Applied Causal Inference Powered by ML and AI

Victor Chernozhukov* Christian Hansen† Nathan Kallus‡
Martin Spindler§ Vasilis Syrgkanis¶

July 28, 2024

Publisher: Online
Version 0.1.1

* MIT
† Chicago Booth
‡ Cornell University
§ Hamburg University
¶ Stanford University

Statistical Inference on Predictive and Causal Effects in Modern Nonlinear Regression Models

10

"Whoever has participated in non-trivial research in any domain of science involving statistical problems must have encountered the difficulty that none of the statistical procedures found in the books fits exactly the practical situation."

– Jerzy Neyman [1].

Here we discuss double/debiased machine learning (DML) methods for performing inference on average predictive or causal effects in two important classes of models: partially linear regression models and interactive regression models. We also present a general DML method for performing inference on a low-dimensional target parameter in the presence of high-dimensional nuisance parameters. Two case studies illustrate the approach.

10.1 Introduction

We recall the predictive effect question:

- How does the predicted value of the outcome,

$$\mathrm{E}[Y \mid D, X],$$

  change if a regressor value D increases by a unit, while regressor values X remain unchanged?

This question may have a causal interpretation within any SEM, where conditioning on X is sufficient for identification of the causal effect of D on Y – that is, in any situation where unconfoundedness holds.[1] When this condition holds, the question becomes the causal effect question:

1: Recall that *unconfoundedness* or *conditional exogeneity* is covered in Chapter 5-Chapter 8.

- How does the predicted value of the potential outcome,

$$\mathrm{E}[Y(d) \mid X],$$

  change if we intervene and change the treatment value d by a unit, conditional on the observed X?

Both questions are interesting and useful to ask, depending on the application. In what follows, we set up double/debiased machine learning (DML) methods for answering these questions with data.[2] These statistical inference methods *do not* distinguish between the two types of questions, so the methods are equally applicable to answering both types.

2: In the book we will use the terms double/debiased machine learning, double machine learning and debiased machine learning interchangeably. It generalizes the double/debiased Lasso approach to generic machine learning methods.

Here we discuss DML methods for performing inference on average predictive or causal effects in two important classes of nonlinear regression models. After presenting these two special cases, we also present a general DML method for performing inference on a low-dimensional target parameter in the presence of high-dimensional nuisance parameters.

Current arguments used to formally establish $\sqrt{n}$ asymptotic normality of DML estimators of target parameters while allowing for the use of general machine learning methods for learning nuisance parameters make use of two key ingredients. First, the DML method is based on a *Neyman orthogonal representation* of the target parameters. Intuitively, Neyman orthogonality is the requirement that estimates of the parameters of interest are locally insensitive to the value of nuisance parameters. This local insensitivity reduces the spillover of regularization biases that are inherent in using regularized methods, such as the machine learning methods discussed in Chapter 9, to learn nuisance parameters onto the estimation of target parameters.

Second, DML makes use of *cross-fitting*: an efficient form of sample splitting that guards against "own-observation biases" that may arise from overfitting.

To illustrate the general principles, we provide two case studies. In the first, we perform inference on the effect of gun ownership on homicide rates. In the second, we perform inference on the effect of 401(k) eligibility on financial assets.

10.2 DML Inference in the Partially Linear Regression Model

We first answer the predictive/causal effect question within the context of the *partially linear regression model* (PLM):

$$Y = \beta D + g(X) + \epsilon, \quad \mathrm{E}[\epsilon \mid D, X] = 0, \tag{10.2.1}$$

where Y is the outcome variable, D is the regressor of interest, and X is a high-dimensional vector of other regressors or features, called "controls." The coefficient β answers the predictive effect question. In this section, we discuss estimation and construction of confidence intervals for β. We also provide a case study in which we examine the effect of gun ownership on homicide rates.

The PLM allows a part of the regression function, $g(X)$, to be fully nonlinear, which generalizes the approach from Chapter 4. However, the model is still not fully general, because it imposes additivity in $g(X)$ and D. We shall consider a fully unrestricted model in the case of a binary treatment D in Section 10.3. It is worth pointing out before turning to that setting that the PLM is not as restrictive as it appears since we can consider explicit interactions within the partially linear framework.

Remark 10.2.1 (Interactions within PLM) Given a raw treatment and a set of controls, $\bar{D}$ and Z, we can create the technical treatment $D := \bar{D}T(Z)$, where $T(Z)$ is an $L-$dimensional dictionary of transformations of Z. For example, $T(Z)$ could be indicators of various subgroups. We can then consider the model

$$Y = \sum_{l=1}^{L} \beta_l D_l + g(Z) + \epsilon,$$

where $\mathrm{E}[\epsilon \mid Z, D] = 0$. We can rewrite this model as

$$Y = \beta_l D_l + g_l(X_l) + \epsilon, \quad \mathrm{E}[\epsilon \mid D_l, X_l] = 0,$$

where $g_l(X_l) := \sum_{k \neq l} \beta_k D_k + g(Z)$ and $X_l := ((D_k)_{k \neq l}, Z)$. We therefore obtain exactly a model of the partially linear form (10.2.1). We can then apply DML methods to learn and perform inference on each element of $(\beta_l)_{l=1}^L$ or carry out joint inference (similarly to what we have done in Chapter 4).

In practice and depending on the learner, it may be convenient to treat $g_l(X_l) = h(\{D_k\}_{k \neq l}, Z)$ as a flexible function during estimation rather than impose the structure $g_l(X_l) := \sum_{k \neq l} \beta_k D_k + g(Z)$.

In what follows, we employ an operation that "partials-out" X from a random variable V by taking V as an input and returning the "residualized" form:

$$\tilde{V} := V - \mathrm{E}[V \mid X].$$

Applying this operation to (10.2.1), we obtain

$$\tilde{Y} = \beta \tilde{D} + \epsilon, \quad \mathrm{E}[\epsilon \tilde{D}] = 0, \tag{10.2.2}$$

where $\tilde{Y}$ and $\tilde{D}$ are the residuals left after predicting Y and D using X. Specifically, we have that

$$\tilde{Y} := Y - \ell(X) \text{ and } \tilde{D} := D - m(X),$$

where $\ell(X)$ and $m(X)$ are defined as conditional expectations of Y and D given X:

$$\ell(X) := \mathrm{E}[Y \mid X] \text{ and } m(X) := \mathrm{E}[D \mid X].$$

Here we recall that the conditional expectations of Y and D given X are the best predictors of Y and D using X under squared error loss.

The equation $\mathrm{E}[\epsilon \tilde{D}] = 0$ above is the Normal Equation for the population regression of $\tilde{Y}$ on $\tilde{D}$. This equation implies the following result:

Theorem 10.2.1 (FWL Partialling-Out for Partially Linear Model) *Suppose that Y, X, and D have bounded second moments. Then the population regression coefficient β can be recovered from the population linear regression of $\tilde{Y}$ on $\tilde{D}$:*

$$\beta := \{b : \mathrm{E}\left[(\tilde{Y} - b\tilde{D})\tilde{D}\right] = 0\} = (\mathrm{E}[\tilde{D}^2])^{-1}\mathrm{E}[\tilde{D}\tilde{Y}], \tag{10.2.3}$$

where the second equality and unique definition of β follow if D cannot be perfectly predicted by X, i.e. if $\mathrm{E}[\tilde{D}^2] > 0$.

Thus, β can be interpreted as a regression coefficient of *residualized* Y on *residualized* D, where the residuals are defined by respectively subtracting the conditional expectation of Y given X and D given X from Y and D. This result generalizes the FWL from linear models to partially linear models.

Our estimation procedure for β in the sample will mimic the partialling out procedure in the population. We also rely on cross-fitting (outlined below) to make sure any overfitting in learning the conditional expectation functions used in constructing the residualized quantity does not spillover to contaminate the final estimator of the quantity of interest. [3]

Double/Orthogonal ML for the Partially Linear Model

1. Partition data indices into random folds of approximately equal size: $\{1, ..., n\} = \cup_{k=1}^{K} I_k$. For each fold $k = 1, ..., K$, compute ML estimators $\hat{\ell}_{[k]}$ and $\hat{m}_{[k]}$ of the conditional expectation functions ℓ and m, leaving out the k-th block of data. Obtain the cross-fitted residuals for each $i \in I_k$:

$$\check{Y}_i = Y_i - \hat{\ell}_{[k]}(X_i), \quad \check{D}_i = D_i - \hat{m}_{[k]}(X_i).$$

2. Apply ordinary least squares of $\check{Y}_i$ on $\check{D}_i$. That is, obtain $\hat{\beta}$ as the root in b of the normal equations:

$$\mathbb{E}_n[(\check{Y} - b\check{D})\check{D}] = 0.$$

3. Construct standard errors and confidence intervals as in standard least squares theory.

3: Step 1 of the **Double/Orthogonal ML for the Partially Linear Model** algorithm is the *cross-fitting* step. Here, the nuisance parameters - the conditional expectation functions in this example - are learned from one set of observations and then applied to a different set of observations to construct the residualized quantities. Heuristically, this step keeps any overfitting that occurred in estimating the conditional expectation from feeding through and contaminating the final estimate of the parameter of interest. See Section 10.4 for more discussion.

In what follows it will be convenient to use the notation

$$\|h\|_{L^2} := \sqrt{\mathrm{E}_X[h^2(X)]},$$

where, as before, E_X computes the expectation over values of X.

Theorem 10.2.2 (Adaptive Inference on a Target Parameter in PLM [2]) *Consider the PLM model. Suppose that estimators* $\hat{\ell}_{[k]}(X)$ *and* $\hat{m}_{[k]}(X)$ *provide approximations to the best predictors* $\ell(X)$ *and* $m(X)$ *that are of sufficiently high-quality:*

$$n^{1/4}(\|\hat{\ell}_{[k]} - \ell\|_{L^2} + \|\hat{m}_{[k]} - m\|_{L^2}) \approx 0.$$

Suppose that $\mathrm{E}[\tilde{D}^2]$ *is bounded away from zero; that is, suppose* $\tilde{D}$ *has non-trival variation left after partialling out. Suppose other regularity conditions listed in [2] hold.*

Then the estimation error in $\check{D}_i$ *and* $\check{Y}_i$ *has no first order effect on* $\hat{\beta}$*:*

$$\sqrt{n}(\hat{\beta} - \beta) \approx (\mathbb{E}_n[\tilde{D}^2])^{-1}\sqrt{n}\mathbb{E}_n[\tilde{D}\epsilon].$$

Consequently, $\hat{\beta}$ *concentrates in a* $1/\sqrt{n}$ *neighborhood of* β *with deviations approximated by the Gaussian law:*

$$\sqrt{n}(\hat{\beta} - \beta) \overset{\text{a}}{\sim} N(0, \mathsf{V}),$$

where

$$\mathsf{V} = (\mathrm{E}[\tilde{D}^2])^{-1}\mathrm{E}[\tilde{D}^2\epsilon^2](\mathrm{E}[\tilde{D}^2])^{-1}.$$

Remark 10.2.2 (When PLM fails to hold) Even when the PLM model fails to hold, Theorem 10.2.2 continues to hold when we directly define β as in Eq. 10.2.3 of Theorem 10.2.1 for any variable triplet (X, D, Y). That is, $\hat{\beta}$ is in fact an estimate of the BLP of $\tilde{Y}$ in terms of $\tilde{D}$ regardless of whether the PLM holds. Per Theorem 10.2.1, this BLP coefficient coincides with β in Eq. (10.2.1) whenever the PLM does hold.

