Title

Double Robust, Flexible Adjustment Methods for Causal Inference: An Overview and an Evaluation

Author

Hoffmann, Nathan Isaac

Publication Date

2024

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
Los Angeles

Double Robust, Flexible Adjustment Methods for Causal Inference: An Overview and an Evaluation

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Statistics

by

Nathan Isaac Hoffmann

2024

ABSTRACT OF THE THESIS

Double Robust, Flexible Adjustment
Methods for Causal Inference:
An Overview and an Evaluation

by

Nathan Isaac Hoffmann
Master of Science in Statistics
University of California, Los Angeles, 2024
Professor Chad J. Hazlett, Chair

This thesis is a guide to some of the latest methods in double robust, flexible covariate adjustment for causal inference, and it compares these methods to more traditional statistical methods and flexible "single robust" methods. It does this by using both simulated data where the treatment effect estimate is known, and then using comparisons of experimental and observational data from the National Supported Work Demonstration. Methods covered include Augmented Inverse Probability Weighting (AIPW), Targeted Maximum Likelihood Estimation (TMLE), and Double/Debiased Machine Learning (DML). Results suggest that some of these methods do outperform traditional methods in a wide range of simulations, but only slightly. In particular, the top performers are TMLE and AIPW in conjunction with flexible machine learning algorithms. But G-computation with the same flexible machine learning algorithms obtains almost identical results, and simple regression methods are nearly comparable in bias and are much more computationally efficient.

The thesis of Nathan Isaac Hoffmann is approved.

Onyebuchi A. Arah

Jennie E. Brand

Chad J. Hazlett, Committee Chair

University of California, Los Angeles

2024

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I deeply thank my committee – Chad Hazlett, Onyebuchi A. Arah, Jennie E. Brand – for their extensive suggestions and advice on previous drafts. I would also like to thank the members of UCLA's Practical Causal Inference Lab, UCLA's Social Inequality Data Science Lab, participants of the 2023 American Sociological Association's Annual Meeting, and participants of the 2024 All-UC Demography Conference for their feedback on previous versions of this paper.

Introduction

In causal inference, functional form misspecification of underlying models can bias estimates of treatment effects (Hernán & Robins, 2020; Morgan & Winship, 2015). There have been two important developments that attempt to overcome this. First, methodologists have developed machine learning methods that allow greater flexibility in estimation, adjusting for covariates in data-driven, complex ways (Balzer & Petersen, 2021; Brand et al., 2023). The second development is double robust methods (Bang & Robins, 2005; Kang & Schafer, 2007), which estimate two models: one for treatment exposure and another for the outcome. These models are robust to misspecification of either one of these "nuisance" models.

A number methods unifying these two developments have proliferated. These double robust methods for flexible covariate adjustment use machine learning methods to adapatively model the data generating processes at play. These models purportedly overcome the shortcomings of both traditional statistical methods and machine learning methods. Common statistical methods – such as OLS regression and matching on propensity scores estimated from logistic regression – have rigid functional form assumptions and fail to calculate stable estimates when the number of covariates is large relative to the number of observations. Machine learning methods, on the other hand, are often difficult to interpret. They can also suffer from overfitting, where the flexibility of the model becomes a weakness and predictions out-of-sample are poor, yet efforts to correct for overfitting can introduce regularization bias (Hastie et al., 2009). Double robust methods with machine learning dispose of the constricting functional form assumptions of common statistical methods, and they correct for the regularization bias of flexible machine learning methods. They can also accommodate large numbers of covariates and produce easily interpretable treatment effect estimates. Despite their apparent advantages, these methods remain rarely utilized by social scientists. Part of the barrier has been lack of familiarity with these methods. It has also been unclear how these methods compare, or whether such methods actually perform better than traditional methods in finite samples.

This thesis makes advances on these fronts. First, it is a guide to some of the latest

methods in double robust, flexible covariate adjustment for causal inference, explaining the methods to a social scientist audience. Methods covered include Augmented Inverse Probability Weighting (AIPW), Targeted Maximum Likelihood Estimation (TMLE), and Double or Debiased Machine Learning (DML). This thesis reviews the theory behind these methods as well as simple R implementations.

Second, this thesis evaluates these methods. They are tested on simulations from Dorie et al. (2019), which cover a range of data-generating processes where ignorability holds – i.e., there are no unmeasured confounders. They are then used to estimate experimental and observational effects from real-world data, the National Support for Work Demonstration (NSW) originally analyzed by LaLonde (1986). In these evaluations, double robust methods are compared to "single robust" methods (i.e., ones with one nuisance model). These include traditional or simpler statistical methods commonly used by social scientists: ordinary least squares (OLS) regression, matching on propensity scores estimated from logistic regression (PSM), and inverse probability weighting (IPW). They are also compared to two more flexible methods that may overcome the misspecification issues that double robust methods aim to overcome: G-computation and the Lin estimator (Lin, 2013).

Results from simulations where ignorability holds show that some double robust methods outperform traditional statistical methods, but only slightly. AIPW and TMLE perform the best, at least when used in conjunction with flexible machine learning algorithms, while DML does slightly worse than traditional methods. G-computation with flexible machine learning performs as well as the lowest-error double robust methods. Despite their lower average error, the computation times of these flexible methods are high. With its still relatively low error and much faster computation time, OLS remains a sensible choice as an estimator, and the Lin estimator – which is as quick as standard OLS regression – performs only slightly worse than the best double robust methods.

Results from the NSW study highlight the importance of ignorability. When observational samples are not comparable to the experimental sample, or when important covariates are not included, all methods – double robust or traditional – fail to recover experimental estimates. When appropriate covariates are included, most methods perform well.

Motivation

Literature Review

Although previous work has compared different methods for covariate adjustment, there has not yet been a thorough comparison and evaluation of the recent and popular double robust methods of AIPW, TMLE, and DML for a social scientist audience.

Although some introductions to double robust methods exist, they do not discuss them in the context of covariate adjustment for causal inference, or their treatment is overly technical for a social scientist audience. For example, Kang & Schafer (2007) provide an an excellent overview and evaluation of double robust methods, but in the context of missing data, and the authors consider AIPW but not TMLE or DML. Bang & Robins (2005) introduce double robust models for both causal inference and missing data, but their treatment is rather technical, and they only discuss AIPW. Lundberg et al. (2022) provide brief schematic overviews of double robust methods for a social scientist audience, but they do not evaluate these methods.

Existing evaluations of double robust methods have focused on only one double robust method and compared it to few traditional statistical methods. Dorie et al. (2019) compare estimations from a number of different flexible methods, but these do not include AIPW or DML. Chatton et al. (2020) compare four methods – G-computation, IPW, full matching, and TMLE – but authors only consider one double robust method, and their focus is on omitted variable bias rather than determining which method is the most useful. Cousineau et al. (2022) evaluates the performance of optimization-based methods for causal inference, but these do not include the double robust methods covered in the current paper. Knaus (2022) reviews DML-based methods in an econometrics setting but does not compare them to traditional statistical methods for covariate adjustment.

An evaluation of multiple double robust methods that compares them to traditional statistical methods as well as flexible "single robust" methods is needed to understand just how practically useful these methods are for a social science audience. This thesis does this

as well as provides a gentle introduction to these methods.

Historical Overview

According to Bang & Robins (2005), double robust methods have their origins in missing data models. Robins et al. (1994) and Rotnitzky et al. (1998) developed augmented orthogonal inverse probability-weighted (AIPW) estimators in missing data models. Drawing on the fact that causal inference is fundamentally a missing data problem, Scharfstein et al. (1999) showed that AIPW was double robust and extended to causal inference.

But Kang & Schafer (2007) argue that double robust methods are older. They cite work by Cassel et al. (1976), who proposed "generalized regression estimators" for population means from surveys where sampling weights must be estimated. Arguably, double robust methods go back even further than this. The form of double robust methods is similar to residual-on-residual regression, which dates back to the Frisch-Waugh-Lowell (FWL) theorem (Frisch & Waugh, 1933; Lovell, 1963):

$$\beta_D = \frac{\text{Cov}(\tilde{Y}_i, \tilde{D}_i)}{\text{Var}(\tilde{D}_i)}$$

where $\tilde{D}_i$ is the residual part of D_i after regressing it on X_i, and $\tilde{Y}_i$ is the residual part of Y_i after regressing it on X_i. This formulation writes the regression coefficient as composed of an outcome model ($\tilde{Y}_i$) and exposure model ($\tilde{D}_i$), the two models used in double robust estimators. Of the methods considered in this thesis, double machine learning (DML) makes this connection most explicit by using residual-on-residual regression as part of its estimation strategy.

There are also links between double robust methods and matching with regression adjustment. This work goes back at least as far as Rubin (1973), who suggested that regression adjustment in matched data produces less biased estimates that either matching (exposure adjustment) or regression (outcome adjustment) do by themselves.

Today, double robust methods abound (e.g. Arkhangelsky et al., 2021; Dukes et al., 2022;

Kennedy, 2023; Ratkovic, 2023; Słoczyński & Wooldridge, 2018). Although double robust methods exist for instrumental variables (Okui et al., 2012; Wang & Tchetgen Tchetgen, 2018), difference-in-differences (Sant'Anna & Zhao, 2020), longitudinal data (Tran et al., 2019; Yu & van der Laan, 2006), and other causal applications, this thesis focuses on three of the most popular and foundational methods for covariate adjustment in a cross-sectional setting.

Aims of Double Robust Methods

Double robust methods for covariate adjustment aim to overcome what many consider to be the shortcomings of both traditional statistical methods and flexible machine learning methods (Díaz, 2020). Statistical methods that are popular with social scientists – such as OLS regression and matching on propensity scores from logistic regression – have two main weaknesses that double robust methods address. First, they assume simple (linear or transformed linear) functional forms. In the presence of highly nonlinear data generating processes, they may provide biased estimates. Second, these methods cannot handle large numbers of covariates relative to sample size, i.e. sparsity. While some machine learning methods can produce estimates even when the number of covariates exceeds the number of observations (such as lasso), OLS fails in this case due to the $X^\top X$ matrix not being of full rank and hence not invertible. In cases with many covariates, but not more than the number of observations, estimation is unstable with many traditional statistical methods.

Flexible machine learning methods also have their drawbacks. First, naive application of these methods can result in overfitting, with predictive accuracy maximized in sample but treatment effect estimation being biased. When regularization is used to correct for overfitting, "regularization bias" can result. Furthermore, results of these machine learning methods can be difficult to interpret without further processing. Machine learning methods have often been developed with a focus on prediction rather than on producing treatment effect point estimates.

Double robust methods attempt to overcome the downsides of both traditional and

machine learning methods by incorporating flexible models into a framework that avoids overfitting and regularization bias and provides easily interpretable estimates. These methods are also motivated by the idea that many older methods ignore information present in the data. Methods tend to model either only the outcome – as in OLS regression and G-computation – or only the treatment assignment – as in propensity score matching or inverse probability weighting. Double robust methods, on the other hand, model both of these.

Conceptual Overview

Double robust methods estimate two models: an *outcome model*:

$$\mu_d(X_i) = E(Y_i \mid D_i = d, X_i) \tag{1}$$

and an *exposure model* (or treatment or propensity score model):

$$\pi(X_i) = E(D_i \mid X_i), \tag{2}$$

where $\mu_d(\cdot)$ is a model of the outcome, $D_i = d_i \in \{0, 1\}$ is the treatment assignment (where 0 is control and 1 is treated), X_i is a vector of covariates for unit $i = 1, \ldots, N$, Y_i is the outcome, and $\pi(\cdot)$ is a model of the exposure. The covariates included in X_i can be different for the two models.

The focus of this thesis is on the average treatment effect (ATE), which under the potential outcomes framework (Rubin, 1974) is defined as

$$\tau = E[Y_i(1) - Y_i(0)],$$

where $Y_i(1)$ and $Y_i(0)$ are the potential outcomes of Y_i under treatment and control, respectively. An estimator is called "double robust" if it achieves consistent estimation of the ATE (or whatever estimand the researcher is interested in) as long as *at least one* of Equations (1) or (2) is consistently estimated. This means that the outcome model can be completely

misspecified, but as long as the exposure model is correct, our estimation of the ATE will be consistent. This also means that the exposure model can be completely wrong, as along as the outcome model is correct.

It is important to consider what is meant by a "correct" model specification (Keil et al., 2018). These estimators are robust from a statistical standpoint, but not necessarily a causal identification one. The researcher must know which variables are possible confounders and to include them in the appropriate models, while not including colliders or mediators (Hünermund et al., 2023). The simulations discussed in this thesis assume conditional ignorability; rather than testing what happens when models are missing important covariates, it focuses on accurate specification of the functional form of the treatment and outcome models.

Assumptions

Most double robust methods require almost all of the standard assumptions necessary for most methods that depend on selection on observables. Although some double robust methods relax one or two of these, the methods discussed in this thesis rely on six standard assumptions when estimating the ATE.

1. Consistency: $Y_i(d) = Y_i \mid D_i = d$, i.e. under treatment (control), we observe the potential outcome under treatment (control).

