusion of textual controls, we consider several ***learner specifications***:

- ▷ OLS
- ▷ cross-validated lasso and ridge
- ▷ three types of random forest
- ▷ three types of `XGBoost`,
- ▷ three types of feed-forward neural networks

Controls: We consider BERT embeddings (Devlin et al., 2019), as well as the original hand-coded controls, resulting in a total of 24 distinct learner specifications.

Monopsony in the labor market

	Dependent variable: log duration							
Panel A.	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Estimate	0.1526	0.1485	0.1463	0.1136	0.1162	0.1171	0.0287	0.0237
	(0.0838)	(0.0634)	(0.0647)	(0.0041)	(0.0040)	(0.0040)	(0.0040)	(0.0036)
Controls	BERT	BERT	BERT	BERT	BERT	BERT	BERT	BERT
ML	OLS	CV-Lasso	CV-Ridge	RF 1	RF 2	RF 3	XGB 1	XGB 2
CV R^2: Outcome	-0.0440	-0.0329	-0.0318	-0.0227	0.0287	0.0259	0.5905	0.6077
CV R^2: Treatment	0.0173	0.0216	0.0199	-0.0288	0.0236	0.0212	0.4144	0.4332
CVC p-value: Outcome	0.0912	0.0872	0.0760	0.0316	0.0340	0.0356	0.1208	0.1548
CVC p-value: Treatment	0.0048	0.0000	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000
Weight: Outcome	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0673	0.0080
Weight: Treatment	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Notes: The table reports DML estimates based on 9 different nuisance estimators that are combined with either BERT controls or the original controls of Dube et al. (2020). We use $K = 2$ cross-fitting folds and employ median aggregation over the $R = 5$ cross-fitting repetitions. The number of observations is $N = 258,352$. The point estimate (standard error) reported by Dube et al. (2020, Table 1, col. 7) is -0.0299 ($s.e. = 0.00402$). We consider the following nuisance estimators: OLS; CV-lasso; CV-ridge; three types of random forest labeled RF 1 (600 trees), RF 2 (600 trees, minimum terminal node size of 500) and RF 3 (600 trees, minimum terminal node size of 2000); three types of `XGBoost` labeled XGB 1 (800 trees, early stopping after 10 rounds), XGB 2 (800 trees, minimum node size of 500) and XGB 3 (800 trees, minimum node size of 2000). DML estimation uses the R package `ddml` (Wiemann et al., 2023). The nuisance estimators were implemented with `glmnet` (Friedman et al., 2010), `ranger` (Wright and Ziegler, 2017) and `XGBoost` (Chen and Guestrin, 2016). $^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Monopsony in the labor market

	Dependent variable: log duration							
Panel B.	(9)	(10)	(11)	(12)	(13)	(14)	(15)	(16)
Estimate	0.0173	0.0875	0.0800	0.0814	-0.3881	-0.3880	-0.3873	-0.1449
	(0.0035)	(0.0463)	(0.0428)	(0.0620)	(0.0578)	(0.0570)	(0.0499)	(0.0049)
Controls	BERT	BERT	BERT	BERT	Orig.	Orig.	Orig.	Orig.
ML	XGB 3	NN 1	NN 2	NN 3	OLS	CV-Lasso	CV-Ridge	RF 1
CV R^2: Outcome	0.6128	0.0334	-0.0306	-0.0644	0.3367	0.3259	0.3266	0.1912
CV R^2: Treatment	0.4398	0.0547	0.1102	0.0971	0.5707	0.5712	0.5678	0.2142
CVC p-value: Outcome	0.1732	0.0952	0.0816	0.1304	0.1088	0.1084	0.0912	0.1268
CVC p-value: Treatment	0.0000	0.0000	0.0000	0.0000	0.0304	0.0228	0.0000	0.0008
Weight: Outcome	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Weight: Treatment	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Notes: The table reports DML estimates based on 9 different nuisance estimators that are combined with either BERT controls or the original controls of Dube et al. (2020). We use $K = 2$ cross-fitting folds and employ median aggregation over the $R = 5$ cross-fitting repetitions. The number of observations is $N = 258,352$. The point estimate (standard error) reported by Dube et al. (2020, Table 1, col. 7) is -0.0299 ($s.e. = 0.00402$). We consider the following nuisance estimators: OLS; CV-lasso; CV-ridge; three types of random forest labeled RF 1 (600 trees), RF 2 (600 trees, minimum terminal node size of 500) and RF 3 (600 trees, minimum terminal node size of 2000); three types of `XGBoost` labeled XGB 1 (800 trees, early stopping after 10 rounds), XGB 2 (800 trees, minimum node size of 500) and XGB 3 (800 trees, minimum node size of 2000). DML estimation uses the R package `ddml` (Wiemann et al., 2023). The nuisance estimators were implemented with `glmnet` (Friedman et al., 2010), `ranger` (Wright and Ziegler, 2017) and `XGBoost` (Chen and Guestrin, 2016). $^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Monopsony in the labor market