Confidence Interval The standard error of $\hat{\beta}$ is $\sqrt{\hat{\mathsf{V}}/n}$, where $\hat{\mathsf{V}}$ is an estimator of V. The result implies that the confidence interval

$$\left[\hat{\beta} - 2\sqrt{\hat{\mathsf{V}}/n}, \hat{\beta} + 2\sqrt{\hat{\mathsf{V}}/n}\right]$$

covers β in approximately 95% of possible realizations of the sample. In other words, if our sample is not atypical, the interval covers the truth.

Selecting the Best ML Learners of ℓ and m. There may be several methods that satisfy the quality requirements of Theorem 10.2.2, and we may therefore ask what ML methods we should use in practice. Consider a collection of ML methods indexed by $j \in \{1, \ldots, J\}$. Our goal would be to select the methods that minimize an upper bound on the bias of the DML estimator.

The bias of the DML estimator is controlled by the mean square approximation errors (MSAE):

$$\frac{1}{K}\sum_{k=1}^{K} \|\hat{\ell}_{[k]} - \ell\|_{L^2}^2 \text{ and } \frac{1}{K}\sum_{k=1}^{K} \|\hat{m}_{[k]} - m\|_{L^2}^2. \tag{10.2.4}$$

Therefore, we can select the best ML method for estimating m and the best method for estimating ℓ to minimize the upper

bound on the bias. We will be using mean square prediction errors as proxies for MSAEs.

MSPEs approximate MSAEs up to terms that do not depend on j. Hence, by doing MSPE minimization, we in fact approximately minimize MSAEs.

Selection of the Best ML Methods for DML to Minimize Bias. Consider a set of ML methods enumerated by $j \in \{1, ..., J\}$.

- For each method j, compute the cross-fitted MSPEs

$$\mathbb{E}_n[\check{Y}_j^2] \text{ and } \mathbb{E}_n[\check{D}_j^2],$$

  where the index j reflects the dependency of residuals on the method.

- Select the ML methods $j \in \{1, ..., J\}$ that give the smallest MSPEs:

$$\hat{j}_\ell = \arg\min_j \mathbb{E}_n[\check{Y}_j^2] \text{ and } \hat{j}_m = \arg\min_j \mathbb{E}_n[\check{D}_j^2].$$

- Use the method $\hat{j}_\ell$ as a learner of ℓ, and $\hat{j}_m$ as a learner of m in the DML algorithm above.

Note that it may well be that different methods provide the best prediction rules for Y and D. By allowing ourselves to consider multiple methods, we allow finding methods that perform best for the different tasks which should improve performance in practice relative to insisting on one fixed, pre-specified method.

Rather than selecting the single best predictors of Y and D, we can also use residuals to form linear ensembles of ML methods that minimize MSPEs.

Corollary 10.2.3 *The previous inferential result continues to hold if the best or aggregated prediction rules are used as estimators $\hat{m}$ and $\hat{\ell}$ of m and ℓ in the DML algorithm. A simple sufficient condition is that the number of ML prediction rules J over which we aggregate or choose from is fixed (meaning small in practice).*

In practical terms, the result of Corollary 10.2.3 means that we should only choose among or aggregate over relatively few ML methods. Otherwise, we may end up overfitting (since we are "cheating" here by using validation data to form the aggregator).

Remark 10.2.3 (More Technical Condition) A sufficient condition for data dependent selection of which predictor to use when forming residuals to perform well in theory often boils down to requiring $\sqrt{\log J} n^{-1/4} \approx 0$ for choosing the single best method and $\sqrt{J} n^{-1/4} \approx 0$ when using the linear aggregation of methods. However, much work in this area is yet to be formally developed.

Discussion of DML Construction

The partialling out operation causes the moment equations defining β to be Neyman orthogonal. That is, the moment conditions are locally insensitive to perturbations of the nuisance parameters ℓ and m.[4] We discussed Neyman orthogonality in the context of high-dimensional linear regression models in Chapter 4. We return to and generalize this discussion formally in Section 10.4. This property alleviates the impact of the bias in estimation of m and ℓ that arises when ML estimators are applied in high-dimensional settings.

4: Recall that we use the term nuisance parameter to name parameters that are not the target parameters. In the PLM, the target parameter is β, and ℓ and m are nuisance parameters.

Naive application of machine learning methods directly to outcome equations may lead to highly biased estimators because the resulting strategy is not Neyman orthogonal. The lack of Neyman orthogonality means that estimates of the parameter of interest are heavily impacted by estimation of the nuisance parameters. This sensitivity means that any biases in estimation of g, which are essentially unavoidable in high-dimensional estimation, create a non-trivial bias in the estimate of the main effect. This bias is large enough to cause failure of conventional inference.

The left panel of Figure 10.1 illustrates the bias arising due to the use of a non-orthogonal, naive approach for learning β; see Remark 10.2.4 for details. Specifically, the figure shows the behavior of a conventional (non-orthogonal) ML estimator, $\tilde{\beta}$, in the partially linear model in a simple simulation experiment where we learn g using a random forest. The g in this experiment is a very smooth function of a small number of variables, so the experiment is seemingly favorable to the use of random forests a priori. The histogram shows the simulated distribution of the centered estimator, $\tilde{\beta} - \beta$. The estimator is badly biased, shifted much to the right relative to the true value β. Furthermore, the distribution of the estimator (approximated by the blue histogram) is substantively different from a normal approximation (shown by the red curve) derived under the assumption that the bias is negligible.

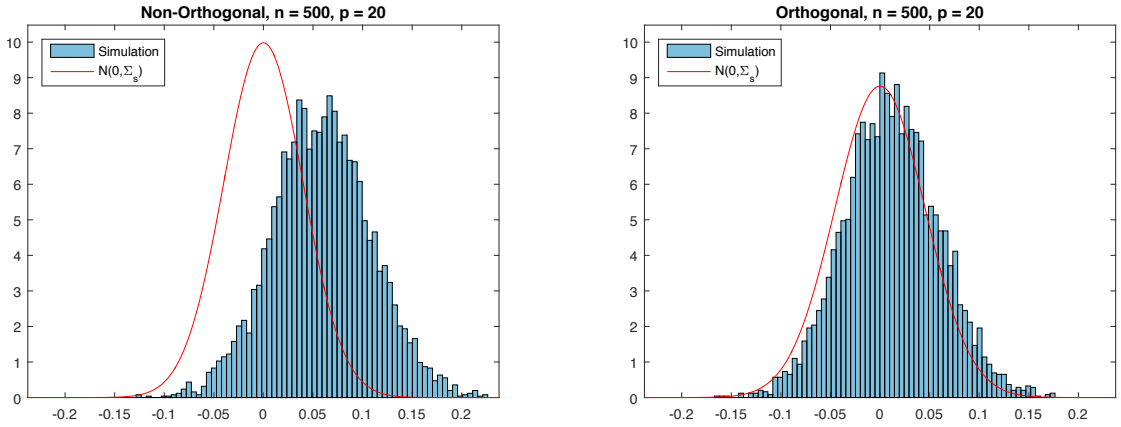

Figure 10.1: Histograms of centered estimates of β from a simulation experiment where random forests are used to learn nuisance functions. The left panel provides the histogram of $\widetilde{\beta} - \beta$ where $\widetilde{\beta}$ is obtained from a procedure that does not satisfy Neyman orthgonality; see Remark 10.2.4. The right panel provides the histogram of $\widehat{\beta} - \beta$ where $\widehat{\beta}$ is the DML estimator. In both cases, the same tuning settings are used for random forest estimation of nuisance parameters.

Remark 10.2.4 (Bias Transmission) The biased performance of the naive estimator can be explained analytically. The naive strategy relies on the moment equation:

$$\mathrm{E}[(Y - \beta D - g(X))D] = 0$$

to identify β. Because we do not know g, we need to use an estimate of g in place of g. Without ex ante knowing the specific form of g, this estimate of g will suffer from bias in finite samples. Unfortunately, the moment condition $\mathrm{E}[(Y - \beta D - g(X))D] = 0$ is sensitive to deviations in g away from the true value. Indeed, let us compute the directional derivative in direction Δ away from the true value:

$$\partial_t \mathrm{E}[(Y - \beta D - g(X) + t\Delta(X))D]\Big|_{t=0} = \mathrm{E}[\Delta(X)D] \neq 0.$$

The derivative generally does not vanish. The result is bias in $\hat{g}$, even if relatively small, will transmit to the resulting estimator of β.

The right panel of Figure 10.1 illustrates the behavior of the (Neyman) orthogonal DML estimator, $\hat{\beta}$, in the partially linear model in a simple experiment where we learn nuisance functions m and ℓ using random forests. Note that the simulated data are exactly the same as those underlying the left panel. The simulated distribution of the centered estimator, $\hat{\beta} - \beta$, (given by the blue histogram) illustrates that the estimator is approximately unbiased, concentrates around β, and is approximately normally distributed. The low bias arises because DML uses

Neyman orthogonal moment equations.

The DML algorithm also uses a form of sample splitting, called cross-fitting, to guard against a less obvious source of bias that may arise when estimation of nuisance parameters results in overfitting. Heuristically, overfitting simply means that an estimator has captured not just generalizable signal but also noise that is idiosyncratic to each observation. The presence of this idiosyncratic noise in the estimates of the nuisance functions may then lead to a type endogeneity bias as the observed estimates of the nuisance functions which are used in place of the unobserved, true nuisance functions are associated with the noise in the observations used to learn the nuisance functions. Cross-fitting guards against this source of bias as overfitting resulting from learning nuisance functions in one subsample will not carry over when the nuisance function estimates are applied on a different, separate subsample. As it is very hard to ensure that highly complex fitting methods such as boosting, deep neural networks, and random forests do not overfit, it is hard to know that their use would not lead to substantial biases without making use of sample splitting. That is, if we don't do sample splitting and the ML estimates overfit, we may end up with very large biases.

Figure 10.2 illustrates how the bias resulting from overfitting in the estimation of nuisance functions can cause DML without sample splitting (i.e. estimation on the full sample using an estimator that satisfied Neyman orthogonality) to be biased and how sample splitting eliminates this problem. In the left panel the histogram shows the finite-sample distribution of the DML estimator in the partially linear model in a simple simulation experiment where nuisance parameters are estimated with overfitting using the full sample, i.e. without sample splitting. The finite-sample distribution is clearly shifted to the left of the true parameter value, demonstrating the substantial bias. In the right panel, the histogram shows the finite-sample distribution of the DML estimator in the same simulation experiment in the partially linear model where nuisance parameters are estimated with sample-splitting using the cross-fitting estimator. Here, we see that the use of sample-splitting has completely eliminated the bias induced by overfitting.

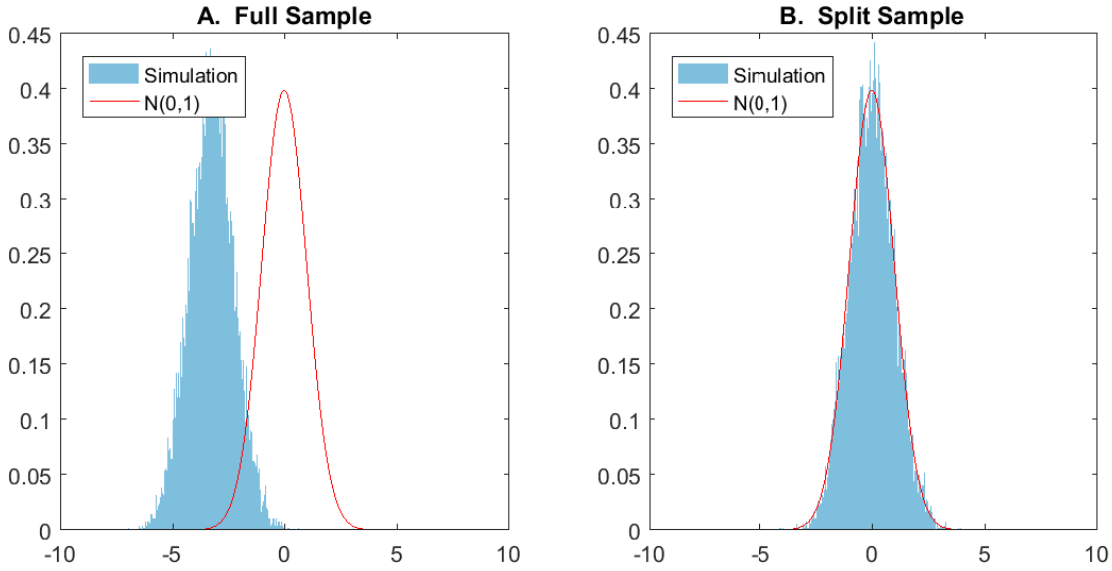

Figure 10.2: Left: DML distribution without sample-splitting. Right: DML distribution with cross-fitting.