2. One version of treatment: All treated units receive the same version of treatment.

3. No interference: $Y_i(D_i, D_j) = Y_i(D_i)$, i.e. the potential outcome for one unit depends only on its own treatment, not the value of other units' treatment.

4. Positivity/overlap: $0 < \Pr(D = 1 \mid X = x) < 1$ for all values of X, i.e. there is non-zero probability of receiving treatment or control for every combination of covariates in the data. This means we can find at least one control unit to compare every treated unit to (and vice versa).

5. Independent and identically distributed (IID) observations: In order to make population-level inference, the sample needs to be representative of the population.

6. Conditional ignorability: $\{Y_{i0}, Y_{i1}\} \perp\!\!\!\perp D_i \mid X_i$, i.e. there are no unmeasured confounders.

The first three assumptions are embedded in the potential outcomes notation. Assumptions 2 and 3, together, are also called the Stable Unit Treatment Value assumption (SUTVA, Rubin, 1980). Special attention should be paid to Assumption 6: double robust methods will not work if we do not measure an important confounder that affects both treatment and exposure. But notably, the double robust methods covered in this tutorial make no functional form assumptions.

Overview of Techniques

Each of the methods reviewed in this thesis can be thought of as a collection of estimation techniques. Each involves a model for the outcome and another for the treatment exposure, but choice of estimation technique for these two models is left to the discretion of the user. The ways these estimated models relate and are combined into a final effect estimate vary between double robust methods.

Augmented Inverse Probability Weighting (AIPW)

The oldest of these modern methods, AIPW arose in the context of missing data imputation (Robins et al., 1994). Scharfstein et al. (1999) showed that AIPW was double robust and extended to causal inference. Introductions to AIPW exist in the contexts of political science (Glynn & Quinn, 2010) and econometrics (Funk et al., 2011). The `AIPW` R package provides a simple implementation of the method (Zhong et al., 2021).

AIPW combines estimates from a model for the treatment exposure, $\pi(X)$, and a model for the outcome, $\mu(X)$. The name comes from the close similarity to inverse probability weights (IPW), but whereas IPW only weights for probability of treatment, AIPW "augments" these weights with an estimate of the response surface.

Formally, the model can be written as the difference between an estimated outcome for treated units and an estimated outcome for untreated units (see the demonstration below):

$$\hat{\tau}_{AIPW} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{D_i(Y_i - \hat{\mu}_1(\mathbf{X}_i))}{\hat{\pi}(\mathbf{X}_i)} + \hat{\mu}_1(\mathbf{X}_i)\right) - \frac{1}{n}\sum_{i=1}^{n}\left(\frac{(1-D_i)(Y_i - \hat{\mu}_0(\mathbf{X}_i))}{1-\hat{\pi}(\mathbf{X}_i)} + \hat{\mu}_0(\mathbf{X}_i)\right)$$

In practice, AIPW weights may be very small or very large, a problem that inverse probability weights also suffer from. This can make AIPW prone to high variance. To remedy this, the predicted probabilities of treatment are often truncated, setting extremely small or large weights to some less extreme value (as in the `AIPW` R package, Zhong et al., 2021).

Below is R code to implement AIPW with truncation of extreme weights. As with all of the double robust methods reviewed here, we begin with predicted values (such as from a machine learning algorithm) for the outcome for treated units `mu1_pred` and untreated units `mu0_pred` as well as predicted values for treatment assignment probability `pi_pred`. We also have `d`, the vector of actual treatment assignments, and `y`, the observed outcome values.

```
require(tidyverse)


aipw_calc <- function(mu1_pred, mu0_pred, pi_pred, d, y){
  n <- length(mu1_pred)

  # Truncate extreme values of the weights
  pi_pred <- case_when(
    pi_pred < .01 ~ .01,
    pi_pred > .99 ~ .99,
    T ~ pi_pred)

  # Calculate the predicted outcome value for treated units
  y1_pred <- (d*(y-mu1_pred))/pi_pred + mu1_pred
```

```r
  # Calculate the predicted outcome value for untreated units
  y0_pred <- ((1-d)*(y-mu0_pred))/(1-pi_pred) + mu0_pred


  # Calculate the ATE
  ate <- (1/n)*(sum(y1_pred)) - (1/n)*sum(y0_pred)


  return(ate)
}
```

Glynn & Quinn (2010) provide an alternate but equivalent formula, where the basic inverse probability weight (IPW) estimator (which incorporates only the exposure model $\hat{\pi}$) is corrected using a weighted average of two outcome regression estimates:

$$\hat{\tau}_{AIPW} = \frac{1}{n}\sum_{i=1}^{n}\left\{\left[\frac{D_i Y_i}{\hat{\pi}(\mathbf{X}_i)} - \frac{(1-D_i)Y_i}{1-\hat{\pi}(\mathbf{X}_i)}\right] - \frac{D_i - \hat{\pi}(\mathbf{X}_i)}{\hat{\pi}(\mathbf{X}_i)(1-\hat{p}i(\mathbf{X}_i))}[(1-\hat{\pi}(\mathbf{X}_i))\hat{\mu}_1(\mathbf{X}_i) + \hat{\pi}(\mathbf{X}_i)\hat{\mu}_0(\mathbf{X}_i)]\right\}.$$

Targeted Maximum Likelihood Estimation (TMLE)

Extending and improving previous double robust methods, van der Laan & Rubin (2006) first proposed TMLE using a parametric framework and the efficient influence curve (Hines et al., 2022) to obtain estimates and standard errors. Mark van der Laan has gone on to collaborate on both a gentle introduction (Gruber & Laan, 2009), two textbooks (van der Laan & Rose, 2011; Van Der Laan & Rose, 2018), and an R package (Gruber & Laan, 2012) for implementing the method. Schuler & Rose (2017) and Luque-Fernandez et al. (2018) provide introductions for epidemiologists.

TMLE begins by estimating the relevant part of the data-generating distribution $P(Y)$, i.e. the conditional density $Q = P(Y \mid X)$. It next estimates the exposure model. Although any estimation method can be used for these steps, the originators of the method suggest using a "SuperLearner," i.e. ensemble learning with cross-validation (van der Laan et al.,

2007). Next, the exposure model is used to calculate a "clever covariate," which is similar to an IPW. The coefficient for this clever covariate is estimated using maximum likelihood – whence the "MLE" in "TMLE." Finally, the estimate of Q is updated in a function involving the clever covariate. This process can be iterated, but usually one iteration is enough. The estimate of the distribution Q can be used to calculate the estimand of interest.

Formally, first generate estimates of $\mu_d(\mathbf{X}_i) = E(Y \mid D = d, \mathbf{X}_i)$ and $\pi(\mathbf{X}_i) = P(D = 1 \mid \mathbf{X}_i)$. Next, calculate the clever covariates for each individual in the data. These quantities are similar to inverse probability weights, with H_{0i} for untreated and H_{1i} for treated units:

$$H_{0i}(D = 0, \mathbf{X} = \mathbf{x}_i) = \frac{1 - d_i}{1 - \hat{\pi}(\mathbf{x}_i)}, \quad H_{1i}(D = 1, \mathbf{X} = \mathbf{x}_i) = \frac{d_i}{\hat{\pi}(\mathbf{x}_i)}.$$

In the next step, we estimate fluctuation parameters $\epsilon = (\epsilon_0, \epsilon_1)$ through maximum likelihood of the following logistic regression with fixed intercept $\text{logit}(\mu_{di})$:

$$\text{logit}[E(Y = 1 \mid D, \mathbf{X})] = \text{logit}(\hat{\mu}_{di}) + \epsilon_0 H_{0i} + \epsilon_1 H_{1i}$$

Here we are assuming that Y is a dichotomous variable taking the values of 0 or 1; the method is extended to continuous outcomes simply by normalizing the value of Y to fall between 0 and 1. Then we generate updated ("targeted") estimates of potential outcomes:

$$\hat{\mu}_0^*(\mathbf{x}_i) = \text{expit}[\text{logit}(\hat{\mu}_0(\mathbf{x}_i)) + \hat{\epsilon} H_{0i}]$$
$$\hat{\mu}_1^*(\mathbf{x}_i) = \text{expit}[\text{logit}(\hat{\mu}_1(\mathbf{x}_i)) + \hat{\epsilon} H_{1i}]$$

where $\text{expit}(\cdot)$ is the inverse logit function.

Finally, we estimate the parameter of interest – in this case, the ATE:

$$\hat{\tau}_{TMLE} = \frac{1}{n} \sum_{i=1}^{n} [\hat{\mu}_1^*(\mathbf{x}_i) - \hat{\mu}_0^*(\mathbf{x}_i)]$$

Here is R code to implement TMLE, again with predicted outcome values `mu1_pred` and `mu0_pred` and predicted probability of treatment `pi_pred`. Since the outcome is bounded

and continuous, it is transformed to fall between 0 and 1 via $\tilde{Y}_i = [Y_i - \min(Y)]/[(\max(Y) - \min(Y)]$.

```
# Functions for normalization and de-normalization
normalize <- function(x, y){(x - min(y)) / (max(y) - min(y))}
denormalize <- function(x, y){x * (max(y) - min(y))}

tmle_calc <- function(mu1_pred, mu0_pred, pi_pred, d, y){
  # Normalize the outcome variable
  mu1_pred <- normalize(mu1_pred, y)
  mu0_pred <- normalize(mu0_pred, y)
  y_tilde <- normalize(y, y))

  n <- length(y)

  # Calculate clever covariates
  H0 = (1-d)/(1-pi_pred)
  H1 = d/pi_pred

  # Estimate fluctuation parameter through maximum likelihood estimation
  epsilon <- glm(y_tilde ~ -1 + H0 + H1 +
                 offset(qlogis((d==1)*mu1_pred + (d==0)*mu0_pred)),
                 family = binomial(link = 'logit')) %>%
    tidy() %>%
    pull(estimate)


  # Targeted estimates of the potential outcomes
  target_0 <- plogis(qlogis(mu0_pred + epsilon[1]*H0))
  target_1 <- plogis(qlogis(mu1_pred + epsilon[2]*H1))
```

```
  # Estimate ATE
  ATE <- mean(target_1 - target_0)
  return(denormalize(ATE, y))
}
```

Double/Debiased Machine Learning (DML)

The most recently developed of the methods reviewed here, DML was proposed in an econometrics context (Chernozhukov et al., 2018) and has since seen a flurry of development (Chernozhukov et al., 2022; Dukes et al., 2022; Farbmacher et al., 2022; Jung et al., 2021; Kennedy, 2023; Semenova & Chernozhukov, 2021). The R package `DoubleML` (Bach et al., 2021) provides straightforward implementation of the method.

DML is motivated by the need to handle problems with high-dimensional nuisance parameters, i.e. a large number of measured confounders. Flexible machine learning is appropriate for this task, but such methods suffer from regularization bias, where efforts to control the overfitting of models can bias estimates. DML removes this bias in a two-step procedure. First, it solves the auxiliary problem of estimating the treatment exposure model $E(D \mid X) = \pi(X)$. It then uses this model to remove bias: Neyman orthogonalization allows the creation of an orthogonalized regressor, essentially partialing out the effect of covariates X from treatment D. The debiased D is then used to estimate the conditional mean of the outcome $E(Y \mid X) = \mu(X)$, which can be used to calculate the estimand of interest.

More formally, suppose we want to estimate τ in the following framework:[1]

$$y_i = \tau d_i + g_0(\mathbf{x}_i) + u_i,$$

$$d_i = m_0(\mathbf{x}_i) + v_i.$$

[1]Note that this basic DML setup assumes a partially linear model and targets the ATE. If we are interested in the CATE and heterogeneous effects, Chernozhukov et al. (2018, p. C35) present an alternative score function that closely resembles AIPW. See also Jacob (2021) and Nie & Wager (2021).

The idea is to estimate g_0 and m_0 separately, then use an orthogonalized or debiased score function – here, residual-on-residual regression – to obtain an estimate of τ, which we can designate $\hat{\tau}$. However, this leaves a term in the asymptotic distribution of $\hat{\tau}$ that biases the estimate. To avoid this, DML uses sample splitting (Angrist & Krueger, 1995).

We randomly split the sample of n observations into two sets, I and I^c, each of size $n/2$.[2] Using any prediction algorithm, we then estimate the response and treatment models using only set I^c:

1) Estimate treatment model $\hat{m}_0$ in the equation $d_i = \hat{m}_0(\mathbf{x}_i) + \hat{v}_i, \forall i \in I^c$.

2) Estimate the outcome model $\hat{g}_0$ in the equation $y_i = \hat{g}_0(\mathbf{x}_i) + \hat{u}_i, \forall i \in I^c$.