	Dependent variable: log duration							
Panel C.	(17)	(18)	(19)	(20)	(21)	(22)	(23)	(24)
Estimate	-0.1078	-0.0930	-0.0224	-0.0160	-0.0160	-0.3683	-0.3142	-0.3419
	(0.0051)	(0.0046)	(0.0058)	(0.0056)	(0.0052)	(0.0828)	(0.0797)	(0.0797)
Controls	Orig.	Orig.	Orig.	Orig.	Orig.	Orig.	Orig.	Orig.
ML	RF 2	RF 3	XGB 1	XGB 2	XGB 3	NN 1	NN 2	NN 3
CV R^2: Outcome	0.1669	0.1574	0.6768	0.6789	0.6775	0.3268	0.3383	0.3458
CV R^2: Treatment	0.1843	0.1727	0.7485	0.7252	0.7071	0.6135	0.6441	0.6520
CVC p-value: Outcome	0.2380	0.2296	0.4928	0.5196	0.4700	0.2536	0.2196	0.1688
CVC p-value: Treatment	0.0212	0.0204	0.8248	0.0876	0.0256	0.0660	0.0708	0.0764
Weight: Outcome	0.0000	0.0000	0.3771	0.2725	0.2334	0.0000	0.0178	0.0216
Weight: Treatment	0.0000	0.0000	0.5403	0.1950	0.0216	0.0000	0.0810	0.1328

Notes: The table reports DML estimates based on 9 different nuisance estimators that are combined with either BERT controls or the original controls of Dube et al. (2020). We use $K = 2$ cross-fitting folds and employ median aggregation over the $R = 5$ cross-fitting repetitions. The number of observations is $N = 258,352$. The point estimate (standard error) reported by Dube et al. (2020, Table 1, col. 7) is -0.0299 ($s.e. = 0.00402$). We consider the following nuisance estimators: OLS; CV-lasso; CV-ridge; three types of random forest labeled RF 1 (600 trees), RF 2 (600 trees, minimum terminal node size of 500) and RF 3 (600 trees, minimum terminal node size of 2000); three types of `XGBoost` labeled XGB 1 (800 trees, early stopping after 10 rounds), XGB 2 (800 trees, minimum node size of 500) and XGB 3 (800 trees, minimum node size of 2000). DML estimation uses the R package `ddml` (Wiemann et al., 2023). The nuisance estimators were implemented with `glmnet` (Friedman et al., 2010), `ranger` (Wright and Ziegler, 2017) and `XGBoost` (Chen and Guestrin, 2016). $^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Selecting/combining learners

Cross-fitted loss metric (e.g., R^2)

- ▷ Compute cross-fitted R^2 for each learner; select the learner with the highest R^2 for each equation.
- ▷ Caveat: Small R^2 differences can alter DML estimates (e.g., XGBoost vs. BERT embeddings).

Cross-Validation with Confidence (CVC) Test

- ▷ Test null: a given learner has lowest predictive risk among all candidates.
- ▷ Build confidence set of learners whose p-value exceeds a significance level (e.g., $\alpha = 0.2$).

(Short-)Stacking a.k.a. Model Averaging

- ▷ Combine all candidate learners via weighted average of cross-fitted predictions.
- ▷ Weights estimated (e.g., constrained least squares) to minimize combined risk.
- ▷ Ahrens et al. (2025) discuss several approaches.

Practical implementation

Recommendation 3 (Validating the nuisance function estimator)

We recommend researchers to carefully choose and tune their nuisance function estimators and consider a rich set of estimation approaches, including parametric benchmarks such as linear regression. DML estimates are only as credible as the choice of nuisance estimator.