Remark 10.2.5 (On overfitting) Note that we did not make use of cross-fitting in the context of doing inference for a regression coefficient in the high-dimensional linear model setting in Chapter 4. Importantly, we only made use of Lasso with the plug-in choice for the penalty level λ in that setting. The plug-in tuning choice *theoretically guarantees* that overfitting is sufficiently well-controlled that sample splitting is not required. Such refined, theoretically rigorous choices of tuning parameters are not yet available for other machine learning methods. Indeed, even when using Lasso, cross-fitting should be employed if cross-validation, rather than the theoretical plug-in, is used for selecting λ.

In practice, experienced researchers and machine learning engineers often use intuition, heuristics, and other empirical tools (six packs or witchcraft tables, for example) to set the tuning parameters. While the resulting methods can perform well for prediction purposes, even modest overfitting can result in large biases in DML, as illustrated in the simulation experiment in Figure 10.2. In reality, any data-driven tuning method, such as cross-validation, is likely to lead to at least mild overfitting as the same data is being used repeatedly. Therefore, it is simply safer to rely on sample-splitting in real settings, especially when using complicated learners, to make sure overfitting during estimation of our residualized quantities does not contaminate the estimates of the objects of interest.

Figure 10.3: Witchcraft tables used by some ML practitioners to tune parameters. There are no known theoretical guarantees attached to this tuning method.

The Effect of Gun Ownership on Gun-Homicide Rates

We consider the problem of estimating the effect of gun ownership on the homicide rate.[5] For this purpose, we estimate the partially linear model:

$$Y_{i,t} = \beta D_{i,(t-1)} + g(X_{i,t}, \bar{X}_i, \bar{X}_t, X_{i,0}, Y_{i,0}, t) + \epsilon_{i,t}.$$

$Y_{i,t}$ is the log homicide rate in county i at time t. $D_{i,t-1}$ is the log fraction of suicides committed with a firearm in county i at time $t-1$, which we use as a proxy for gun ownership, $G_{i,t}$, which is not observed. $X_{i,t}$ is a set of demographic and economic characteristics of county i at time t. We use $\bar{X}_i$ to denote the within county average of $X_{i,t}$ and $\bar{X}_t$ to denote the within time period average of $X_{i,t}$. $X_{i,0}$ and $Y_{i,0}$ denote initial conditions in county i. We use $Z_{i,t}$ to denote the set of observed control variables $\{X_{i,t}, \bar{X}_i, \bar{X}_t, X_{i,0}, Y_{i,0}, t\}$. The sample covers 195 large United States counties between the years 1980 through 1999, giving us 3900 observations.

Raw control variables $X_{i,t}$ are from the U.S. Census Bureau and contain demographic and economic characteristics of the counties such as features of the age distribution, the income distribution, crime rates, federal spending, home ownership rates, house prices, educational attainment, voting patterns, employment statistics, and migration rates.

The intent here is that parameter β is an approximation of the causal effect of gun ownership, $G_{i,t}$, on homicide rates $Y_{i,t}$, controlling for county-level demographic and economic characteristics; see Figure 10.4 for a potential DAG representation. To attempt to flexibly account for fixed heterogeneity across counties, common time factors, and deterministic time trends, we include county-level averages, time period averages, initial conditions, and the time index as additional control variables. This strategy is related to strategies for addressing latent sources of heterogeneity via conditioning as in [4]. Finally, for simplicity in this illustration, we assume that all sources of dependence are accounted for by observed variables such that we may take $\epsilon_{i,t}$ as independent across counties, i, and over time, t.

As a summary statistic we first look at a simple regression of $Y_{i,t}$ on $D_{i,t-1}$ without controls. The point estimate is 0.302 with 95% confidence interval, based on the assumption that $\epsilon_{i,t}$ is independent over time and space, ranging from 0.277 to 0.327. These results suggest that increases in gun ownership rates are associated with (predict) gun homicide rates – if gun

5: We adapt the basic strategy from Cook and Ludwig [3] who consider using suicide rates as a proxy for gun ownership.

Python Notebook on DML for Impact of Gun Ownership on Homicide Rates and R Notebook on DML for Impact of Gun Ownership on Homicide Rates provide code for the example.

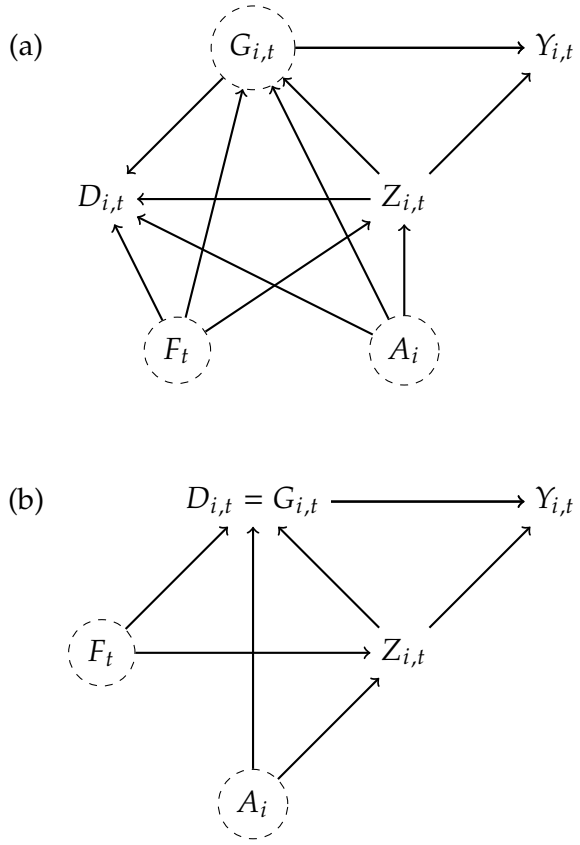

Figure 10.4: Possible DAG Structure for the Gun Ownership Example. Figure (a) provides a relatively general DAG structure that could represent the gun ownership example. We include nodes for latent county specific and time period specific shocks (A_i and F_t). Often such shocks are accounted for with so-called "fixed effects." In practice, estimating models with fixed effects are typically leverages strong functional form assumptions. Here, we instead leverage the different, though still strong, assumption that flexibly conditioning on observables, including time- and county-specific variables, is sufficient to account for latent county specific and time period specific shocks. Unfortunately, neither the causal effect of the unobserved $G_{i,t}$ or observed $D_{i,t}$ is identified assuming only structure (a). Figure (b) allows identification of the average causal effect $G_{i,t} \rightarrow Y_{i,t}$ by imposing $G_{i,t} = D_{i,t}$. To the extent that we believe $G_{i,t} \approx D_{i,t}$, the structure in (b) allows us to approximate the effect of interest. We discuss a further generalization in Section 10.A where we rely on the the assumption that $D_{i,t}$ is equal to $G_{i,t}$ plus an additive, independent measurement error. In this case, the target parameter β will be attenuated relative to the true causal effect.

ownership increases by 1% the predicted gun homicide rate goes up by around 0.3% – without controlling for any time factors or county characteristics.

Since our goal is to estimate the effect of gun ownership after controlling for a rich set characteristics, we next include the controls and estimate the model by an array of the modern regression methods that we've learned. Specifically, we consider ten candidate learners for predicting the outcome and for predicting the target variable. We consider linear models estimated with OLS using no control variables (OLS - No Controls), using only the raw control variables (OLS - Basic), and using the raw control variables plus the constructed cross-sectional and time series averages and initial conditions (OLS - All). The remaining methods always take as inputs the complete set of candidate control variables including cross-sectional averages, time-specific averages, and initial conditions. We use cross-

validation to choose tuning parameters for Lasso, Ridge, and Elastic Net. We consider a random forest with the software default tuning choices and boosted trees constrained to have depth four. Finally, we consider neural nets with four hidden layers of 50 nodes each and either dropout or early stopping. For further training details, refer to the example's notebooks.

	RMSE Y	RMSE D
OLS - No Controls	1.0964	1.2106
OLS - Basic	0.4889	0.1269
OLS - All	0.4259	0.1259
Lasso (CV)	0.4625	0.1353
Ridge (CV)	0.5303	0.1448
Elastic Net (CV)	0.4651	0.1339
Random Forest	0.4027	0.1246
Boosted trees - depth 4	0.4018	0.1223
DNN dropout	0.6142	0.7594
DNN early stopping	0.5352	0.1931

Table 10.1: Cross-fitted RMSE for predicting outcome (Y) and variable of interest (D) in the gun illustration.

.

Before turning to estimation results for β, we look at estimated out-of-sample predictive performance in Table 10.1 which reports cross-fitted root mean square error (RMSE) for the different procedures we consider. The column RMSE Y gives the RMSE for predicting the outcome (log gun homicide rate), and the column RMSE D gives the RMSE for predicting our gun prevalence variable (log of the lagged firearm suicide rate). Here we see evidence of the potential relevance of trying several learners rather than just relying on a single, pre-specified choice. There are noticeable differences between performance of most of the learners, with Boosted Trees and Random Forests providing the best performance for predicting both the outcome and policy variable.

Table 10.2 presents the estimated effects of the lagged gun ownership rate on the gun homicide rate as well as the corresponding standard errors. Looking across the results, we see relatively large differences in estimates. These differences suggest that the choice of learner has a material impact in this example. Looking at the measures of predictive performance in Table 10.1, we see that Random Forest and Boosted trees performed best among the considered learners, and we also see that their performance is relatively similar in terms of point estimates of the effect of the lagged gun ownership rate on the gun homicide rate and standard errors. Focusing on the Boosted trees row, the point estimate suggests a 1% increase in the gun proxy is associated

	Estimate	Standard Error
OLS - No Controls	0.3018	0.0126
OLS - Basic	0.3539	0.0738
OLS - All	0.2016	0.0570
Lasso (CV)	0.2758	0.0565
Ridge (CV)	0.4607	0.0600
Elastic Net (CV)	0.2944	0.0583
Random Forest	0.0631	0.0524
Boosted trees - depth 4	0.1025	0.0537
DNN dropout	0.2444	0.0125
DNN early stopping	0.4850	0.0492
Best	0.1025	0.0537
Ensemble	0.1079	0.0548

Table 10.2: Cross-fit estimates for the coefficient on our gun control proxy and standard errors in the gun illustration.

.

with around .1% increase in the gun homicide rate, though the 95% confidence interval is relatively wide: (-0.003,0.208).