Next, we use the estimated models to perform residual-on-residual regression *on the left out set* I to obtain an estimate of τ:

$$\hat{\tau}(I^c, I) = \left(\sum_{i \in I} \hat{v}_i d_i \right)^{-1} \sum_{i \in I} \hat{v}_i (y_i - \hat{g}_0(\mathbf{x}_i)),$$

where $\hat{v}_i = d_i - \hat{m}_0(\mathbf{x}_i)$. Using half the sample results in efficiency loss. To rectify this, we repeat the above procedure, switching the split sets. We then have $\hat{\tau}(I^c, I)$ and $\hat{\tau}(I, I^c)$. The cross-fitting DML estimator is:

$$\hat{\tau}_{DML} = \frac{\hat{\tau}(I^c, I) + \hat{\tau}(I, I^c)}{2}.$$

R code to implement DML is shown below. Since DML involves sample splitting, this code is a little different from the above examples. We start with observed outcome values `y`, treatment assignment `d`, and covariate matrix `x`. First, a pre-processing function `dml_pre()` randomly splits the sample, outputting I and I^c sets of each of these variables. The second step predicts outcome values and treatment probabilities for each half of the sample, using models fit to the other half. In the code chunk here, generalized random forests from the

[2] In practice, we can split the sample into any number of folds, and more than two sets might be better.

grf package are used to predict these, but any prediction algorithm can be used. Finally, a post-prediction function dml_post() performs the residual-on-residual regression for each half of the sample and finds the average of the two estimates to produce an ATE estimate.

```
# Pre-processing: sample splitting
dml_pre <- function(y, d, x, seed = 1758){
  set.seed(seed)

  n <- length(y)
  n_2 <- round(n/2)

  # Split the sample
  random_vec <- sample(1:n, n, replace = F)
  I <- random_vec[1:n_2]
  I_c <- random_vec[(n_2+1):n]

  return(list(
    y_I = y[I],
    d_I = d[I],
    x_I = x[I,],
    y_I_c = y[I_c],
    d_I_c = d[I_c],
    x_I_c = x[I_c,]
    ))
}

# Predictor function: in this case, generalized random forests
grf_dml <- function(y_I, d_I, x_I, y_I_c, d_I_c, x_I_c){
  # Train on the I_c sample, predict on the I sample
  mu_mod1 <- grf::regression_forest(X = x_I_c, Y = y_I_c,
```

```
  tune.parameters = "all")
  mu_pred1 <- predict(mu_mod1, newdata = x_I)$predictions

  pi_mod1 <- grf::regression_forest(X = x_I_c, Y = d_I_c,
  tune.parameters = "all")
  pi_pred1 <- predict(pi_mod1, newdata = x_I)$predictions

  # Train on the I sample, predict on the I_c sample
  mu_mod2 <- grf::regression_forest(X = x_I, Y = y_I,
  tune.parameters = "all")
  mu_pred2 <- predict(mu_mod2, newdata = x_I_c)$predictions

  pi_mod2 <- grf::regression_forest(X = x_I, Y = d_I,
  tune.parameters = "all")
  pi_pred2 <- predict(pi_mod2, newdata = x_I_c)$predictions

  return(list(
    mu_pred1 = mu_pred1,
    pi_pred1 = pi_pred1,
    mu_pred2 = mu_pred2,
    pi_pred2 = pi_pred2
  ))
}

# Implement DML: takes outputs from pre_dml() and grf_dml()
dml_post <- function(y_I, d_I, x_I, y_I_c, d_I_c, x_I_c,
                     mu_pred1, pi_pred1, mu_pred2, pi_pred2){

  # Residual-on-residual regression for each sample separately
```

```
  v1 <- d_I - pi_pred1
  delta1 <- (sum(v1 * d_I))^-1 * sum(v1 * (y_I - pi_pred1))

  v2 <- d_I_c - pi_pred2
  delta2 <- (sum(v2 * d_I_c))^-1 * sum(v2 * (y_I_c - pi_pred2))

  # Average estimates from each sample
  ate <- (delta1 + delta2)/2

  return(ate)
}

dml_pre_out <- dml_pre(y = y, d = d, x = x)
grf_dml_out <- do.call(grf_dml, dml_pre_out)
dml_post_out <- do.call(dml_post, append(dml_pre_out, grf_dml_out))
```

A simple demonstration using AIPW

To demonstrate double robustness, this section presents one of the simpler double robust estimators, AIPW. As shown above, we can write this estimator as follows:

$$\hat{\tau} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{D_i(Y_i - \hat{\mu}_1(X_i))}{\hat{\pi}(X_i)} + \hat{\mu}_1(X_i)\right) - \frac{1}{N}\sum_{i=1}^{N}\left(\frac{(1-D_i)(Y_i - \hat{\mu}_0(X_i))}{1-\hat{\pi}(X_i)} + \hat{\mu}_0(X_i)\right)$$

For each individual in the sample, this estimator calculates two quantities:

- The treated potential outcome

$$\hat{Y}_{1i} = \frac{D_i(Y_i - \hat{\mu}_1(X_i))}{\hat{\pi}(X_i)} + \hat{\mu}_1(X_i) \tag{3}$$

- The control potential outcome

$$\hat{Y}_{0i} = \frac{(1 - D_i)(Y_i - \hat{\mu}_0(X_i))}{1 - \hat{\pi}(X_i)} + \hat{\mu}_0(X_i) \tag{4}$$

Let's focus on the treated model, Equation (3). First, assume that the outcome model $\mu_1(X_i)$ is *correctly* specified and the exposure model $\pi(X_i)$ is *incorrectly* specified. Let's also assume (for now) that we're dealing with a treated unit, i.e. $D_i = 1$. Then

$$E[\hat{\mu}_1(X_i)] = E[Y_1 \mid X_i].$$

This means that

$$E[Y_i - \hat{\mu}_1(X_i)] = 0,$$

and hence

$$E[\hat{Y}_{1i}] = 0 + \hat{\mu}_1(X_i) = \hat{\mu}_1(X_i).$$

So the model relies *only* on the outcome model. The incorrectly specified exposure model completely disappears from the equation. If we're dealing with a control unit ($D_i = 0$), we get the same result:

$$\hat{Y}_{1i} = \frac{0(Y_i - \hat{\mu}_1(X_i))}{\hat{\pi}(X_i)} + \hat{\mu}_1(X_i) = \hat{\mu}_1(X_i).$$

Now, what if the *exposure* model $\pi(X_i)$ is correctly specified and the outcome model $\mu_1(X)$ is incorrect? First, we rewrite the estimator for the treated outcome:

$$\begin{aligned}
\hat{Y}_{1i} &= \frac{D_i(Y_i - \hat{\mu}_1(X_i))}{\hat{\pi}(X_i)} + \hat{\mu}_1(X_i) \\
&= \frac{D_i Y_i}{\hat{\pi}(X_i)} - \frac{D_i \hat{\mu}_1(X_i)}{\hat{\pi}(X_i)} + \frac{\hat{\pi}(X_i)\hat{\mu}_1(X_i)}{\hat{\pi}(X_i)} \\
&= \frac{D_i Y_i}{\hat{\pi}(X_i)} - \left(\frac{D_i - \hat{\pi}(X_i)}{\hat{\pi}(X_i)}\right) \hat{\mu}_1(X_i).
\end{aligned} \tag{5}$$

Since the exposure model is correctly specified, we have $D_i = \hat{\pi}(X_i)$ on average, so

$$E[D_i - \hat{\pi}(X_i)] = 0.$$

This means that the second term in Equation (5) is 0, so

$$E[\hat{Y}_{1i}] = E\left[\frac{D_i Y_i}{\hat{\pi}(X_i)}\right].$$

This shows that when the exposure model is correct, then the estimator depends *only* on the exposure model. We can make similar arguments for the control model for $\hat{Y}_{0i}$ in Equation (4).

This demonstration shows that this estimator achieves double robustness: The estimator is robust to misspecification of either the exposure or the outcome model (but not both). The other two double robust methods considered in this thesis can be shown to have the same property, but proving so is more complicated.

Evaluation Strategy

These double robust methods have many similarities. How do the results they give compare? This section tests the performance of each in practice using two strategies. First, results are compared using simulated data from a causal inference competition (Dorie et al., 2019). The true treatment effect is known and all potential confounders are observed, so these simulations allow assessment of bias and related quantities. Second, these methods are applied to data from LaLonde's (1986) study of the National Supported Work Demonstration (NSW). The NSW randomly provided training to disadvantaged workers, allowing an experimental estimate of the effect of the intervention, and data assembled by Dehejia & Wahba (1999) compares these experimental estimates to observational ones.

The three double robust methods are compared to two sets of traditional or "single robust" methods used as benchmarks (see Table 1 for an overview of all methods used). By "single robust," I mean methods that are not robust to any misspecification. First are one-model

Table 1: Double and single robust methods used for evaluation

Type	Models	Method	Label	Estimators
Single Robust	One	Linear Regression	ols	OLS
Single Robust	One	Propensity Score Matching	psm	logit
Single Robust	One	Inverse Probability Weights	ipw	logit, generalized random forests, super learner
Single Robust	Two	G-Computation	g-comp	generalized random forests, super learner
Single Robust	Two	Lin Estimator	lin	OLS
Double Robust	Two	Augmented Inverse Probability Weights	aipw	OLS/logit, generalized random forests, super learner
Double Robust	Two	Targeted Maximum Likelihood Estimation	tmle	OLS/logit, generalized random forests, super learner
Double Robust	Two	Double/Debiased Machine Learning	dml	OLS/logit, generalized random forests, super learner

methods. The most classic method considered – linear regression – models only the response surface. It is estimated using ordinary least squares regression ("OLS"), entering each variable separately without any interactions or higher-order terms. Two other one-model methods model only the treatment assignment mechanism. In propensity score matching (PSM), propensity scores are estimated from logistic regression with each variable entered separately and without any higher order terms, then matched using the `MatchIt` package. Finally, stabilized inverse probability weights (IPW, Austin & Stuart, 2015, p. 3663) are used for weighted OLS regression. These IPWs are estimated using propensity scores estimated by

each of the three under-the-hood estimation techniques described below; extreme propensity scores are truncated so that they range from 0.01 to 0.99.

The second set of single robust methods are two-model methods, which estimate separate models for treated and untreated units. In theory these could solve the misspecification problem that double robust methods are meant to solve, but they could still suffer from overfitting. G-computation (Robins, 1986; Snowden et al., 2011) uses some estimation technique to predict outcomes under treatment and control for each unit in the dataset. The ATE estimate is the difference in the average prediction under treatment and the average prediction under control. The second two-model method is the Lin estimator (Lin, 2013). This method aims to solve issues with the bias induced by OLS regression in a randomization framework by interacting the treatment indicator with mean-centered covariates. Hazlett & Shinkre (2024) show that this method is equivalent to estimating two separate OLS regression models for the treated and control units.

Because many of these methods allow the user to choose the underlying estimation method, results compare three techniques. The first technique uses a logistic regression for the exposure model and an OLS regression for the outcome model. Second is generalized random forests (GRF, Athey et al., 2019) using the `grf` R package, with separate models for exposure and outcome. The final technique is the SuperLearner (as promoted by the makers of TMLE) using the `SuperLearner` package (Polley et al., 2023), again with separate models for exposure and outcome. GLM, glmnet (a weighted average of lasso and ridge regression, Friedman et al., 2021), and XGBoost (Chen & Guestrin, 2016) models are considered for the SuperLearner. These three estimation techniques are used for each of the three double robust methods and for IPW. GRF and SuperLearner are also used for G-computation (OLS predictions with G-computation return identical results to OLS regression coefficient estimates).

For the simulations, results compare only point estimates from these methods. For evaluation of the experimental LaLonde data, I use bootstrapping with 100 samples to obtain standard errors and confidence intervals. Using AIPW as written above results in some wildly biased estimates, due to dividing by some very small propensity scores. Hence I present

estimates from a truncated AIPW estimator, where predicted exposure model values are set to 0.01 if they are less than 0.01 and to 0.99 if they are greater than 0.99.

Simluations with Dorie et al. (2019) Data

In 2016, the Atlantic Causal Inference Conference hosted a competition for causal inference methods that adjust on observables. Dorie et al. (2019) published the results of this competition, along with the data used in the competition. Below, I test double robust methods on the 20 data sets used for the "do-it-yourself" part of the competition. The data represent a hypothetical twins study investigating the impact of birth weight on IQ. The data have 4,802 observations and 52 covariates. The authors of the study specify a different data generating process for the potential outcomes in each data set. In all cases, ignorability holds (all potential confounders are observed), but the authors vary the following:

- degree of nonlinearity
- percentage of treated
- overlap for the treatment group
- alignment (correspondence in variables used to generate the assignment mechanism and the response surface)
- treatment effect heterogeneity

The true treatment effect also varies, but as a function of the other DGP characteristics. It has a mean of 3.6, standard deviation of 1.6, and range of -1.7 to 12.

Main results

The 20 data sets used here cover a range of these attributes; see the supplemental material from Dorie et al. (2019) for details. I use 10 simulations of each data set, resulting in 200 data sets. I then calculate bias, percent bias (the estimator's bias as a percentage of its standard error), root mean squared error (rmse), and median absolute error (mae). I also present the number of datasets for which the method fails and the median computation time

for each data set, in seconds.[3] In the main text I present average bias and RMSE, while the appendix contains tables with full results (Table 5).