Remarks:

- ▷ Choosing the nuisance function estimator is about the "match-quality" between estimator and (unknown) data-generating process.
- ▷ The optimal learner for $E[Y|X]$ might not be the same as for $E[D|X]$ — DML allows relaxing an assumption implicit in many estimation approaches.

Concluding remarks

- DML does not tell a researcher what interesting target parameters are or how to identify them. Rather, it complements careful reasoning about objects of interest and their identification.
- While straightforward in the abstract, implementing cross-fitting under dependence may pose practical difficulties; more research required.
- What if rate requirement fails?
  - van der Laan et al. (2025) apply a correction to the nuisance function estimators to enforce empirical orthogonality. The resulting estimator is double robust in the sense that it only requires one nuisance function estimator to satisfy the convergence rate requirement.

References I

Ahrens, A., Hansen, C. B., Schaffer, M. E., and Wiemann, T. (2024). ddml: Double/debiased machine learning in stata. *The Stata Journal*, 24(1):3–45.

Ahrens, A., Hansen, C. B., Schaffer, M. E., and Wiemann, T. (2025). Model averaging and double machine learning. *Journal of Applied Econometrics*, 40(3):249–269.

Angrist, J. D., Imbens, G. W., and Krueger, A. B. (1999). Jackknife instrumental variables estimation. *Journal of Applied Econometrics*, 14(1):57–67.

Angrist, J. D. and Krueger, A. B. (1995). Split-sample instrumental variables estimates of the return to schooling. *Journal of Business & Economic Statistics*, 13(2):225–235.

Bach, P., Chernozhukov, V., Kurz, M. S., and Spindler, M. (2021). DoubleML – An object-oriented implementation of double machine learning in R. arXiv:2103.09603 [stat.ML].

References II

Bach, P., Chernozhukov, V., Kurz, M. S., and Spindler, M. (2022). DoubleML – An object-oriented implementation of double machine learning in Python. *Journal of Machine Learning Research*, 23(53):1–6.

Bekker, P. A. (1994). Alternative approximations to the distributions of instrumental variables estimators. *Econometrica*, 63:657–681.

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.

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.

Bickel, P. J. (1982). On adaptive estimation. *Annals of Statistics*, 10:647–671.

References III

Callaway, B. and Sant'Anna, P. H. C. (2021). Difference-in-differences with multiple time periods. *Journal of Econometrics*, 225:200–230.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA. ACM.

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. *Econometrics Journal*, 21(1):C1–C68. tex.ids= Chernozhukov2018a publisher: John Wiley & Sons, Ltd (10.1111).

de Chaisemartin, C. and d'Haultfoeuille, X. (2023). Two-way fixed effects and differences-in-differences with heterogeneous treatment effects: a survey. *Econometrics Journal*, 26(3):C1–C30.

References IV

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint 1810.04805*.

Dobkin, C., Finkelstein, A., Kluender, R., and Notowidigdo, M. J. (2018). The economic consequences of hospital admissions. *American Economic Review*, 108(2):308–352.

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

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22.

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

Newey, W. K. (1994). The asymptotic variance of semiparametric estimators. *Econometrica*, 62(6):1349–1382.

References V

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.

Robinson, P. M. (1988a). Root-*N*-consistent semiparametric regression. *Econometrica*, 56(4):931–954.

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

Roth, J., Sant'Anna, P. H., Bilinski, A., and Poe, J. (2023). What's trending in difference-in-differences? a synthesis of the recent econometrics literature. *Journal of Econometrics*, 235(2):2218–2244.

Sant'Anna, P. H. C. and Zhao, J. (2020). Doubly robust difference-in-differences estimators. *Journal of Econometrics*, 219(1):101–122.

References VI

Sun, L. and Abraham, S. (2021). Estimating dynamic treatment effects in event studies with heterogeneous treatment effects. *Journal of Econometrics*, 225(2):175–199.

van der Laan, L., Luedtke, A., and Carone, M. (2025). Doubly robust inference via calibration. *arXiv preprint 2411.02771*.

Velez, A. (2024). On the asymptotic properties of debiased machine learning estimators. *arXiv preprint arXiv: 2411.01864*.

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.

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

References VII

Wright, M. N. and Ziegler, A. (2017). `ranger`: A fast implementation of random forests for high dimensional data in C++ and R. *Journal of Statistical Software*, 77(1):1–17.