The final two rows of Table 10.2 provide estimates based on using ensembles of the individual methods to estimate the nuisance functions. The row "Best" uses the method with the lowest MSE as the estimator for $\hat{\ell}(X)$ and $\hat{m}(X)$. In this example, Boosted trees give the best performances in predicting both $Y_{i,t}$ and $D_{i,t-1}$, so the results for "Best" and Boosted trees are identical. The row "Ensemble" uses the linear combination of all ten of the predictors the produces the lowest MSE for predicting $Y_{i,t}$ or $D_{i,t-1}$ as the estimator $\hat{\ell}(X)$ or $\hat{m}(X)$ respectively. Here the results are similar to the results using only Boosted trees, but differ somewhat due to non-zero linear combination coefficients on the other learners. In either case, we see mild evidence of positive effect of our gun ownership proxy on the gun homicide rate, though the 95% confidence interval includes 0 and small negative values in both cases.

We also wish to emphasize that this example helps illustrate the practical importance of considering several learners as it is generally *ex ante* unknown which learner will work best and there are substantial differences between the better performing learners (random forests, boosted trees, and the two ensemble methods) and the others both in terms of predictive accuracy and the results DML estimates and standard errors for the effect of interest.

Revisiting the Price Elasticity for Toy Cars

We now revisit again the example from Chapter 0. We are interested in the coefficient α in the PLM:

$$Y = \alpha D + g(W) + \epsilon,$$

where Y is log-reciprocal-sales-rank, D is log-price, and W are product features. In Chapter 4, we let $g(W) = \beta' T(W)$ be a high-dimensional regression using a transformation that included powers and interactions. We now employ flexible nonlinear regression models using DML. We take W to consist of indicators for brand and subcategory along with physical dimensions interacted with missingness indicators, using no further transformation, leading to a 2083-dimensional feature vector. We consider inference on α using DML with different choices of learners applied to both $m(W)$ and $g(W)$: decision trees, gradient boosted trees (with 1000 trees), random forests (with 2000 trees), or a neural network (with two hidden layers of 200 and 20 neurons, respectively, and ReLU activations).

This example makes use of proprietary data, so no notebooks are provided.

In Table 10.3, we report the cross-validated R^2 for predicting D and Y with each of the learners along with the resulting DML point estimate, standard error estimate, and 95% confidence interval. The first thing we note is that all confidence intervals indicate a substantial negative effect, with a clear indication not only of the direction of the effect but also of its overall magnitude.

Let us first compare these results to the previous ones from when we last revisited this example in Chapter 4. There we saw that OLS with varying number of features failed to exclude 0 from the confidence interval and that Double Lasso led to an interval [-0.099, -0.029]. We can attribute the latter more negative interval to controlling more for confounding, as we expect confounding effects to push the apparent price-sales relationship upward compared to the theorized downward causal relationship.

Here, we see that with more flexible nonlinear methods we obtain an even more negative estimate and confidence interval. This result appears to be consistent with the degree to which we are able to control for confounders. Lasso has a cross-validated R^2 of 0.09 and 0.32 for predicting Y and D, respectively. The R^2's in Table 10.3 are substantially larger. That the corresponding estimates and intervals are also more negative seems to coincide with our theoretical prediction.

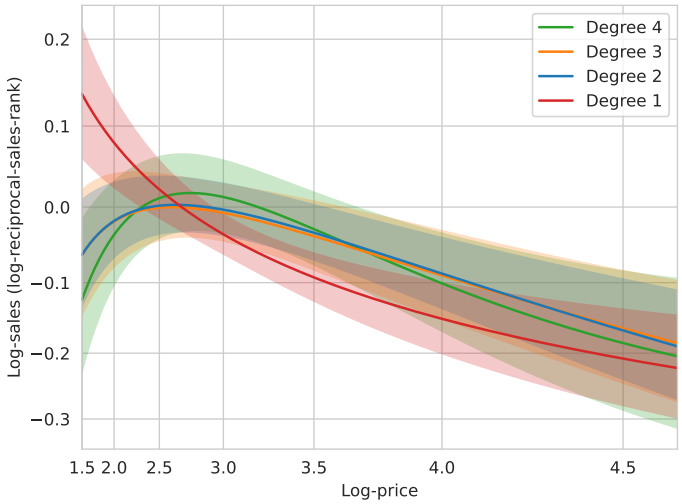

Figure 10.5: DML estimates of the price-sales relationship using PLM with higher-order transformations of price. Note the exponential scaling in the axes, which transforms the overall scale back to (non-log) price and sales (reciprocal sales rank).

Comparing between the nonlinear methods, this theory appears to remain consistent. Forest and neural net methods have higher R^2's than tree and gradient boosting methods, and, at the same time, have more negative point estimates and confidence intervals.

	R^2_D	R^2_Y	Estimate	Std. Err.	95% CI
Tree	0.40	0.19	-0.109	0.018	[-0.143, -0.074]
Boost	0.41	0.17	-0.102	0.019	[-0.139, -0.064]
Forest	0.49	0.26	-0.134	0.019	[-0.171, -0.096]
NNet	0.47	0.21	-0.132	0.020	[-0.171, -0.093]

Table 10.3: DML estimates of price elasticity based on different learners, along with their R^2 for predicting D and Y.

Note that just as we can play with transformations in linear models, we can do the same in the PLM. That is, we can modify from partial linearity in the univariate D to partial linearity in a multivariate set of transformations, $T(D)$. We can use this to investigate potentially non-linear price-sales relationships in this data. Let us transform D using the first r (probabilist's) Hermite polynomials (applied to a location-scale-standardized D). We then use DML with neural network learners to learn the coefficients on these polynomial terms. That is, we estimate a model of the form

$$Y = \sum_{j=1}^{r} \alpha_j T_j(D) + g(W) + \epsilon$$

where $T_j(D)$ represents the j^{th} term in the Hermite polynomial of order r.

We plot the resulting estimated functions for $r = 1, \ldots, 4$ in

Figure 10.5. As can be seen, the price-sales relationship seems to not be exactly linear, as it stabilizes around a flat-then-decreasing shape for degrees 2, 3, and 4. This shape either suggests that indeed there is less elasticity at lower price points (the mean log-price is 3.06) or that we simply failed to account well for confounding effects at lower price points, which may be idiosyncratic compared to higher-priced toy trucks.

The relationship being not exactly linear does not invalidate using a PLM (in the untransformed univariate D). It still corresponds to an average derivative – see Remark 10.3.3 – which can be more interpertable than nonlinear estimates of a causal effect.

10.3 DML Inference in the Interactive Regression Model

DML Inference on APEs and ATEs

We consider estimation of average treatment effects when treatment effects are fully heterogeneous and the treatment variable is binary. We consider observable variables $W = (Y, D, X)$ and the pair of regression equations:

$$Y = g_0(D, X) + \epsilon, \qquad \mathrm{E}[\epsilon \mid X, D] = 0, \tag{10.3.1}$$

$$D = m_0(X) + \tilde{D}, \qquad \mathrm{E}[\tilde{D} \mid X] = 0, \tag{10.3.2}$$

where the second regression equation captures that D and X are confounded. Here Y is an outcome of interest, $D \in \{0, 1\}$ is a binary policy or treatment variable, and X are controls/confounding factors. Since D is not additively separable in the first equation, this model is more general than the partially linear model for the case of binary D.

A common target parameter of interest in this model is the average predictive effect (APE),

$$\theta_0 = \mathrm{E}[g_0(1, X) - g_0(0, X)].$$

This quantity is the average predictive effect of switching $D = 0$ to $D = 1$. Under ignorability/conditional exogeneity, the APE coincides with the average treatment effect (ATE) of the intervention that moves $D = 0$ to $D = 1$.

The confounding factors X affect the policy variable via the propensity score $m_0(X)$ and the outcome variable via the function $g_0(D, X)$. Both of these functions are unknown (except for the case of RCTs, where $m_0(X)$ is known) and potentially complicated. Fortunately, we can employ ML methods to learn them.

Our construction of the efficient estimator for the APE/ATE will be based upon the relation[6]

$$\theta_0 = \mathrm{E}[\varphi_0(W)], \tag{10.3.3}$$

where

$$\varphi_0(W) = g_0(1, X) - g_0(0, X) + (Y - g_0(D, X))H_0$$

and

$$H_0 = \frac{1(D = 1)}{m_0(X)} - \frac{1(D = 0)}{1 - m_0(X)}$$

is the Horvitz-Thompson transformation.

6: This representation is known as "doubly robust" parameterization, which refers to the fact that θ_0 is recovered whenever the g or H is specified correctly. We don't dwell on this property here – for us, only the Neyman orthogonality property is important.

Remark 10.3.1 (Regression Adjustment or Propensity Score Reweighting? Use both) We realize that this representation encompasses two equally valid representations of the target parameter: the regression adjusted representation,

$$\theta_0 = \mathrm{E}[g_0(1, X) - g_0(0, X)],$$

and the propensity score reweighting representation,

$$\theta_0 = \mathrm{E}[YH_0].$$

Unfortunately *neither* of these representations is Neyman orthogonal, making them unsuitable for plugging-in machine learning estimators. In sharp contrast, the representation (10.3.3) is Neyman orthogonal, which implies that we can readily deploy ML methods for estimation using the empirical analog of this expression coupled with cross-fitting.

Recall we introduced Neyman orthogonality in Chapter 4. We continue this discussion formally in Section 10.4.

The construction provided in (10.3.1) is equally applicable in cases where the propensity score $\mathrm{P}(D = 1 \mid X)$ is known, as in stratified randomized experiments, and in cases where the propensity score is unknown. When the propensity score is known, the role of regression adjustment in (10.3.1) is to reduce estimation noise.

We will employ the Neyman orthogonal parameterization and cross-fitting to construct a high-quality estimator and perform statistical inference on the target parameter.

DML for APEs/ATEs in IRM

1. Partition sample indices into random folds of approximately equal size: $\{1, ..., n\} = \cup_{k=1}^{K} I_k$. For each $k = 1, ..., K$, compute estimators $\hat{g}_{[k]}$ and $\hat{m}_{[k]}$ of the

conditional expectation functions g_0 and m_0, leaving out the k-th block of data, such that $\epsilon \leq \hat{m}_{[k]} \leq 1-\epsilon$, and for each $i \in I_k$ compute

$$\hat{\varphi}(W_i) = \hat{g}_{[k]}(1, X_i) - \hat{g}_{[k]}(0, X_i) + (Y_i - \hat{g}_{[k]}(D_i, X_i))\hat{H}_i$$

with

$$\hat{H}_i = \frac{1(D_i = 1)}{\hat{m}_{[k]}(X_i)} - \frac{1(D_i = 0)}{1 - \hat{m}_{[k]}(X_i)}.$$

2. Compute the estimator

$$\hat{\theta} = \mathbb{E}_n[\hat{\varphi}(W)].$$

3. Construct standard errors via

$$\sqrt{\hat{\mathbb{V}}/n}, \quad \hat{\mathbb{V}} = \mathbb{E}_n[\hat{\varphi}(W) - \hat{\theta}]^2,$$

and use standard normal critical values for inference.

Remark 10.3.2 (Trimming) An important practical issue is trimming $|\hat{H}_i|$ so it does not take on very large values. Large values can occur when estimated propensity scores are near 0 or 1, which may indicate failure of the overlap condition – Assumption 5.2.2 in Chapter 5 and restated in Theorem 10.3.1 below. In the algorithm above, $\hat{H}_i$ can take on the largest absolute value of $\bar{H} = 1/\epsilon$. Therefore, setting $\epsilon = .01$ corresponds to $\bar{H} = 100$. While this choice provide a simple rule-of-thumb, there does not currently seem to be a good theoretical and practical resolution to the question of how to do trimming. Exploring this topic further is potentially an interesting area for more research.