Bias results for the full range of simulations are shown in Figure 1. Bias is quite low for many of the methods, however IPW (logit) and AIPW (OLS/logit) have high bias and variance, while TMLE (OLS/logit) has moderately high bias and variance. With an average true treatment effect of 3.6, bias with absolute value greater than 1 is substantial. The traditional methods, however, achieve fairly low bias in general.

Figure 2 orders methods by RMSE and presents both bias and RMSE. The lowest RMSE is achieved by three of the methods using SuperLearner estimators (AIPW, TMLE, and G-computation) with values of about 0.35, followed closely by the same three methods using GRF, with values closer to 0.5. The computationally efficient Lin estimator does not do much worse, with an RMSE of 0.6, and OLS and PSM achieve acceptable RMSE of 0.7 to 0.9. Interestingly, DML with the computationally efficient OLS/logit estimators achieves lower RMSE than with GRF or SuperLearner (0.8 compared to 0.9 and 1.6, respectively). The only estimators with RMSE that exceed 2 are TMLE (OLS/logit) with 2.3, IPW (logit) with 8.1, and AIPW (OLS/logit) with 10.1. These methods all use logit models to estimate probability of treatment, and these high error rates are likely due to extreme values of these estimates.

Overall, traditional methods perform surprisingly well in comparison with the double robust methods, and flexible single robust methods may be as effective as double robust methods. Even in the full range of datasets – which include highly nonlinear exposure and outcome data-generating processes – OLS, propensity score matching, and the Lin estimator obtain some of the smallest bias and RMSE. While double robust methods achieve the lowest RMSE, the choice of underlying estimator appears more important than the choice of method. AIPW and TMLE both do well with a flexible underlying estimator, while DML does worse than OLS. Of the estimators considered, the SuperLearner (which considers GLM, glmnet, and XGBoost models) appears to be best for the double robust methods, with GRF following closely. G-computation does only a hair worse than these two double robust

[3]Simulations were run on a 2020 MacBook Pro laptop computer with a 2 GHz Quad-Core Intel Core i5 Processor and 16 GB of memory.

methods without explicitly accounting for regularization bias. Notably, the method with the longest computation time – DML with a SuperLearner – takes nearly 2,000 times as long as OLS (an average of 129 seconds per simulation compared to 0.061 seconds).

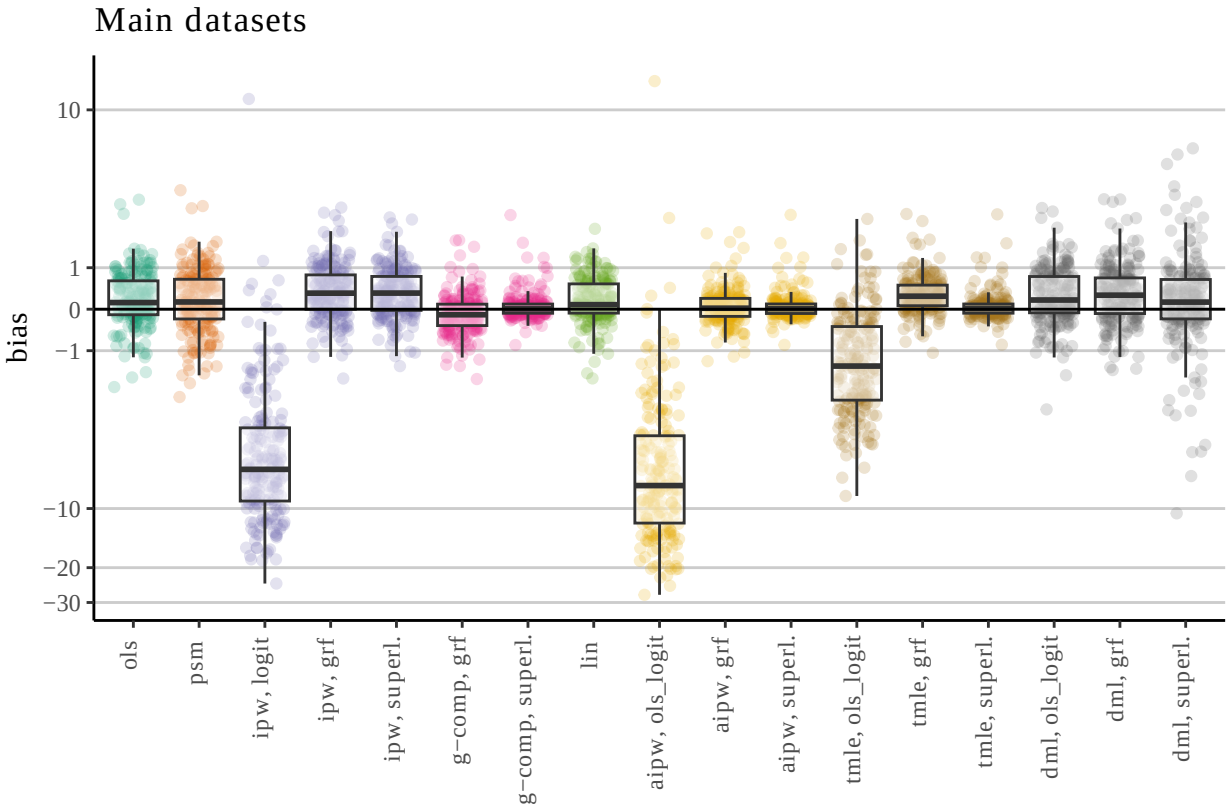

Figure 1: Bias of Monte Carlo simulations using the first 20 datasets from Dorie et al. (2019), 10 replications each.

Linear DGPs

Due to their functional form assumptions, traditional methods may perform better when the data generating processes are linear. To test this, I use 100 simulations of each of the two datasets from Dorie et al. (2019) with linear data generating processes for both exposure and outcome (numbers 1 and 3). In these two datasets, the average true treatment effect is 3.9 with a standard deviation of 1.5.

Figure 3 shows the bias from each simulation for each method (in the Appendix, Table 6 shows the full results). Similarly to the full set of simulations, bias is fairly low for most methods. IPW (logit) and AIPW (OLS/logit) again suffer from greater bias than other

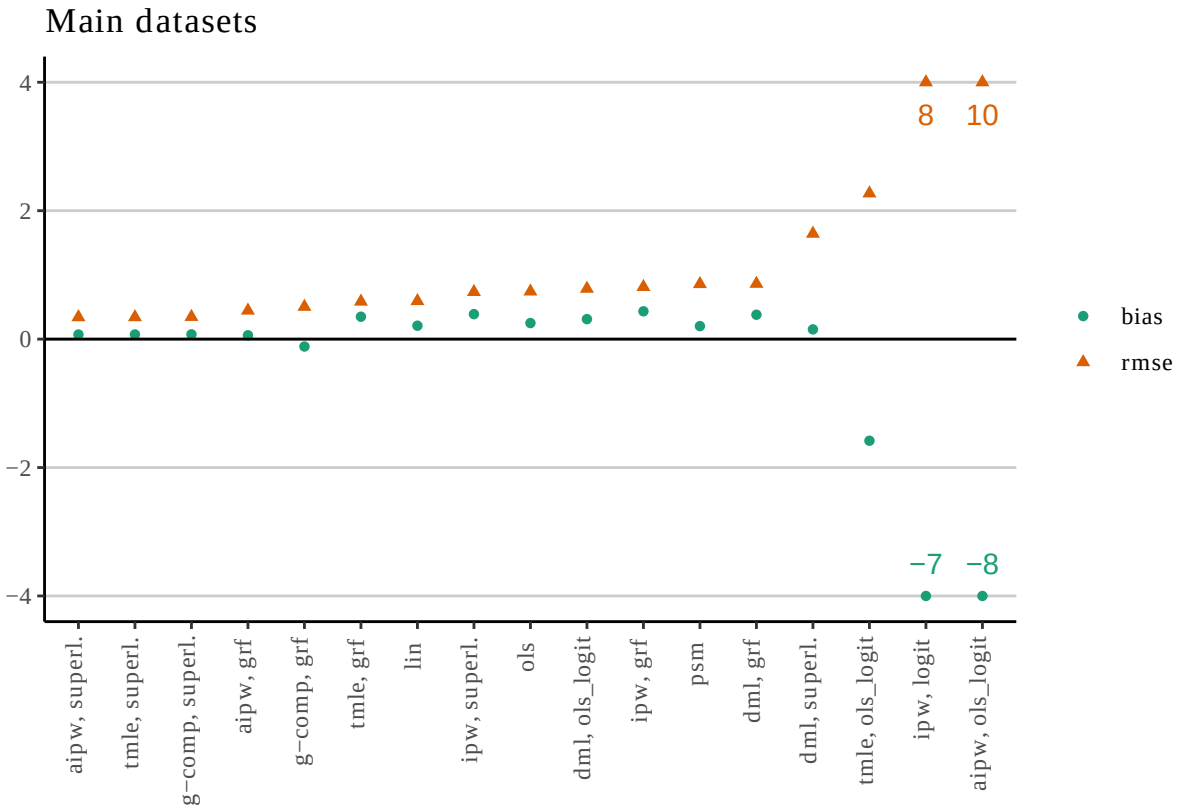

Figure 2: Root mean squared error and bias for Monte Carlo simulations using the first 20 datasets from Dorie et al. (2019), 10 replications each. Values greater in absolute value than 4 are plotted at 4 and labeled with their actual value.

methods, though not to quite as extent as in the full range of simulations. Unsurprisingly, methods that assume linearity – OLS, PSM, the Lin model – achieve low bias and variance. IPW does less well than expected, even when treatment assignment is modeled with flexible GRF and SuperLearner.

Figure 4 orders the methods by RMSE and presents both RMSE and bias. The methods achieving the lowest RMSE are again three of the SuperLearner methods (TMLE, AIPW, and G-computation) followed by the Lin estimator, DML (OLS/logit), and OLS. The only methods with RMSE above 1 are IPW (logit) and AIPW (OLS/logit), again likely due to unstable logit predictions.

With these linear DGPs, there seems little reason to sacrifice computational efficiency for a very slight reduction in RMSE. The Lin estimator has an RMSE of 0.28 compared to the lowest-RMSE method, TMLE (SuperLearner), of 0.25, and computes in 0.15 seconds compared to the latter's 126 seconds. Even standard OLS has quite a low RMSE of 0.39. While DML performs better with the linear DGPs than the full range of simulations, it still obtains higher RMSE than at least certain AIPW and TMLE variants. Finally, G-computation again shows its strength, outperforming most other methods when it is estimated using a SuperLearner.

Do results vary by DGP?

While the AIPW, TMLE, and G-computation with a SuperLearner may be the top performing methods overall, this does not mean that there are some data-generating processes (DGPs) where some other method may do better. Across the the 20 datasets, Dorie et al. (2019) vary six DGP characteristics: degree of nonlinearity, the percentage treated, overlap for the treatment group, alignment (correspondence in variables used to generate the exposure and response models), and treatment effect heterogeneity. To test how the methods perform across different values of these characteristics, I begin with the 200 simulations from the 20 datasets used in the main results (Figures 1 and 2, Table 5). I limit datasets to those generated by a particular value of a DGP characteristic, and I then calculate RMSE for each

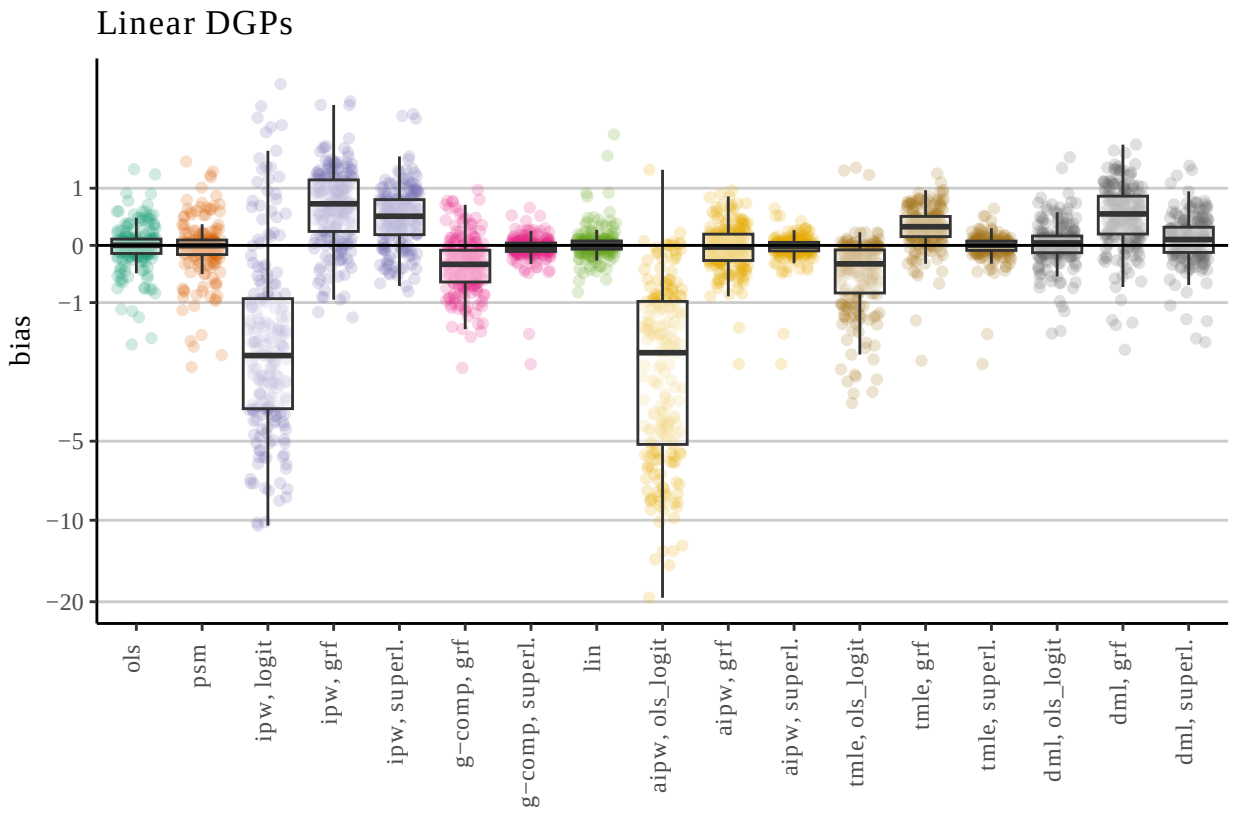

Figure 3: Bias of Monte Carlo simulations using the two datasets from Dorie et al. (2019), with linear data generating processes, 100 replications each ("linear").

Table 2: Data generating process: Three lowest RMSE methods by DGP for Monte Carlo simulations using the first 20 datasets from Dorie et al. (2019), 10 replications each.

DGP parameter	dgp_value	Lowest RMSE	Second-lowest	Third-lowest
Treat. assign.	linear	aipw, superl.: 0.303	tmle, superl.: 0.304	g-comp, superl.: 0.309
Treat. assign.	polynomial	aipw, superl.: 0.362	tmle, superl.: 0.363	g-comp, superl.: 0.366
Treat. assign.	step	tmle, superl.: 0.323	aipw, superl.: 0.328	g-comp, superl.: 0.33
Prob. of treat.	0.35	tmle, superl.: 0.282	aipw, superl.: 0.284	g-comp, superl.: 0.292
Prob. of treat.	0.65	aipw, superl.: 0.414	g-comp, superl.: 0.414	tmle, superl.: 0.416
Overlap	full	g-comp, superl.: 0.124	aipw, superl.: 0.127	tmle, superl.: 0.132
Overlap	one-term	aipw, superl.: 0.35	tmle, superl.: 0.35	g-comp, superl.: 0.354
Response surface	exponential	aipw, superl.: 0.307	tmle, superl.: 0.307	g-comp, superl.: 0.315
Response surface	linear	g-comp, superl.: 0.406	aipw, superl.: 0.406	tmle, superl.: 0.408
Response surface	step	tmle, superl.: 0.323	aipw, superl.: 0.328	g-comp, superl.: 0.33
Alignment	0	g-comp, superl.: 0.104	aipw, superl.: 0.106	tmle, superl.: 0.11
Alignment	0.25	aipw, superl.: 0.493	tmle, superl.: 0.494	g-comp, superl.: 0.495
Alignment	0.75	tmle, superl.: 0.267	aipw, superl.: 0.267	g-comp, superl.: 0.274
Treat. heterogeneity	high	tmle, superl.: 0.355	aipw, superl.: 0.356	g-comp, superl.: 0.36
Treat. heterogeneity	none	g-comp, superl.: 0.17	aipw, superl.: 0.176	tmle, superl.: 0.193

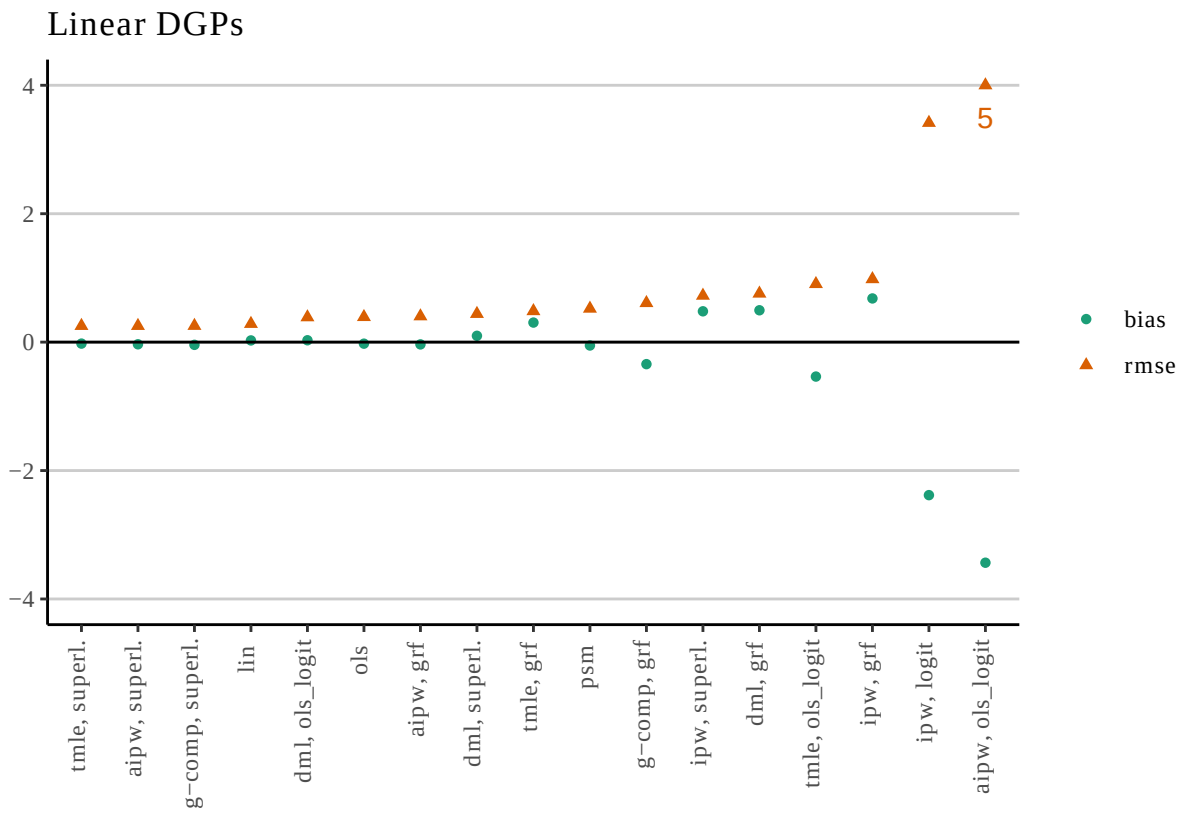

Figure 4: Root mean squared error and bias for Monte Carlo simulations using the two datasets from Dorie et al. (2019), with linear data generating processes, 100 replications each ("linear"). Values greater in absolute value than 4 are plotted at 4 and labeled with their actual value.

Table 3: Data generating process: Fourth- to sixth-lowest RMSE methods by DGP for Monte Carlo simulations using the first 20 datasets from Dorie et al. (2019), 10 replications each.

DGP parameter	dgp_value	Fourth-lowest	Fifth-lowest	Sixth-lowest
Treat. assign.	linear	aipw, grf: 0.457	g-comp, grf: 0.529	lin: 0.534
Treat. assign.	polynomial	aipw, grf: 0.439	g-comp, grf: 0.495	lin: 0.583
Treat. assign.	step	aipw, grf: 0.446	g-comp, grf: 0.481	tmle, grf: 0.487
Prob. of treat.	0.35	aipw, grf: 0.419	g-comp, grf: 0.508	tmle, grf: 0.555
Prob. of treat.	0.65	aipw, grf: 0.481	g-comp, grf: 0.499	lin: 0.566
Overlap	full	g-comp, grf: 0.242	aipw, grf: 0.272	ols: 0.51
Overlap	one-term	aipw, grf: 0.452	g-comp, grf: 0.514	tmle, grf: 0.583
Response surface	exponential	aipw, grf: 0.418	g-comp, grf: 0.487	tmle, grf: 0.534
Response surface	linear	aipw, grf: 0.494	g-comp, grf: 0.544	lin: 0.588
Response surface	step	aipw, grf: 0.446	g-comp, grf: 0.481	tmle, grf: 0.487
Alignment	0	ipw, superl.: 0.194	ols: 0.228	lin: 0.234
Alignment	0.25	lin: 0.512	aipw, grf: 0.566	g-comp, grf: 0.588
Alignment	0.75	aipw, grf: 0.395	g-comp, grf: 0.472	tmle, grf: 0.561
Treat. heterogeneity	high	aipw, grf: 0.458	g-comp, grf: 0.505	tmle, grf: 0.583
Treat. heterogeneity	none	aipw, grf: 0.297	psm: 0.365	ols: 0.378

method for only these datasets. For example, alignment is 0 for datasets 8 and 16 (meaning there is 0 correlation between the terms included in the treatment and outcome models), so for this value of the DGP RMSE is calculated only for those two datasets.

Tables 2 and 3 shows the six top-performing methods by lowest RMSE for each each value of the DGP characteristics. Across DGPs, the same methods dominate as in the full range of simulations: the SuperLearner with AIPW, TMLE, and G-compuation. In fact, these three methods take the top three spots for *every* DGP variation in the simulations.

Table 3 shows the fourth- to sixth-lowest RMSE methods for each DGP variation. GRF with AIPW, G-computation, and TMLE take most of these spots, but the Lin estimator and OLS appear a few times.

Overall, there is little variation across different types of DGPs in which method performs the best. Flexible estimators take the top spots (though notably not those used with DML), and traditional methods do fairly well across the board.

Do results vary by sample size?

In the above simulations, double robust methods have only slightly outperformed traditional methods. Is the issue with previous results simply that the sample size of the simulation data ($n = 4,802$) is too small for double robust methods to seriously outperform traditional methods? The double robust methods reviewed here have been shown to have lower bias asymptotically, so perhaps their superiority to traditional or single robust methods will be starker in larger samples. To test this, this section uses simulated datasets of varying sizes, from 150 to 96,040 (20 times the original sample size). These datasets are also derived from the Dorie et al. (2019) `aciccomp2016` package, using parameter set 7, a fairly nonlinear DGP with high heterogeneity. For sample sizes less than 4,802, the sample is randomly drawn from a randomly generated 4,802-unit sample. For sample sizes greater than 4,802, the design matrix (but not the outcome variable) is duplicated, then fed into the `dgp_2016` function. This preserves covariate distributions but retains stochasticity in the outcome.

Table 4 presents the four methods with the lowest RMSE for each sample size (Table 7

Table 4: Sample size: Four lowest RMSE methods by sample size for Monte Carlo simulations using dataset 7 from Dorie et al. (2019), 20 replications each

size	Lowest	Second-lowest	Third-lowest	Fourth-lowest
150	dml, grf: 1.09	ipw, superl.: 1.18	tmle, superl.: 1.234	ipw, grf: 1.237
300	aipw, superl.: 0.852	g-comp, superl.: 0.89	tmle, superl.: 0.901	aipw, grf: 0.981
600	g-comp, superl.: 0.746	aipw, superl.: 0.767	aipw, grf: 0.809	tmle, superl.: 0.81
1200	psm: 0.548	g-comp, superl.: 0.602	aipw, superl.: 0.619	aipw, grf: 0.655
2400	g-comp, superl.: 0.378	aipw, superl.: 0.381	tmle, superl.: 0.401	aipw, grf: 0.496
4802	aipw, superl.: 0.272	g-comp, superl.: 0.278	tmle, superl.: 0.288	tmle, grf: 0.395
9604	tmle, superl.: 0.26	aipw, superl.: 0.281	aipw, grf: 0.288	g-comp, superl.: 0.327
24010	g-comp, superl.: 0.137	tmle, superl.: 0.137	aipw, superl.: 0.14	aipw, grf: 0.245
48020	tmle, superl.: 0.139	tmle, grf: 0.16	aipw, superl.: 0.199	aipw, grf: 0.205
96040	aipw, superl.: 0.16	g-comp, superl.: 0.215	tmle, grf: 0.229	aipw, grf: 0.242

in the Appendix presents full results). Again, AIPW, TMLE, and G-computation methods that incorporate the SuperLearner or GRF dominate. There are two exceptions: In the tiny sample of 150, IPW (SuperLearner) and IPW (GRF) figure into the best four methods. PSM is the best-performing method in samples of 1200, but it fails to estimate 17 of the 20 datasets. (With 58 covariates, some of the methods fail in smaller samples.)

Figure 5 presents RMSE for all sample sizes for all methods. In the smallest samples, GRF and SuperLearner are able to provide estimates, while traditional methods fail. Beginning at 1,200 observations, all methods are able to provide estimates for all datasets, with the exception of PSM, which fails to calculate even in some large samples. For most methods, RMSE decreases nearly monotonically as sample size grows. Two exceptions are IPW (logit) and AIPW (OLS/logit), whose error is highest in the maximum sample of 96,040, likely due to extreme logit estimates. In addition, the rank order of methods is nearly constant across sample sizes. DML does not achieve lower RMSE than OLS, PSM, or the Lin estimator in most sample sizes, and methods using the SuperLearner or GRF perform the best across sample sizes.