Theorem 10.3.1 (Adaptive Inference on ATE with DML) *Suppose conditions specified in [2] hold. In particular, suppose that the overlap condition holds, namely for some $\epsilon > 0$ with probability 1*

$$\epsilon < m_0(X) < 1 - \epsilon.$$

If estimators $\hat{g}_{[k]}(D, X)$ and $\hat{m}_{[k]}(X)$ are such that $\epsilon \leq \hat{m}_{[k]}(X) \leq 1 - \epsilon$ and provide sufficiently high-quality approximations to the best predictors $g_0(D, X)$ and $m_0(X)$ such that

$$\|\hat{g}_{[k]} - g_0\|_{L^2} + \|\hat{m}_{[k]} - m_0\|_{L^2} + \sqrt{n}\|\hat{g}_{[k]} - g_0\|_{L^2}\|\hat{m}_{[k]} - m_0\|_{L^2} \approx 0,$$

then the estimation error in these nuisance parameter has no first

order effect on $\hat{\theta}$*:*

$$\sqrt{n}(\hat{\theta} - \theta_0) \approx \sqrt{n}\mathbb{E}_n[\varphi_0(W) - \theta_0].$$

Consequently, the estimator concentrates in $1/\sqrt{n}$ *neighborhood of* θ_0*, with deviations controlled by the Gaussian law:*

$$\sqrt{n}(\hat{\theta} - \theta_0) \overset{a}{\sim} N(0, \mathsf{V})$$

where

$$\mathsf{V} = \mathrm{E}[(\varphi_0(W) - \theta_0)^2].$$

The condition on the quality of estimators of g_0 and m_0 provides a possibility of "trading off" the quality of each estimator while retaining the adaptive inference property. The better we estimate the propensity score m_0, the worse our estimate of the regression function g_0 can be; and vice versa.

DML Inference for GATEs and ATETs

As discussed in Chapter 5, we may also be interested in average effects for interesting subpopulations such as group ATEs (GATEs) or average treatment effect on the treated (ATET). Recall that a GATE is defined as the average treatment effect within a group:

$$\theta_0 = \mathrm{E}[g_0(1, X) - g_0(0, X) \mid G = 1],$$

where G is a group indicator. For example, we might be interested in the impact of a vaccine on teenagers, in which case we could set $G = 1(13 \leq \texttt{Age} \leq 19)$, or on older individuals, in which case we might set $G = 1(65 \leq \texttt{Age})$.

GATEs are of interest for describing heterogeneity of the average treatment effects across groups. This parameter also has a predictive interpretation in a non-causal sense: It measures the average change in prediction as D switches from 0 to 1, averaging over characteristics of the group $G = 1$.

Another common target parameter ATET:

$$\theta_0 = \mathrm{E}[g_0(1, X) - g_0(0, X) \mid D = 1].$$

In business applications, the ATET is often of the interest for attribution calculations. For example, if the treatment of interest is having experience with a new product, the ATET captures the effect of the new product on those that actually received it.

DML estimation and inference for GATEs can be carried out similarly to estimation and inference for the ATE by exploiting the relation

$$\theta_0 = \mathrm{E}[\varphi_0(X) \mid G = 1] = \mathrm{E}[\varphi_0(X)G]/\mathrm{P}(G = 1).$$

We provide further detail for DML estimators of GATEs and ATETs in Section 10.4.

Remark 10.3.3 (Misspecification of PLM as inference on an overlap-weighted APE) In the case of binary treatment $D \in \{0, 1\}$, the IRM (Eqs. 10.3.1 and 10.3.2) generalizes the PLM of Section 10.2 (Eq. 10.2.1) by permitting interaction between the treatment and controls. The PLM, nonetheless, admits a very simple estimator for the treatment coefficient via partialling out: simply regress cross-fitted outcome residuals on cross-fitted treatment residuals, never dividing by propensity scores. What does this get at, however, when the PLM fails to hold? Per Remark 10.2.2, we need only consider the BLP of $\tilde{Y}$ in terms of $\tilde{D}$ in the more general IRM. Writing

$$g_0(D, X) = g_0(0, X) + D(g_0(0, X) - g_0(1, X)),$$

we see that

$$\tilde{Y} = \tilde{D}(g_0(1, X) - g_0(0, X)) + \epsilon.$$

Since $\mathrm{E}[\tilde{D}^2 \mid X] = m_0(X)(1 - m_0(X))$, we find that the estimand is

$$\beta = \frac{\mathrm{E}[m_0(X)(1 - m_0(X))(g_0(1, X) - g_0(0, X))]}{\mathrm{E}[m_0(X)(1 - m_0(X))]}.$$

That is, the APE on the population reweighted by $m_0(X)(1 - m_0(X))/\mathrm{E}[m_0(X)(1 - m_0(X))]$. These weights are known as *overlap weights* as they upweight when $m_0(X)$ is close to $1/2$ and downweight when $m_0(X)$ is close to 0 or 1.

In the case of a continuous univariate treatment on $[0, 1]$, we can leverage the same idea of writing $g_0(D, X)$ as a baseline plus the effect of D using the fundamental theorem of calculus: $g_0(D, X) = g_0(0, X) + \int_0^1 \mathbb{1}[D > t] g_0'(t, X) dt$, where g_0' is the derivative in the first argument. We can then find that β identifies the weighted average derivative

$$\beta = \mathrm{E}[w(D, X)g_0'(D, X)]/\mathrm{E}[w(D, X)]$$

for nonnegative weights

$$w(d, x) = \mathrm{E}[\tilde{D}\mathbb{I}[D > d] \mid X = x]/f(d \mid x) \geq 0,$$

where $f(d \mid x)$ is the conditional density of D given $X = x$. See, e.g., Sec. 2.3.1 of [5]. That is, the PLM coefficient estimates *some* average causal effect of increasing every value of D by an infinitesimal amount. However, the population over which we average may be highly uninterpertable.

The Effect of 401(k) Eligibility on Net Financial Assets

R Notebook on DML for Impact of 401(K) Eligibility on Financial Wealth and Python Notebook on DML for Impact of 401(K) Eligibility on Financial Wealth provide application of DML inference to learn predictive/causal effects of 401(K) eligibility on net financial wealth.

Here we re-analyze the impact of 401(k) eligibility on financial assets (Poterba et al., [6] and [7]). The data covers a short period a few years after the introduction of 401(k)'s when they were rapidly increasing in popularity.

The key problem in determining the effect of 401(k) eligibility is that working for a firm that offers access to a 401(k) plan is not randomly assigned. To overcome the lack of random assignment, we follow the strategy developed in [6] and [7]. In these papers, the authors use data from the 1991 Survey of Income and Program Participation and argue that eligibility for enrolling in a 401(k) plan in this data can be taken as exogenous after conditioning on a few observables of which the most important for their argument is income.

The basic idea of their argument is that, at least around the time 401(k)'s initially became available, people were unlikely to be basing their employment decisions on whether an employer offered a 401(k) but would instead focus on income and other aspects of the job. Following this argument, whether one is eligible for a 401(k) may then be taken as exogenous after appropriately conditioning on income and other control variables related to job choice.

Compare this argument to the one given below using DAGs.

A key component of the argument underlying the exogeneity of 401(k) eligibility is that eligibility may only be taken as exogenous after conditioning on income and other variables related to job choice that may correlate with whether a firm offers a 401(k). [6] and [7] and many subsequent papers adopt this argument but control for parsimonious, pre-specified functions of what they deem to be relevant characteristics. One might wonder whether such specifications are able to adequately control for income and other related confounders. At the same time, the power to learn about treatment effects decreases as

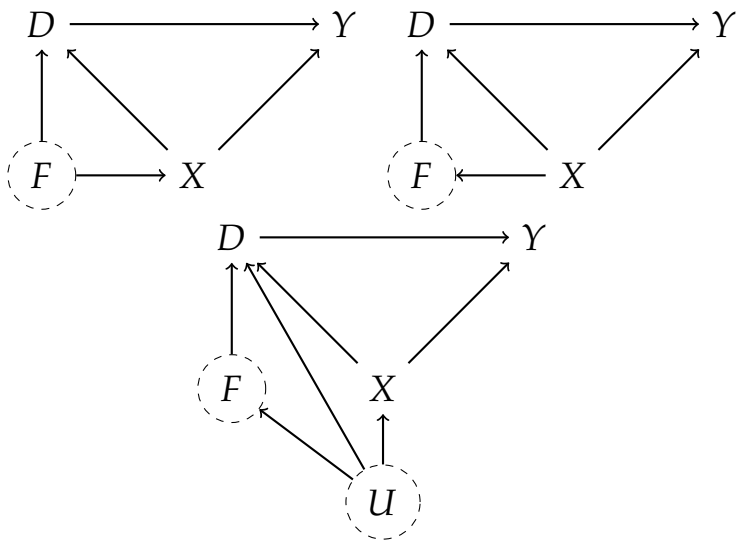

Figure 10.6: Three Causal DAGs for analysis of the 401(K) example in which adjusting for X is a valid identification strategy. The bottom figure encompasses the other two as special cases.

one allows more flexible models. The principled use of flexible ML tools offers one resolution to this tension.

In what follows, we use net financial assets[7] as the outcome variable, Y, in the analysis. The treatment variable, D, is an indicator for being eligible to enroll in a 401(k) plan. The vector of raw covariates, X, consists of age, a self-reported male indicator, income, family size, years of education, a marital status indicator, a two-earner status indicator, a defined benefit pension status indicator, an IRA participation indicator, and a home ownership indicator.

7: Defined as the sum of IRA balances, 401(k) balances, checking accounts, U.S. saving bonds, other interest-earning accounts in banks and other financial institutions, other interest-earning assets (such as bonds held personally), stocks, and mutual funds less non-mortgage debt.

It is useful to think about a causal diagram that represents our thinking about identification in this example. In Figure 10.6, we provide three example DAGs for Y, the outcome; D, the 401(K) eligibility offer which depends on firm characteristics, F, which are not observed; and X, the worker characteristics. In one structure, F determines the workers characteristics (via the hiring decision), so we have $F \to X$. In another structure, workers determine the characteristics of the company they choose to work at, $X \to F$. Finally, in the last structure F, X, and D are jointly determined by a set of latent factors U. In any of these cases, X is a valid adjustment set because it is the only parent of Y (other than D).

You can explore these DAG structures in R Notebook on Dagitty-Based Identification in 401(K) Example and Python Notebook on Pgmpy-Based Identification in 401(K) Example.

It is also useful to consider structures that would break down the identification strategy. We illustrate two such structures in Figures 10.7 and 10.8. In these figures, we introduce a node for the employer match amount, M,[8] which could mediate the effect of 401(k) eligibility and have an important effect on financial wealth.

8: Employers often offer a benefit where they will match a proportion of an employee's contribution to their 401k, up to a limit. The limit is referred to as the employer match amount.

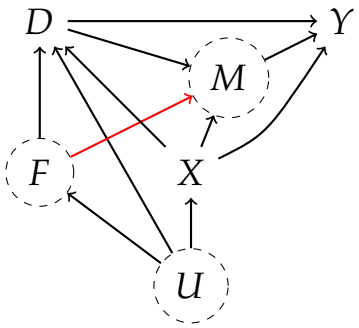

Figure 10.7: A DAG Structure where adjusting for X is not sufficient for identifying the causal effect from D to Y. If there were no arrow from F to M, adjusting for X would sufficient.

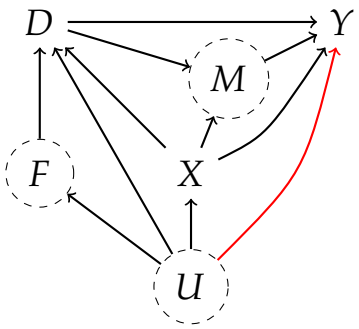

Figure 10.8: Another DAG Structure where adjusting for X is not sufficient for identifying the causal effect from D to Y. Here the latent confounder U affects all variables, so even in the absence of an arrow connecting F to M, causal effects cannot be determined after adjusting for X. The presence of such latent confounders is always a threat to causal interpretability of any observational study.