In sum, although methods incorporating SuperLearner or GRF do the best in most sample sizes, traditional methods still perform fairly well, achieving low RMSE once the sample size is large enough for them to stably compute.

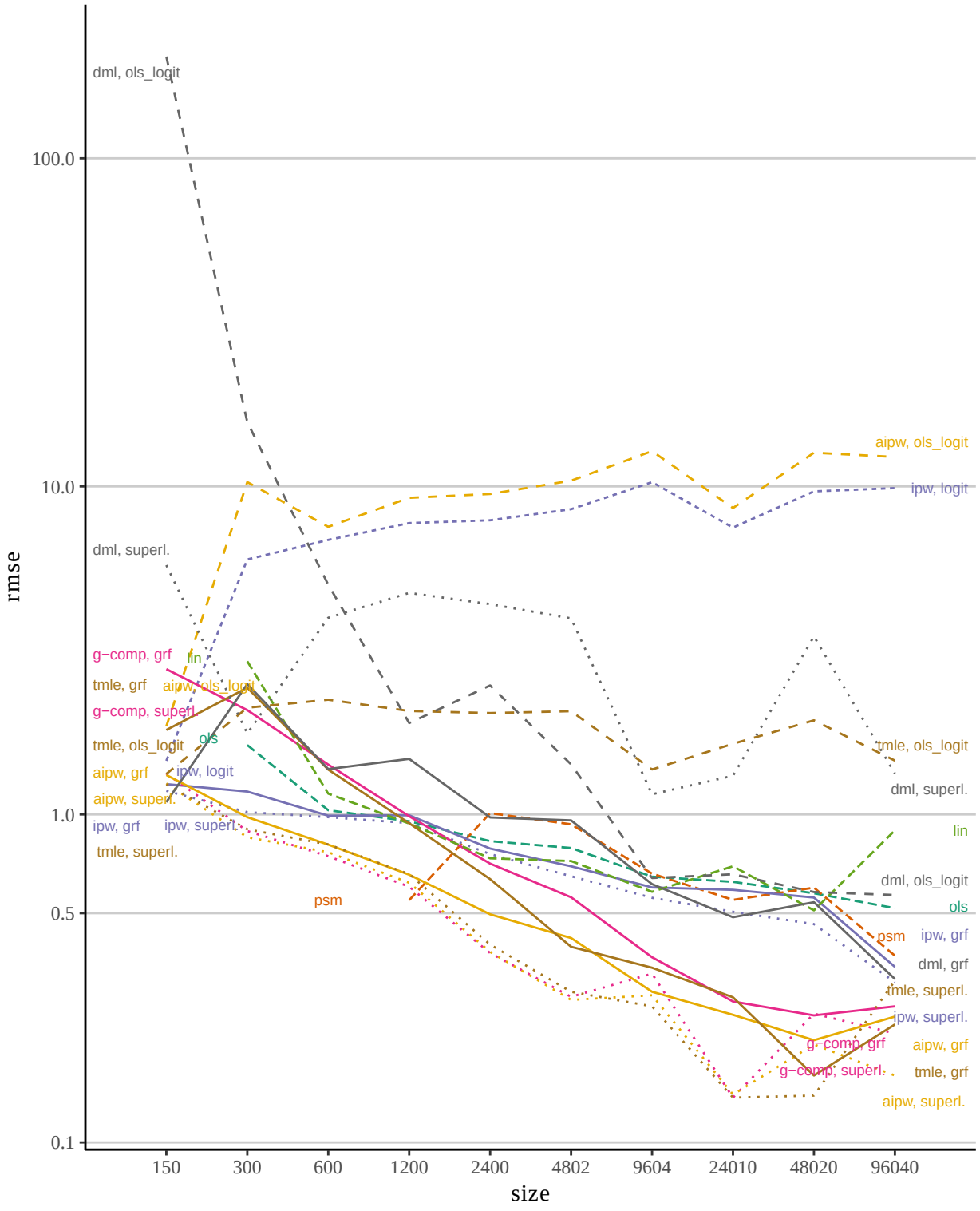

Figure 5: RMSE of Monte Carlo simulations using dataset 7 from Dorie et al. (2019) with varying sample sizes, 20 replications each

LaLonde NSW Data

As another evaluation of these methods, I use data from LaLonde's (1986) study of the National Supported Work Demonstration (NSW), as provided by Dehejia & Wahba (1999). Between March 1975 and July 1977, the NSW randomly provided training to disadvantaged workers. LaLonde used earnings in 1978 as the outcome of interest; comparing earnings in this year for treated and untreated workers allows an experimental estimate of the effect of the intervention. Restricting the sample to men, this study had 297 treated and 425 control participants. Covariates include age, education in years of schooling, earnings in 1975, and dichotomous variables for Black and Hispanic race, married, and not having a high school degree. Following Dehejia & Wahba (1999), I add a variable indicating whether each respondent's earnings in 1975 was \$0 – i.e., they were unemployed.

LaLonde compared these experimental estimates to control samples drawn from the Panel Study of Income Dynamics (PSID) and Westat's Matched Current Population Survey-Social Security Administration File (CPS). The PSID-1 sample (n = 2,490) contains all male household heads under 55 who did not classify themselves as retired in 1975, and the PSID-3 sample (n = 128) further restricts this to men who were not working in the spring of 1976 or 1975. The CPS-1 sample (n = 15,992) includes all CPS males under 55, and CPS-3 (n = 429) restricts this two those who were not working in March 1976 whose earnings in 1975 were below the poverty level. Restricting these observational samples gets closer to the group eligible for the NSW.

Following Dehejia & Wahba (1999), I present results for the original samples analyzed by LaLonde (1986), but I also include results using a subsample of the experimental group that has 1974 earnings data available (185 treated and 260 control participants) and include this additional covariate, along with an indicator variable for no earnings in 1974.

Results are presented in Figure 6, with a table in the Appendix (Table 8). Standard errors are based on 100 bootstrap samples. We first focus on the original LaLonde dataset, which did not include 1974 earnings. The "experimental" estimates provide a baseline for the comparison, suggesting that the program resulted in an earnings gain of about \$800.

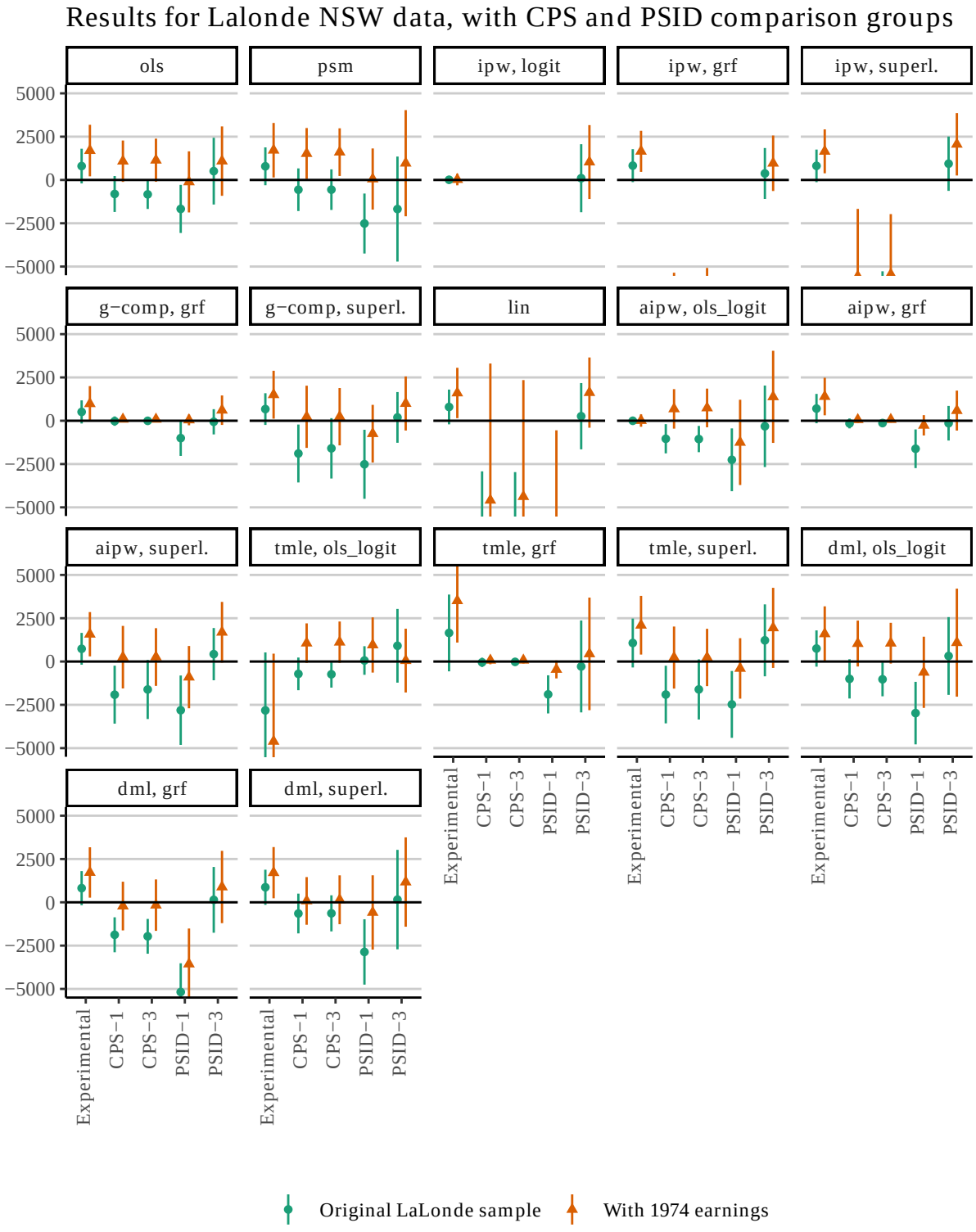

Figure 6: ATE estimates and 95-percent bootstrap standard error confidence intervals for Lalonde NSW data as provided by Dehejia and Wahba (1999), with CPS and PSID comparison groups. Standard errors shown in parentheses. Covariates include age, education in years of schooling, earnings in 1975, and dichotomous variables for Black and Hispanic race, married, not having a high school degree, and having no earnings in 1975. The "With 1974 earnings" estimates additionally include earnings in 1974 as a covariate, along with an indicator for having no earnings in 1974.

Some methods calculate widely different results for the experimental estimates, highlighting their instability. Echoing results from the simulations, methods that include logit models are particularly unstable.

If selection on observables holds, then we should be able to recover experimental estimates from the non-experimental control groups. Most of the methods do not perform very well, estimating treatment effects with the wrong sign. The exception is in the PSID-3 sample, where 12 of the 17 methods estimate treatment effects with the correct (positive) sign. This sample is chosen to be closer to the experimental sample.

Including 1974 earnings data results in much better estimates with the observational control groups. OLS, PSM, G-computation (SuperLearner), AIPW (SuperLearner), TMLE (SuperLearner), and DML (OLS/logit) compute fairly stable estimates across the samples. On the other hand, the estimates produced by IPW (logit), IPW (GRF), IPW (SuperLearner), the Lin estimator, and DML (GRF) vary widely across samples.

These results highlight the importance of selection on observables holding. Without including 1974 earnings as a covariate, it appears that selection on observables does not hold, as most methods provide highly inaccurate estimates with the wrong sign. Once 1974 earnings are included, most of the methods provide estimates much closer to the experimental values.

Conclusion

This thesis aims to provide an introduction to and evaluation of double robust methods for covariate adjustment in causal inference. By comparing AIPW, TMLE, and DML to more traditional statistical methods such as OLS and PSM as well as flexible "single robust" methods such as G-computation and the Lin estimator, it allows evaluation of whether these methods are worth the effort and (computational) time for social scientists to adopt them.

Results are nuanced. In the full range of simulated data, AIPW and TMLE with a SuperLearner or GRF are able to obtain smaller RMSE than OLS or PSM. However these

differences are quite small, and G-computation with the same flexible machine learning methods performs just as well as the double robust methods. DML does not perform as well as AIPW or TMLE (however, a version of DML that allows for heterogeneous effects might perform better; see Chernozhukov et al. (2018, p. C35)). The Lin estimator performs slightly better than OLS or PSM, without any increase in computation time. Methods relying on logit models to estimate propensity weights have the most error; researchers should use these methods with caution.

Methods that come out on top in the full range of simulations also tend to do the best regardless of the data generating process, though the Lin estimator and OLS rise in the rankings when the true treatment and outcome models are linear. AIPW, TMLE, and G-computation with a SuperLearner or GRF perform the best, followed closely by the Lin estimator, OLS, and PSM. As sample size varies, the same rank order generally holds, though traditional methods are unable to produce estimates in small samples when there are many covariates, while the double robust methods are able to do so.

Results from the experimental LaLonde data are more difficult to interpret. The sample appears much more important than the choice of method. In the original LaLonde data, most methods fail to estimate treatment effects in the observational data with the same sign as the experimental estimate. The exception is the PSID-3 sample, which includes only men who were not working in the spring of 1976 or 1975 and is thus more comparable to the individuals were recruited to the NSW study. When 1974 earnings are included as a covariate, many of the methods provide estimates across samples that are close to the experimental estimates. OLS, PSM, G-computation (SuperLearner), AIPW (SuperLearner), TMLE (SuperLearner), and DML (OLS/logit) appear the most reliable, while IPW (logit), IPW (GRF), IPW (SuperLearner), the Lin estimator, and DML (GRF) prove unstable. It is surprising that the Lin estimator provides unstable results in the NSW data, while it is among the lowest-RMSE methods in the simulation study. This highlights the complexity of doing causal inference with observational data where ignorability may not hold.

This thesis has a number of limitations. First, although it considers some of the most popular double robust, machine learning, traditional, and single robust methods, there are

many methods that it could not evaluate, including variations and extensions of the three methods. Second, although the simulations are meant to cover a wide range of data generating processes, they only consider continuous outcomes and binary treatments; simulations with binary or categorical outcomes and continuous or multi-armed treatments may yield different results. Finally, in considering only functional form misspecification, the simulations in this thesis do not consider situations where ignorability does not hold. In particular, it does not evaluate situations where causal identification is misspecified (Keil et al., 2018). Future research should assess violations of this and other assumptions underlying these methods.

In conclusion, if researchers want small gains in accuracy, they may opt for AIPW, TMLE, or G-computation with a flexible machine learning algorithm. But these methods are computationally costly, taking over two minutes per dataset of 4,802 observations and 52 covariates, while OLS, PSM, and the Lin estimator each take a fraction of a second. While double robust methods are useful for social scientists to understand, in most applications, OLS or PSM provide similar results. However, in certain circumstances these double robust methods may be a better choice. Especially when paired with a highly flexible estimator like a SuperLearner or GRF, these methods may be slightly more accurate, and they can be useful when the number of covariates is high or even exceeds the number of observations. In longitudinal settings with time-varying confounders, they may be more useful (Tran et al., 2019). They can also be useful as a sensitivity check; if researchers obtain similar estimates across traditional and double robust methods, they can be more confident in the reliability of their estimates.

Appendix

Table 5: Main datasets: Results of Monte Carlo simulations using the first 20 datasets from Dorie et al. (2019), 10 replications each. Percent bias is calculated as the estimator's bias as a percentage of its standard error, rmse is root mean squared error, mae is median absolute error, and comp_time is median computation time measured in seconds for each dataset.

method	estimator	bias	percent_bias	rmse	mae	comp_time	fail_count
ols	NA	0.250	0.157	0.74	0.41	0.061	0
psm	NA	0.203	0.131	0.86	0.53	0.668	6
ipw	logit	-6.690	-1.590	8.14	6.37	0.560	0
ipw	grf	0.433	0.266	0.81	0.46	32.227	0
ipw	superlearner	0.389	0.234	0.73	0.43	130.013	0
g-comp	grf	-0.115	-0.073	0.50	0.26	32.222	0
g-comp	superlearner	0.074	0.048	0.35	0.10	130.006	0
lin	NA	0.209	0.117	0.59	0.28	0.152	0
aipw	ols_logit	-8.240	-1.547	10.06	7.63	0.555	0
aipw	grf	0.060	0.039	0.45	0.22	32.223	0
aipw	superlearner	0.072	0.047	0.34	0.11	130.007	0
tmle	ols_logit	-1.583	-0.676	2.27	1.47	0.575	0
tmle	grf	0.349	0.230	0.58	0.33	32.241	0
tmle	superlearner	0.073	0.047	0.34	0.10	130.027	0
dml	ols_logit	0.311	0.192	0.79	0.42	0.665	0
dml	grf	0.380	0.247	0.86	0.51	31.574	0
dml	superlearner	0.152	0.068	1.64	0.46	129.160	0

Table 6: Linear datasets: Results of Monte Carlo simulations using the two datasets from Dorie et al. (2019), with linear data generating processes, 100 replications each ("linear"). Percent bias is calculated as the estimator's bias as a percentage of its standard error, rmse is root mean squared error, mae is median absolute error, and comp_time is median computation time measured in seconds for each dataset.

method	estimator	bias	percent_bias	rmse	mae	comp_time	fail_count
ols	NA	-0.024	-0.017	0.39	0.123	0.058	0
psm	NA	-0.053	-0.039	0.52	0.125	0.570	12
ipw	logit	-2.383	-0.897	3.42	2.256	0.596	0
ipw	grf	0.680	0.428	0.98	0.763	30.264	0
ipw	superlearner	0.480	0.309	0.72	0.517	126.091	0
g-comp	grf	-0.343	-0.219	0.61	0.390	30.259	0
g-comp	superlearner	-0.042	-0.029	0.25	0.077	126.085	0
lin	NA	0.027	0.017	0.28	0.072	0.155	0
aipw	ols_logit	-3.435	-1.012	4.75	2.062	0.592	0
aipw	grf	-0.035	-0.024	0.40	0.228	30.261	0
aipw	superlearner	-0.034	-0.023	0.25	0.076	126.086	0
tmle	ols_logit	-0.536	-0.339	0.91	0.329	0.609	0
tmle	grf	0.305	0.220	0.48	0.331	30.276	0
tmle	superlearner	-0.023	-0.016	0.25	0.077	126.102	0
dml	ols_logit	0.029	0.020	0.39	0.140	0.679	0
dml	grf	0.496	0.336	0.76	0.576	28.689	0
dml	superlearner	0.098	0.069	0.44	0.228	129.669	0

Table 7: Sample size: Results of Monte Carlo simulations using dataset 7 from Dorie et al. (2019) with varying sample sizes, 20 replications each. Percent bias is calculated as the estimator's bias as a percentage of its standard error, rmse is root mean squared error, mae is median absolute error, and comp_time is median computation time measured in seconds for each dataset.

method	estimator	size	bias	perc_bias	rmse	mae	comp_time	fail_count
ols	NA	150	NaN	NaN	NaN	NA	0.010	20
ols	NA	300	-0.125	-0.148	1.63	1.769	0.012	17
ols	NA	600	-0.042	-0.034	1.03	0.471	0.018	7
ols	NA	1200	0.215	0.171	0.95	0.590	0.025	0
ols	NA	2400	0.186	0.149	0.83	0.680	0.035	0
ols	NA	4802	0.186	0.150	0.79	0.692	0.061	0
ols	NA	9604	0.275	0.185	0.65	0.477	0.115	0
ols	NA	24010	0.253	0.136	0.62	0.440	0.273	0
ols	NA	48020	0.228	0.242	0.57	0.357	0.597	0
ols	NA	96040	0.183	0.121	0.52	0.314	1.090	0
psm	NA	150	NaN	NaN	NaN	NA	0.013	20
psm	NA	300	NaN	NaN	NaN	NA	0.015	20
psm	NA	600	NaN	NaN	NaN	NA	0.085	20
psm	NA	1200	0.441	0.230	0.55	0.514	0.151	17
psm	NA	2400	0.201	0.150	1.01	0.626	0.274	6
psm	NA	4802	0.174	0.137	0.93	0.761	0.579	1
psm	NA	9604	0.203	0.140	0.66	0.467	1.528	1
psm	NA	24010	0.190	0.105	0.55	0.318	7.664	1
psm	NA	48020	0.198	0.223	0.60	0.334	27.511	1
psm	NA	96040	0.128	0.093	0.37	0.165	113.044	1
ipw	logit	150	0.331	0.185	1.46	1.008	0.077	0
ipw	logit	300	-4.856	-1.395	5.99	3.923	0.098	0
ipw	logit	600	-5.981	-1.964	6.87	5.217	0.140	0
ipw	logit	1200	-7.081	-2.511	7.73	7.662	0.206	0
ipw	logit	2400	-7.201	-2.436	7.88	7.618	0.319	0
ipw	logit	4802	-7.888	-2.757	8.52	8.004	0.529	0
ipw	logit	9604	-9.370	-2.434	10.30	9.599	1.135	0
ipw	logit	24010	-6.516	-2.179	7.49	6.492	1.783	0
ipw	logit	48020	-8.914	-2.652	9.66	8.666	3.513	0
ipw	logit	96040	-8.635	-1.779	9.88	8.161	7.372	0
ipw	grf	150	0.351	0.206	1.24	0.787	1.613	0
ipw	grf	300	0.377	0.248	1.17	0.588	2.533	0
ipw	grf	600	0.351	0.248	0.99	0.421	4.428	0
ipw	grf	1200	0.425	0.303	0.99	0.552	7.752	0
ipw	grf	2400	0.334	0.241	0.79	0.576	15.160	0
ipw	grf	4802	0.320	0.240	0.69	0.492	36.832	0
ipw	grf	9604	0.244	0.148	0.60	0.268	77.877	0
ipw	grf	24010	0.299	0.162	0.59	0.294	217.183	0
ipw	grf	48020	0.172	0.143	0.56	0.390	589.802	0
ipw	grf	96040	0.128	0.077	0.34	0.182	1872.404	0
ipw	superlearner	150	0.294	0.180	1.18	0.869	10.070	0
ipw	superlearner	300	0.334	0.238	1.02	0.661	12.480	0
ipw	superlearner	600	0.367	0.264	0.98	0.451	21.676	0
ipw	superlearner	1200	0.392	0.285	0.94	0.583	37.717	0
ipw	superlearner	2400	0.342	0.255	0.76	0.519	67.716	0

ipw	superlearner	4802	0.319	0.248	0.65	0.449	125.311	0
ipw	superlearner	9604	0.249	0.152	0.56	0.157	251.578	0
ipw	superlearner	24010	0.273	0.148	0.51	0.288	623.468	0
ipw	superlearner	48020	0.103	0.090	0.46	0.322	1175.452	0
ipw	superlearner	96040	0.021	0.013	0.31	0.131	2240.567	0
g-comp	grf	150	-2.439	-1.828	2.77	2.528	1.611	0
g-comp	grf	300	-1.758	-1.177	2.08	1.951	2.531	0
g-comp	grf	600	-1.089	-0.769	1.42	1.187	4.426	0
g-comp	grf	1200	-0.634	-0.498	0.99	0.655	7.751	0
g-comp	grf	2400	-0.386	-0.324	0.71	0.378	15.157	0
g-comp	grf	4802	-0.250	-0.219	0.56	0.192	36.828	0
g-comp	grf	9604	-0.127	-0.073	0.37	0.239	77.868	0
g-comp	grf	24010	-0.079	-0.045	0.27	0.156	217.163	0
g-comp	grf	48020	-0.117	-0.115	0.24	0.103	589.773	0
g-comp	grf	96040	-0.154	-0.098	0.26	0.123	1872.343	0
g-comp	superlearner	150	-0.567	-0.385	1.33	0.607	10.035	0
g-comp	superlearner	300	-0.359	-0.283	0.89	0.528	12.479	0
g-comp	superlearner	600	-0.197	-0.157	0.75	0.293	21.673	0
g-comp	superlearner	1200	0.048	0.041	0.60	0.301	37.716	0
g-comp	superlearner	2400	0.006	0.005	0.38	0.150	67.708	0
g-comp	superlearner	4802	-0.004	-0.003	0.28	0.089	125.308	0
g-comp	superlearner	9604	0.122	0.078	0.33	0.107	251.571	0
g-comp	superlearner	24010	0.019	0.011	0.14	0.050	623.449	0
g-comp	superlearner	48020	-0.010	-0.010	0.25	0.053	1175.419	0
g-comp	superlearner	96040	0.041	0.025	0.21	0.151	2240.497	0
lin	NA	150	NaN	NaN	NaN	NA	0.011	20
lin	NA	300	-1.027	-0.557	2.93	2.322	0.011	17
lin	NA	600	0.175	0.138	1.16	0.550	0.030	7
lin	NA	1200	0.310	0.225	0.94	0.493	0.048	0
lin	NA	2400	0.255	0.184	0.73	0.540	0.078	0
lin	NA	4802	0.230	0.170	0.72	0.517	0.157	0
lin	NA	9604	0.258	0.158	0.58	0.440	0.325	0
lin	NA	24010	0.402	0.206	0.69	0.489	0.824	0
lin	NA	48020	0.111	0.112	0.51	0.354	1.646	0
lin	NA	96040	0.007	0.004	0.89	0.464	3.359	0
aipw	ols_logit	150	-0.368	-0.140	1.86	0.681	0.075	0
aipw	ols_logit	300	-6.828	-0.814	10.31	4.078	0.097	0
aipw	ols_logit	600	-6.623	-1.890	7.53	5.425	0.139	0
aipw	ols_logit	1200	-8.480	-2.252	9.23	8.810	0.204	0
aipw	ols_logit	2400	-8.606	-2.188	9.48	8.856	0.318	0
aipw	ols_logit	4802	-9.584	-2.455	10.41	10.369	0.527	0
aipw	ols_logit	9604	-11.622	-2.343	12.81	11.258	1.127	0
aipw	ols_logit	24010	-7.479	-2.309	8.59	6.902	1.772	0
aipw	ols_logit	48020	-11.523	-2.400	12.66	11.548	3.481	0
aipw	ols_logit	96040	-10.762	-1.780	12.29	9.426	7.334	0
aipw	grf	150	-0.517	-0.349	1.32	0.927	1.612	0
aipw	grf	300	-0.383	-0.285	0.98	0.581	2.532	0
aipw	grf	600	-0.290	-0.222	0.81	0.330	4.426	0
aipw	grf	1200	-0.116	-0.092	0.66	0.183	7.751	0
aipw	grf	2400	-0.086	-0.071	0.50	0.213	15.158	0
aipw	grf	4802	-0.073	-0.063	0.42	0.189	36.829	0
aipw	grf	9604	-0.035	-0.021	0.29	0.227	77.869	0
aipw	grf	24010	-0.001	-0.001	0.24	0.087	217.166	0
aipw	grf	48020	-0.075	-0.074	0.20	0.055	589.777	0