In Figure 10.7, we suppose that M is determined by unobserved firm characteristics, F, and worker characteristics, X. In this case, adjustment for X is not sufficient for identifying the causal effect from D to Y as there is a path from latent firm characteristics, which are related to the treatment, to the outcome that is not closed by X. However, if M is determined solely by D and X, so the red arrow is erased, adjustment for X is sufficient. Therefore, interpreting the target parameter of our estimation strategy as a causal effect is only valid if the match amount is independent of F given D and X, that is, if there is no arrow from F to M in the graph. Otherwise, the default interpretation is that we are estimating predictive effects of 401(k) eligibility.

In the second example, Figure 10.8, we maintain the assumption that M is independent of F given D and X by eliminating the arrow between nodes F and M. However, we now allow for the possibility that latent variables U have a direct effect on Y. That is, we have an unobserved confounder or omitted variable. In this example, such a counfounder may be unobserved risk preferences that relate to an individual's preference over jobs, an individual's characteristics, and also have direct effects on savings decisions not channeled purely through observed individual or job characteristics. In general, the possibility of latent confounders always poses a challenge to obtaining estimates of causal effects in non-experimental data. The presence or absence of latent confounders cannot be determined solely from the data in general, and thus their presence must be argued against

Table 10.4: Estimated Effect of 401(k) Eligibility on Net Financial Assets

	Lasso	Tree	Forest	Boost	Best	Ensemble
A. Partially Linear Regression Model						
Estimate	9418	8634	9112	8859	8859	9051
Std. Error	(1476)	(1303)	(1281)	(1321)	(1321)	(1315)
RMSE D	0.447	0.457	0.459	0.443	0.443	0.443
RMSE Y	58242	56819	55385	54153	54153	53917
B. Interactive Regression Model						
Estimate	8860	7856	8349	7871	8204	8146
Std. Error	(1347)	(1250)	(1502)	(1157)	(1144)	(1142)
RMSE D	0.448	0.457	0.459	0.443	0.443	0.443
RMSE Y	58300	54866	57293	55112	54866	53804

Note: Estimated ATE and standard errors from a partially linear model (Panel A) and heterogeneous effect model (Panel B) based on orthogonal estimating equations. Column labels denote the method used to estimate nuisance functions. For Lasso, we report results based on using ℓ_1 penalized logistic regression to estimate $\mathrm{E}[D|X]$. The first row provides the point estimate of the coefficient on D in the PLM in Panel A and of the ATE in Panel B, and the second row provides the standard error. Rows RMSE D and RMSE Y respectively report the cross-fitted RMSE for predicting D and Y.

based on scientific and institutional knowledge in different contexts. See, e.g., discussion in the original papers, [6] and [7], underlying this example. As in the previous example, we must interpret our estimates as predictive effects of 401(k) eligibility if we believe the connection from U to Y exists.

In Table 10.4, we report DML estimates of ATE of 401(k) eligibility on net financial assets both in the partially linear model and the interactive regression model allowing for heterogeneous treatment effects. To reduce the disproportionate impact of extreme propensity score weights in the interactive model, we trim the propensity scores at 0.01 and 0.99.

Turning to the results, it is first worth noting that when no controls are used, the estimated ATE of 401(k) eligibility on net financial assets is $19,559 with an estimated standard error of 1413. Of course, this number is not a valid estimate of the causal effect of 401(k) eligibility on financial assets if there are neglected confounding variables as suggested by [6] and [7]. When we turn to the estimates that flexibly account for confounding reported in Table 10.4, we see that they are substantially attenuated relative to this baseline that does not account for confounding, suggesting much smaller causal effects of 401(k) eligibility on financial asset holdings.

It is interesting and reassuring that the results obtained from the different flexible methods are broadly consistent with each other. Given that the predictive performance of the methods is relatively similar, this similarity is consistent with the theory that suggests that results obtained through the use of orthogonal estimating equations and any method that provides sufficiently high-quality estimates of the necessary nuisance functions should be similar. Finally, it is interesting that these results are also broadly consistent with those reported in the original work of [6] and [7] which used a simple, intuitively-motivated functional form, suggesting that this intuitive choice was sufficiently flexible to capture much of the confounding variation in this example.

Finally, we can conclude the discussion with a more sobering note that there are credible deviations in the graph structure (e.g. unobserved firm characteristics may affect the match amount) that challenge causal interpretation of the estimates. One approach to dealing with such deviations would be to conduct thorough sensitivity analysis.[9] We discuss an approach to sensitivity analysis in the DML framework in Chapter 12.

9: We have done some informal simulations to assess the impact of this threat using the that firms match up to 5% of employees' salary. In this scenario, we estimate the size of the bias to be in the ball park of 10%. Given this, we believe the results reported here are reasonable approximations to the causal effects.

10.4 Generic Debiased (or Double) Machine Learning

Key Ingredients

As a general framework, we consider DML estimation and inference based upon a method-of-moments estimator for some low-dimensional target parameter θ_0 based upon the empirical analog of the moment condition

$$\mathrm{E}[\psi(W; \theta_0, \eta_0) = 0]. \tag{10.4.1}$$

In (**??**), ψ is the *score function*, W denotes a data vector, θ_0 denotes the true value of a low-dimensional parameter of interest, and η denotes nuisance parameters with true value η_0.

The first key input of the generic DML procedure is using a score function $\psi(W; \theta, \eta)$ such that (i)

$$\mathsf{M}(\theta, \eta) = \mathrm{E}[\psi(W; \theta, \eta)]$$

identifies θ_0 when $\eta = \eta_0$ – that is,

$$\mathsf{M}(\theta, \eta_0) = 0 \text{ if and only if } \theta = \theta_0 -$$

and (ii) the Neyman orthogonality condition –

$$\partial_\eta \mathsf{M}(\theta_0, \eta)\Big|_{\eta=\eta_0} = 0 - . \tag{10.4.2}$$

is satisfied.

Here, (10.4.2) ensures that the moment condition (10.4.1) used to identify and estimate θ_0 is insensitive to small perturbations of the nuisance function η around η_0.

Remark 10.4.1 The orthogonality condition is named after Neyman [8], because he was the first to propose it in the context of parametric models with nuisance parameters.

Using a Neyman orthogonal score eliminates the first order biases arising from the replacement of η_0 with a ML estimator $\hat{\eta}_0$. Eliminating this bias is important because estimators $\hat{\eta}_0$ must be heavily regularized in high-dimensional settings, so these estimators will be biased in general. The Neyman orthogonality property is responsible for the adaptivity of these estimators – namely, their approximate distribution will not depend on the fact that the estimate $\hat{\eta}_0$ contains error as long as the error is sufficiently mild.

Remark 10.4.2 (Definition of the Derivative) The derivative ∂_η denotes the pathwise (Gateaux) derivative operator. Formally it is defined via usual derivatives taken in various directions: Given any "admissible" direction $\Delta = \eta - \eta_0$ and scalar deviation amount t, we have that

$$\partial_\eta \mathsf{M}(\theta, \eta)[\Delta] := \partial_t \mathsf{M}(\theta, \eta + t\Delta)\Big|_{t=0}.$$

The statement

$$\partial_\eta \mathsf{M}(\theta_0, \eta_0) = 0$$

means that $\partial_\eta \mathsf{M}(\theta_0, \eta_0)[\Delta] = 0$ for any admissible direction Δ. The direction Δ is admissible if $\eta_0 + t\Delta$ is in the parameter space for η for all small values of t.

The second key input is the use of high-quality machine learning estimators of the nuisance parameters. A sufficient condition in the examples given includes the requirement

$$n^{1/4}\|\hat{\eta} - \eta_0\|_{L^2} \approx 0.$$

Different structured assumptions on η_0 allow us to use different machine-learning tools for estimating η_0. For instance,

1) approximate sparsity for η_0 with respect to some dictionary calls for the use of Lasso, post-Lasso, or other sparsity-based techniques;
2) well-approximability of η_0 by trees calls for the use of regression trees and random forests;
3) well-approximability of η_0 by sparse deep neural nets calls for the use of ℓ_1-penalized deep neural networks;
4) well-approximability of η_0 by at least one model mentioned in 1)-3) above calls for the use of an ensemble/best choice method over the estimation methods mentioned in 1)-3).

There are performance guarantees for most of these ML methods that make it possible to satisfy the sufficient rate condition $n^{1/4}\|\hat{\eta} - \eta_0\|_{L^2} \approx 0$. We note that many of these convergence guarantees rely on constructions and tuning choices that do not necessarily align with the way these methods are often applied in practice. There thus remains more work to be done in understanding the behavior of ML estimators. Finally, the use of Ensemble and best choice methods ensures that the performance guarantee is no worse than the performance of the best method.

The third key input is to use sample splitting were nuisance functions are estimated on different data than are used in their evaluation when producing the estimator of the main parameter θ_0. The use of sample splitting allows us to avoid *biases* arising from overfitting.

Overfitting can easily occur when using highly complex fitting methods such as boosting, random forests, deep nets, ensembles, and other hybrid machine learning methods. We may heuristically think of overfitting as capturing noise that is particular to the observations used to fit a model in addition to signal. Using overfit estimates of nuisance parameters obtained

using the same data as used to estimate the target parameter then heuristically leads to estimation error in these parameters being correlated to outcomes which introduces a type of bias. This bias can be very large, as illustrated in Figure 10.2. We specifically use cross-fitted forms of the empirical moments, as detailed in the **Generic DML Algorithm** below, in estimation of θ_0 to avoid this problem.

Neyman Orthogonal Scores for Regression Problems

Scores for the Partially Linear Regression Model. In the PLM, we employ the score function

$$\begin{aligned}\psi(W;\theta,\eta) := \\ \{Y - \ell(X) - \theta(D - m(X))\}(D - m(X)),\end{aligned} \tag{10.4.3}$$

where $W = (Y, D, X)$ are observable variables, and η is the nuisance parameter $\eta = (\ell, m)$ with true value $\eta_0 = (\ell_0, m_0)$. Here, ℓ and m are square-integrable functions mapping the support of X to $\mathbb{R}$ whose true values are given by

$$\ell_0(X) = \mathrm{E}[Y \mid X], \quad m_0(X) = \mathrm{E}[D \mid X].$$

The score above is Neyman orthogonal by elementary calculations delegated to Section 10.B. The objects $Y - \ell(X)$ and $D - m(X)$ in the PLM score function (10.4.3) are also clearly the flexible analogs of taking residuals from linear models discussed in Chapter 1.

Scores for Interactive Regression Model. For estimation of the ATE parameter in the IRM model, we employ the score

$$\begin{aligned}\psi_1(W;\theta,\eta) := (g(1,X) - g(0,X)) \\ + H(D,X)(Y - g(D,X)) - \theta,\end{aligned} \tag{10.4.4}$$

where

$$H(D,X) := \frac{D}{m(X)} - \frac{(1-D)}{1-m(X)}, \tag{10.4.5}$$

$W = (Y, D, X)$ are observable variables, and $\eta := (g, m)$ is the nuisance parameter with true value $\eta_0 = (g_0, m_0)$. Here, g is a square-integrable function mapping the support of (D, X) to $\mathbb{R}$, and m is a function mapping the support of X to $(\varepsilon, 1-\varepsilon)$ for

some $\varepsilon \in (0, 1/2)$. The true values of g and m are given by

$$g_0(D, X) = \mathrm{E}[Y \mid D, X], \quad m_0(X) = \mathrm{P}[D = 1 \mid X]. \quad (10.4.6)$$

The score above is Neyman orthogonal by elementary calculations delegated to Section 10.B.

For estimation of GATEs, we use the score

$$\psi(W; \theta, \eta) := \tfrac{G}{p}\psi_1(W; \theta, \eta); \quad (10.4.7)$$

where G denotes the group membership indicator, the nuisance parameter η is (g, m, p) with true value $\eta_0 = (g_0, m_0, p_0)$ for g_0 and m_0 defined in (10.4.6) and $p_0 = \mathrm{P}(G = 1)$, and ψ_1 is the score for the ATE parameter defined in (10.4.4).

For estimation of the ATET parameter, we use the score

$$\psi(W; \theta, \eta) := H(D, X)\frac{m(X)}{p}(Y - g(0, X)) - \frac{D\theta}{p}, \quad (10.4.8)$$

where $H(D, X)$ is given in (10.4.5), and $\eta = (g, m, p)$ is the nuisance parameter with the true value $\eta_0 = (g_0, m_0, p_0)$ for g_0 and m_0 defined in (10.4.6) and $p_0 = \mathrm{P}(D = 1)$. Note that this score does not require estimating $g_0(1, X)$.

The scores for GATEs and ATET can be shown to be Neyman orthogonal by calculations similar to those in Section 10.B.

The DML Inference Method

We assume that we have a sample $\{W_i\}_{i=1}^n$, modeled as i.i.d. copies of random variable W, whose law is determined by the probability measure P.

Generic DML Algorithm

1. **Inputs:** Provide the data frame $\{W_i\}_{i=1}^n$, the Neyman orthogonal score/moment function $\psi(W, \theta, \eta)$ that identifies the statistical parameter of interest, and estimation method(s) for η.

2. **Train ML Predictors on Folds:** Take a K-fold random partition $(I_k)_{k=1}^K$ of observation indices $\{1, ..., n\}$ such that the size of each fold is about the same. For each $k \in \{1, \ldots, K\}$, construct a high-quality machine learning estimator $\hat{\eta}_{[k]}$ that depends only on the

subset of data $(X_i)_{i \notin I_k}$ that excludes the k-th fold.

3. **Estimate Moments:** Letting $k(i) = \{k : i \in I_k\}$, construct the moment equation estimate

$$\hat{\mathsf{M}}(\theta, \hat{\eta}) = \frac{1}{n} \sum_{i=1}^{n} \psi(W_i; \theta, \hat{\eta}_{[k(i)]})$$

4. **Compute the Estimator:** Set the estimator $\hat{\theta}$ as the solution to the equation.

$$\hat{\mathsf{M}}(\hat{\theta}, \hat{\eta}) = 0. \tag{10.4.9}$$

5. **Estimate Its Variance:** Estimate the asymptotic variance of $\hat{\theta}$ by

$$\hat{\mathsf{V}} = \frac{1}{n} \sum_{i=1}^{n} [\hat{\varphi}(W_i)\hat{\varphi}(W_i)'] - \frac{1}{n} \sum_{i=1}^{n} [\hat{\varphi}(W_i)] \frac{1}{n} \sum_{i=1}^{n} [\hat{\varphi}(W_i)]',$$

where

$$\hat{\varphi}(W_i) = -\hat{J}_0^{-1} \psi(W_i; \hat{\theta}, \hat{\eta}_{[k(i)]})$$

and

$$\hat{J}_0 := \partial_\theta \frac{1}{n} \sum_{i=1}^{n} \psi(W_i; \hat{\theta}, \hat{\eta}_{[k(i)]}).$$

6. **Confidence Intervals:** Form an approximate $(1 - \alpha)\%$ confidence interval for any functional $c'\theta_0$, where c is a vector of constants, as

$$[c'\hat{\theta} \pm z_{1-\alpha/2} \sqrt{c'\hat{\mathsf{V}}c/n}],$$

where $z_{1-\alpha/2}$ is the $(1 - \alpha/2)$ quantile of the $N(0, 1)$ distribution.

Remark 10.4.3 (The Case of Linear Scores) The score for most of our examples is linear in θ; that is, the score can be written as

$$\psi(W; \theta, \eta) = \psi^b(W; \eta) - \psi^a(W; \eta)\theta.$$

In such cases the estimator takes the form

$$\hat{\theta} = \hat{J}_0^{-1} \frac{1}{n} \sum_{i=1}^{n} \psi^b(W_i; \hat{\eta}_{[k(i)]}). \tag{10.4.10}$$

where $\hat{J}_0 = \frac{1}{n} \sum_{i=1}^{n} \psi^a(W_i; \hat{\eta}_{[k(i)]})$.

Remark 10.4.4 (Sample Splitting) In step 2), the estimator $\hat{\eta}_{[k]}$ can be an ensemble or aggregation of several estimators as long as we only use the data $(X_i)_{i \notin I_k}$ outside the k-th fold to construct the estimators.

Remark 10.4.5 (Choosing the number of folds) In our experience, choosing $K = 4 - 5$ has seemed to work well in a variety of empirical examples and in simulations for medium-sized data sets. For smaller data sets, a larger K seems to work better, and we typically recommend $K \geq 10$ for small data sets. There is still room for work on obtaining a better understanding of the impact of K and more principled guidance on its choice.

Properties of the General DML Estimator

We turn now to the properties of the DML estimator under the assumption of strong identification.

Definition 10.4.1 (Strong Identification) *We have that* $\mathsf{M}(\theta, \eta_0) = 0$ *if and and only if* $\theta = \theta_0$, *and that*

$$J_0 := \partial_\theta \mathrm{E}[\psi(W; \theta_0, \eta_0)]$$

has singular values that are bounded away from zero.

In the context of the PLM, the latter condition is satisfied if $\mathrm{E}[\tilde{D}^2]$ is bounded away from 0, that is, if $\tilde{D}$ has non-trivial variation left after partialing-out controls. In the context of the IRM, the latter condition is satisfied if the overlap condition holds.

Theorem 10.4.1 (Generic Adaptive Inference with DML) *Assume that estimates of nuisance parameters are of sufficiently high-quality, as specified in [2]. Assume strong identification holds.*

Then, estimation of nuisance parameter does not affect the behavior

of the estimator to the first order; namely,

$$\sqrt{n}(\hat{\theta} - \theta_0) \approx \sqrt{n}\mathbb{E}_n[\varphi_0(W)],$$

where

$$\varphi_0(W) = -J_0^{-1}\psi(W; \theta_0, \eta_0), \quad J_0 := \partial_\theta \mathrm{E}[\psi(W; \theta_0, \eta_0)],$$

and $J_0 = \mathrm{E}[\psi^a(W; \eta_0)]$ *for linear scores (see Remark 10.4.3).*

Consequently, $\hat{\theta}$ *concentrates in a* $1/\sqrt{n}$*-neighborhood of* θ_0 *and the sampling error* $\sqrt{n}(\hat{\theta} - \theta_0)$ *is approximately normal:*

$$\sqrt{n}(\hat{\theta} - \theta_0) \overset{a}{\sim} N(0, \mathsf{V}), \quad \mathsf{V} := \mathrm{E}[\varphi_0(W)\varphi_0(W)'].$$

Theorem 10.4.2 *Under the same regularity conditions, the interval* $[c'\hat{\theta} \pm z_{1-\alpha/2}\sqrt{c'\hat{\mathsf{V}}c/n}]$ *where* $z_{1-\alpha/2}$ *is the* $(1-\alpha/2)$ *quantile of the* $N(0,1)$ *distribution contains* $c'\theta_0$ *for approximately* $(1-\alpha)\times 100$ *percent of data realizations:*

$$\mathrm{P}\left(c'\theta_0 \in [c'\hat{\theta} \pm z_{1-\alpha/2}\sqrt{c'\hat{\mathsf{V}}c/n}]\right) \approx (1-\alpha).$$

Selection of the Best ML Methods for DML to Minimize Upper Bounds on Bias. In many problems the nuisance parameters are regression functions

$$\eta_m = \mathrm{E}[V_m \mid X_m], \quad m \in \{1, ..., M\},$$

where V_m are some response variables and X_m are covariate vectors. Consider a set of ML methods enumerated by $j \in \{1, ..., J\}$ that produce estimates $\hat{\eta}_{mj[k]}$ when applied to data excluding the k-th fold. We have that

$$\check{V}_{i,mj} = V_i - \hat{\eta}_{mj[k(i)]}(X_i), \quad i \in I_k.$$

Selection of the Best ML Methods for DML to Minimize Bias.

- For each method j, compute the cross-fitted MSPEs

$$\mathbb{E}_n[\check{V}_{mj}^2].$$

- Select the best ML method for predicting V_m via

$$\hat{j}_m = \arg\min_j \mathbb{E}_n[\check{V}_{mj}^2].$$

- Use the method $\hat{f}_m$ as a learner of η_m in the Generic DML Algorithm.

Corollary 10.4.3 *The results of Theorems 10.4.1 and 10.4.2 continue to hold if J is small.*

The precise conditions may depend on the problem at hand. See Remark 10.2.3 for discussion in the context of the partially linear model.

Notebooks

- R Notebook on DML for Impact of Gun Ownership on Homicide Rates and Python Notebook on DML for Impact of Gun Ownership on Homicide Rates provide an application of DML inference to learn predictive/causal effects of gun ownership on homicide rates across U.S. counties.

- R Notebook on Dagitty-Based Identification in 401(K) Example and Python Notebook on Pgmpy-Based Identification in 401(K) Example analyze graph structures that enable identification of the causal effect of 401(K) eligibility on net financial wealth.

- R Notebook on DML for Impact of 401(K) Eligibility on Financial Wealth and Python Notebook on DML for Impact of 401(K) Eligibility on Financial Wealth provide application of DML inference to learn predictive/causal effects of 401(K) eligibility on net financial wealth.

- R Notebook on DML for Growth Regression Analysis and Python Notebook on DML for Growth Regression Analysis build upon the application discussed in Chapter 4 by providing an application of DML inference based on ML on predictive/causal effects of countries' initial wealth on the rate of economic growth.

Notes

For a detailed literature review and technical regularity conditions needed for each of theorems, see [2], which also gives an

overview of various analytical methods for generating Neyman orthogonal scores in a wide variety of problems.

The paper [9] goes further and describes methods for generating higher-order orthogonal scores:

$$\partial_\eta \partial_\eta \mathrm{E}[\psi(\theta_0, \eta_0)] = 0.$$

The use of higher-order orthogonal scores allows even weaker requirements for the quality of machine learning estimators of the form,

$$n^{1/6} \|\hat{\eta} - \eta_0\|_{L^2} \approx 0,$$

with the caveat that such higher-order orthogonal scores may not always exist for certain subsets of distributions.

The DML method, developed in Chernozhukov, Chetverikov, Demirer, Duflo, Hansen, Newey, and Robins [2], is simply a practical meta-recipe that explicitly incorporates many classical ideas from the parametric and semi-parametric econometrics and statistics literature; see, e.g., Neyman [8]; Bickel, Klassen, Ritov, Wellner [10]; Newey [11]; Robinson [12]; and Robins and Rotnitzky [13]. The intent was to combine ideas from the classical semi-parametric learning literature and prediction methods from the modern machine learning literature to provide immediately practical methods that are ready for rigorous statistical inference on predictive and causal effects. In essence, the approach can be viewed as a modernized version of the "one"-step debiasing correction proposed by Neyman; see, e.g. [14] for a review.

The partialling-out approach has long been employed in classical econometrics. Robinson [12] was the first to employ it in the context of kernel regressions. [2] extended this approach to more modern settings where ML estimators are used for partialling out, with cross-fitting enabling the extension.

For ATE, GATEs and ATET parameters, DML (or "doubly robust" ML) reduces to the use of machine learned "doubly robust scores" with cross-fitting. The idea of using doubly robust scores (also called augmented inverse propensity score weighted scores) is due to Robins and Rotnitzky [13], but also arises as a special case of Newey's [11] fundamental analysis.

Targeted maximum likelihood estimation (TMLE) is another general approach for building orthogonal estimators [15]. This approach relies on doing maximum likelihood estimation for a target parameter, using a least favorable parametric submodel for the parameter of interest as the likelihood function. As with DML, TMLE needs to be combined with cross-fitting in order to

deal with general ML estimators to avoid overfitting. The DML and cross-fitted TMLE should generally produce first order equivalent answers under correct specification. However, using TMLE can refine the finite-sample properties.

In the context of ATE, TMLE can be seen as applying a calibrated correction to a nonlinear regression function. We regress $\check{Y}_i = Y_i - \hat{g}(D_i, X_i)$ on $\hat{H}_i$, obtaining

$$\hat{b} = \mathbb{E}_n[\check{Y}\hat{H}]/\mathbb{E}_n[\hat{H}^2].$$

Then we correct the regression function estimate by $\bar{g}(D_i, X_i) = \hat{g}(D_i, X_i) + \hat{b}\hat{H}_i$. This correction was first proposed by Sharfstein, Rotnitzky and Robins [16]. The basic idea is that we know that $Y_i - g(D_i, X_i)$ should be orthogonal to H_i. Thus, if our estimate of the regression function does not have this property, we can recalibrate the regression function so the property holds.

For guidance on using DML in empirical studies and on hyper-parameter tuning related to DML we refer to [17].

Study Problems

1. Experiment with one of the notebooks for the partially linear models (Guns example, Guns with DNNs, or Growth example). For example,
   (a) Apply the methods to a different empirical example (e.g., Penn reemployment experiment from CI-1),
   (b) or, using the same empirical example, try to use the H20 Auto ML framework as the machine learning tool to estimate m and ℓ functions. (See Chapter 9 H20 Auto ML to get started).

   Explain what you are doing to a fellow student.

2. Study the 401(K) identification notebook that uses Dagitty. Extend it to another empirical example of your choice. Explain the principles you are using to a fellow student.

3. Study the 401(K) empirical analysis notebook. Extend it to another empirical example of your choice (the Penn reemployment experiment from Chapter 1, for example) or estimate ATE for 401(K) eligibility for a subset of low income (or high-income) workers (Group ATEs).

4. (Theoretical). Explain to a friend the concept of Neyman orthogonality, illustrating it with one of the examples in

Appendix B. Extend the calculations in Appendix B to verify Neyman orthogonality for the ATET score specified in (10.4.8).

5. (Theoretical). Explain to a friend the concept of Neyman orthogonality, and explain why the formulations given in Remark 10.3.1 are not Neyman orthogonal.

10.A Bias Bounds with Proxy Treatments

Here we explain the measurement error bias in the partially linear structural equation model where treatment is measured with error:

$$\begin{aligned} Y &:= \alpha G + g_Y(X) + \epsilon_Y; \\ D &:= G + g_D(X) + \epsilon_D; \\ G &:= g_G(X) + \epsilon_G; \\ X &:= \epsilon_X; \end{aligned}$$

where ϵ's are independent and centered. The second equation states that D is generated as a proxy for the actual treatment G using a partially linear structure. In partialled-out form

$$\begin{aligned} \tilde{Y} &:= \alpha\epsilon_G + \epsilon_Y; \\ \tilde{D} &:= \epsilon_G + \epsilon_D; \\ \tilde{G} &:= \epsilon_G. \end{aligned}$$

The projection of $\tilde{Y}$ on $\tilde{D}$ recovers the projection coefficient:

$$\beta = \mathrm{E}[\tilde{Y}\tilde{D}]/\mathrm{E}[\tilde{D}^2] = \alpha\mathrm{E}[\epsilon_G^2]/(\mathrm{E}[\epsilon_G^2] + \mathrm{E}[\epsilon_D^2]).$$

It follows that there is attenuation bias in the estimable quantity β relative to the target parameter α:

$$|\beta| < |\alpha|.$$

As the proxy error $\mathrm{E}[\epsilon_D^2]$ becomes small, the difference between β and α becomes small. Specifically, if $\mathrm{E}[\epsilon_D^2] \to 0$, then $\beta \to \alpha$.

If we somehow knew that

$$R^2_{\tilde{D}\sim\tilde{G}} := \mathrm{E}[\epsilon_G^2]/(\mathrm{E}[\epsilon_G^2] + \mathrm{E}[\epsilon_D^2]) \geq \underline{r} -$$

that is, if we knew that the true treatment G explains at least $\underline{r}$ of the variance of the proxy treatment D – then we could construct the upper and lower bound on α from β. E.g. when $\beta > 0$, we would have

$$\beta \leq \alpha \leq \beta/R^2_{D\sim G} = (1/\underline{r})\beta.$$

10.B Illustrative Neyman Orthogonality Calcuations

The Score in the Partially Linear Model. Consider the score for the PLM given in (10.4.3). We have that

$$\mathrm{E}[\psi(W;\beta_0,\eta_0)] = 0$$

by definition of β_0 of η_0. Let $U = (Y - \ell_0(X)) - (D - m_0(X))\beta_0)$. Then, 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 given by

$$\begin{aligned}
&\partial_\eta \mathrm{E}[\psi(W;\beta_0,\eta_0)][\Delta] \\
&\quad = -\mathrm{E}\Big[U(m(X) - m_0(X))\Big] \\
&\qquad - \mathrm{E}\Big[\Big((m(X) - m_0(X))\beta_0 + (\ell(X) - \ell_0(X))\Big)(D - m_0(X))\Big] \\
&\quad = 0,
\end{aligned}$$

by the law of iterated expectations since $\mathrm{E}[D - m_0(X) \mid X] = 0$ and $\mathrm{E}[U \mid D, X] = 0$.

The Score for IRM. Consider the score for the ATE in the IRM given in (10.4.4). We have that

$$\mathrm{E}[\psi(W;\theta_0,\eta_0)] = 0$$

by definition of θ_0 and η_0. Also, for any $\eta = (g, m)$ that are square integrable with $1/m + 1/(1 - m)$ uniformly bounded, the Gateaux derivative in the direction

$$\Delta = \eta - \eta_0 = (g - g_0, m - m_0)$$

is given by

$$
\begin{aligned}
&\partial_\eta \mathrm{E}[\psi(W;\theta_0,\eta_0)][\Delta] \\
&\quad = \mathrm{E}\Big[g(1,X) - g_0(1,X)\Big] \\
&\qquad - \mathrm{E}\Big[g(0,X) - g_0(0,X)\Big] \\
&\qquad - \mathrm{E}\Big[\frac{D(g(1,X) - g_0(1,X))}{m_0(X)}\Big] \\
&\qquad + \mathrm{E}\Big[\frac{(1-D)(g(0,X) - g_0(0,X))}{1-m_0(X)}\Big] \\
&\qquad - \mathrm{E}\Big[\frac{D(Y - g_0(1,X))(m(X) - m_0(X))}{m_0^2(X)}\Big] \\
&\qquad - \mathrm{E}\Big[\frac{(1-D)(Y - g_0(0,X))(m(X) - m_0(X))}{(1-m_0(X))^2}\Big],
\end{aligned}
$$

which is 0 by the law of iterated expectations since $\mathrm{E}[D \mid X] = m_0(X)$, $\mathrm{E}[1 - D \mid X] = 1 - m_0(X)$, $\mathrm{E}[D(Y - g_0(1,X)) \mid X] = 0$, and $\mathrm{E}[(1-D)(Y - g_0(0,X)) \mid X] = 0$.

Bibliography

[1] Jerzy Neyman. '$C(\alpha)$ tests and their use'. In: *Sankhyā: The Indian Journal of Statistics, Series A* (1979), pp. 1–21 (cited on page 250).

[2] Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. 'Double/debiased machine learning for treatment and structural parameters'. In: *Econometrics Journal* 21.1 (2018), pp. C1–C68 (cited on pages 254, 255, 269, 282, 284, 285).

[3] Philip J. Cook and Jens Ludwig. 'The social costs of gun ownership'. In: *Journal of Public Economics* 90 (2006), pp. 379–391 (cited on page 261).

[4] Jeffrey M. Wooldridge. 'Two-Way Fixed Effects, the Two-Way Mundlak Regression, and Difference-in-Differences Estimators'. In: *Available at SSRN: https://ssrn.com/abstract=3906345 or http://dx.doi.org/10.2139/ssrn.3906345* (2021) (cited on page 261).

[5] Joshua D. Angrist and Alan B. Krueger. 'Empirical Strategies in Labor Economics'. In: *Handbook of Labor Economics. Volume 3*. Ed. by O. Ashenfelter and D. Card. Elsevier: North-Holland, 1999 (cited on page 272).

[6] James M. Poterba, Steven F. Venti, and David A. Wise. '401(k) Plans and Tax-Deferred savings'. In: *Studies in the Economics of Aging*. Ed. by D. A. Wise. Chicago, IL: University of Chicago Press, 1994, pp. 105–142 (cited on pages 272, 275, 276).

[7] James M. Poterba, Steven F. Venti, and David A. Wise. 'Do 401(k) Contributions Crowd Out Other Personal Saving?' In: *Journal of Public Economics* 58.1 (1995), pp. 1–32 (cited on pages 272, 275, 276).

[8] Jerzy Neyman. 'Optimal asymptotic tests of composite hypotheses'. In: *Probability and Statistics* (1959), pp. 213–234 (cited on pages 277, 285).

[9] Lester Mackey, Vasilis Syrgkanis, and Ilias Zadik. 'Orthogonal machine learning: Power and limitations'. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 3375–3383 (cited on page 285).

[10] Peter J. Bickel, Chris A.J. Klaassen, Ya'acov Ritov, and Jon A. Wellner. *Efficient and Adaptive Estimation for Semiparametric Models*. Johns Hopkins University Press Baltimore, 1993 (cited on page 285).

[11] Whitney K. Newey. 'The asymptotic variance of semiparametric estimators'. In: *Econometrica* 62.6 (1994), pp. 1349–1382 (cited on page 285).

[12] Peter M. Robinson. 'Root-N-consistent semiparametric regression'. In: *Econometrica* 56.4 (1988), pp. 931–954. DOI: 10.2307/1912705 (cited on page 285).

[13] James M. Robins and Andrea Rotnitzky. 'Semiparametric efficiency in multivariate regression models with missing data'. In: *Journal of the American Statistical Association* 90.429 (1995), pp. 122–129 (cited on page 285).

[14] Victor Chernozhukov, Christian Hansen, and Martin Spindler. 'Valid post-selection and post-regularization inference: An elementary, general approach'. In: *Annual Review of Economics* 7.1 (2015), pp. 649–688 (cited on page 285).

[15] Mark J. van der Laan and Sherri Rose. *Targeted Learning: Causal Inference for Observational and Experimental Data*. Springer Science & Business Media, 2011 (cited on page 285).

[16] Daniel O. Scharfstein, Andrea Rotnitzky, and James M. Robins. 'Adjusting for Nonignorable Drop-Out Using Semiparametric Nonresponse Models'. In: *Journal of the American Statistical Association* 94.448 (1999), pp. 1096–1120. (Visited on 01/25/2023) (cited on page 286).

[17] Philipp Bach, Oliver Schacht, Victor Chernozhukov, Sven Klaassen, and Martin Spindler. *Hyperparameter Tuning for Causal Inference with Double Machine Learning: A Simulation Study*. 2024 (cited on page 286).