aipw	grf	96040	-0.132	-0.084	0.24	0.107	1872.350	0
aipw	superlearner	150	-0.347	-0.238	1.24	0.590	10.035	0
aipw	superlearner	300	-0.283	-0.225	0.85	0.391	12.479	0
aipw	superlearner	600	-0.169	-0.133	0.77	0.268	21.674	0
aipw	superlearner	1200	0.057	0.048	0.62	0.300	37.717	0
aipw	superlearner	2400	0.011	0.009	0.38	0.149	67.708	0
aipw	superlearner	4802	-0.003	-0.002	0.27	0.081	125.308	0
aipw	superlearner	9604	0.102	0.064	0.28	0.095	251.572	0
aipw	superlearner	24010	0.023	0.013	0.14	0.050	623.451	0
aipw	superlearner	48020	-0.016	-0.016	0.20	0.050	1175.424	0
aipw	superlearner	96040	-0.020	-0.012	0.16	0.100	2240.504	0
tmle	ols_logit	150	-0.138	-0.077	1.33	0.587	0.080	0
tmle	ols_logit	300	-1.275	-0.650	2.12	0.954	0.101	0
tmle	ols_logit	600	-1.648	-0.850	2.24	1.452	0.147	0
tmle	ols_logit	1200	-1.350	-0.732	2.07	1.317	0.212	0
tmle	ols_logit	2400	-1.359	-0.653	2.04	1.065	0.327	0
tmle	ols_logit	4802	-1.283	-0.625	2.06	0.948	0.544	0
tmle	ols_logit	9604	-0.680	-0.311	1.37	1.063	1.185	0
tmle	ols_logit	24010	-1.124	-0.498	1.64	0.797	1.875	0
tmle	ols_logit	48020	-1.248	-0.622	1.94	0.923	3.693	0
tmle	ols_logit	96040	-0.864	-0.543	1.46	0.643	7.596	0
tmle	grf	150	1.186	0.693	1.81	1.649	1.615	0
tmle	grf	300	1.413	0.638	2.43	1.178	2.537	0
tmle	grf	600	0.795	0.537	1.37	0.871	4.430	0
tmle	grf	1200	0.573	0.422	0.94	0.704	7.756	0
tmle	grf	2400	0.341	0.265	0.64	0.541	15.167	0
tmle	grf	4802	0.225	0.183	0.40	0.416	36.842	0
tmle	grf	9604	0.137	0.085	0.34	0.255	77.898	0
tmle	grf	24010	0.113	0.064	0.28	0.143	217.227	0
tmle	grf	48020	-0.003	-0.003	0.16	0.071	589.889	0
tmle	grf	96040	-0.106	-0.067	0.23	0.113	1872.553	0
tmle	superlearner	150	-0.031	-0.022	1.23	0.757	10.039	0
tmle	superlearner	300	-0.110	-0.088	0.90	0.557	12.484	0
tmle	superlearner	600	-0.088	-0.070	0.81	0.303	21.679	0
tmle	superlearner	1200	0.089	0.075	0.66	0.298	37.722	0
tmle	superlearner	2400	0.030	0.023	0.40	0.150	67.717	0
tmle	superlearner	4802	0.005	0.004	0.29	0.098	125.321	0
tmle	superlearner	9604	0.086	0.054	0.26	0.091	251.605	0
tmle	superlearner	24010	0.014	0.008	0.14	0.048	623.518	0
tmle	superlearner	48020	0.002	0.002	0.14	0.058	1175.546	0
tmle	superlearner	96040	-0.140	-0.092	0.31	0.105	2240.722	0
dml	ols_logit	150	-56.640	-0.281	204.19	3.448	0.093	0
dml	ols_logit	300	-2.596	-0.162	15.70	2.425	0.140	0
dml	ols_logit	600	-0.263	-0.050	5.01	0.902	0.196	0
dml	ols_logit	1200	0.654	0.362	1.90	0.770	0.265	0
dml	ols_logit	2400	-0.291	-0.105	2.47	0.656	0.450	0
dml	ols_logit	4802	-0.098	-0.054	1.42	0.695	0.660	0
dml	ols_logit	9604	0.263	0.174	0.64	0.432	1.173	0
dml	ols_logit	24010	0.294	0.156	0.66	0.475	2.197	0
dml	ols_logit	48020	0.218	0.227	0.58	0.327	3.121	0
dml	ols_logit	96040	0.122	0.079	0.57	0.335	6.145	0
dml	grf	150	-0.444	-0.260	1.09	0.996	1.756	14
dml	grf	300	-0.270	-0.100	2.50	1.062	3.010	0
dml	grf	600	0.379	0.238	1.38	1.014	4.722	0

dml	grf	1200	0.593	0.356	1.48	0.619	8.475	0
dml	grf	2400	0.409	0.301	0.98	0.744	16.105	0
dml	grf	4802	0.423	0.306	0.96	0.626	31.840	0
dml	grf	9604	0.283	0.186	0.61	0.399	65.090	0
dml	grf	24010	0.179	0.097	0.49	0.377	174.612	0
dml	grf	48020	0.227	0.223	0.54	0.368	391.386	0
dml	grf	96040	0.188	0.119	0.32	0.210	855.111	0
dml	superlearner	150	-1.382	-0.229	5.75	1.051	9.933	0
dml	superlearner	300	0.366	0.206	1.76	1.072	18.926	0
dml	superlearner	600	1.394	0.356	3.98	0.898	23.790	0
dml	superlearner	1200	1.884	0.427	4.73	0.884	40.678	0
dml	superlearner	2400	1.729	0.423	4.38	1.166	75.058	0
dml	superlearner	4802	1.672	0.456	3.96	1.027	129.318	0
dml	superlearner	9604	0.110	0.062	1.15	0.586	242.361	0
dml	superlearner	24010	-0.081	-0.037	1.31	0.630	574.176	0
dml	superlearner	48020	-0.107	-0.028	3.49	0.573	1105.293	0
dml	superlearner	96040	-0.092	-0.041	1.33	0.340	2218.412	0

Table 8: ATE estimates for Lalonde NSW data as provided by Dehejia and Wahba (1999), with CPS and PSID comparison groups. Bootstrap standard errors shown in parentheses. Covariates include age, education in years of schooling, earnings in 1975, and dichotomous variables for Black and Hispanic race, married, not having a high school degree, and having no earnings in 1975. The "With 1974 earnings" estimates additionally include earnings in 1974 as a covariate, along with an indicator for having no earnings in 1974.

'74?	Method	Experimental	CPS-1	CPS-3	PSID-1	PSID-3
No	ols	802 (511)	-809 (530)	-831 (430)	-1672 (709)	510 (983)
No	psm	788 (557)	-564 (628)	-560 (598)	-2515 (884)	-1680 (1546)
No	ipw, logit	11 (64)	-7299 (673)	-7275 (685)	-10357 (1508)	101 (1002)
No	ipw, grf	827 (485)	-7905 (679)	-7827 (606)	-13333 (1074)	375 (750)
No	ipw, superl.	814 (480)	-7607 (962)	-7338 (1054)	-13654 (1185)	939 (800)
No	g-comp, grf	509 (341)	-22 (138)	-6 (59)	-1004 (524)	-60 (372)
No	g-comp, superl.	673 (467)	-1891 (853)	-1594 (889)	-2514 (1017)	192 (747)
No	lin	795 (512)	-5801 (1469)	-5789 (1440)	-11100 (1537)	262 (976)
No	aipw, ols_logit	-2 (62)	-1041 (430)	-1057 (387)	-2256 (925)	-320 (1200)
No	aipw, grf	702 (432)	-156 (146)	-135 (89)	-1617 (569)	-144 (509)
No	aipw, superl.	734 (469)	-1915 (854)	-1614 (870)	-2811 (1022)	426 (769)
No	tmle, ols_logit	-2819 (1707)	-714 (481)	-737 (393)	59 (421)	909 (1086)
No	tmle, grf	1650 (1131)	-47 (135)	-20 (83)	-1896 (562)	-284 (1354)
No	tmle, superl.	1069 (720)	-1907 (849)	-1611 (886)	-2476 (985)	1223 (1059)
No	dml, ols_logit	750 (534)	-1001 (578)	-1026 (501)	-2977 (921)	320 (1145)
No	dml, grf	817 (502)	-1876 (515)	-1964 (512)	-5180 (847)	142 (968)
No	dml, superl.	869 (517)	-647 (585)	-636 (531)	-2866 (963)	157 (1464)
Yes	ols	1698 (758)	1083 (609)	1140 (635)	-111 (899)	1089 (1022)
Yes	psm	1717 (803)	1525 (751)	1606 (701)	54 (901)	968 (1563)
Yes	ipw, logit	23 (169)	-8115 (704)	-7998 (584)	-11317 (1910)	1034 (1088)
Yes	ipw, grf	1655 (605)	-7480 (1083)	-7277 (1121)	-12658 (2031)	965 (817)
Yes	ipw, superl.	1652 (646)	-5560 (1984)	-5444 (1772)	-11066 (2513)	2060 (919)
Yes	g-comp, grf	981 (520)	87 (73)	85 (65)	45 (157)	609 (434)
Yes	g-comp, superl.	1502 (706)	228 (916)	233 (844)	-746 (849)	996 (796)
Yes	lin	1603 (742)	-4590 (4028)	-4380 (3432)	-8717 (4165)	1625 (1035)
Yes	aipw, ols_logit	10 (182)	684 (581)	738 (570)	-1249 (1256)	1381 (1357)
Yes	aipw, grf	1398 (553)	60 (92)	69 (87)	-264 (299)	586 (588)
Yes	aipw, superl.	1576 (652)	251 (921)	260 (850)	-901 (917)	1689 (894)
Yes	tmle, ols_logit	-4609 (2583)	1062 (582)	1115 (609)	956 (813)	54 (940)
Yes	tmle, grf	3509 (1231)	72 (117)	75 (97)	-455 (265)	441 (1658)
Yes	tmle, superl.	2096 (864)	229 (914)	235 (844)	-398 (890)	1943 (1180)
Yes	dml, ols_logit	1599 (810)	1038 (676)	1059 (601)	-621 (1048)	1090 (1591)
Yes	dml, grf	1723 (742)	-217 (717)	-164 (757)	-3565 (1047)	884 (1065)
Yes	dml, superl.	1711 (754)	80 (702)	147 (720)	-585 (1094)	1170 (1316)

References

Angrist, J. D., & Krueger, A. B. (1995). Split-Sample Instrumental Variables Estimates of the Return to Schooling. *Journal of Business & Economic Statistics*, *13*(2), 225–235. https://doi.org/10.2307/1392377

Arkhangelsky, D., Imbens, G. W., Lei, L., & Luo, X. (2021). *Double-Robust Two-Way-Fixed-Effects Regression For Panel Data* (No. arXiv:2107.13737). arXiv. https://doi.org/10.48550/arXiv.2107.13737

Athey, S., Tibshirani, J., & Wager, S. (2019). Generalized random forests. *The Annals of Statistics*, *47*(2), 1148–1178. https://doi.org/10.1214/18-AOS1709

Austin, P. C., & Stuart, E. A. (2015). Moving towards best practice when using inverse probability of treatment weighting (IPTW) using the propensity score to estimate causal treatment effects in observational studies. *Statistics in Medicine*, *34*(28), 3661–3679. https://doi.org/10.1002/sim.6607

Bach, P., Chernozhukov, V., Kurz, M. S., & Spindler, M. (2021). *DoubleML – An Object-Oriented Implementation of Double Machine Learning in R*. https://doi.org/10.48550/arXiv.2103.09603

Balzer, L. B., & Petersen, M. L. (2021). Invited Commentary: Machine Learning in Causal Inference—How Do I Love Thee? Let Me Count the Ways. *American Journal of Epidemiology*, *kwab048*. https://doi.org/10.1093/aje/kwab048

Bang, H., & Robins, J. M. (2005). Doubly Robust Estimation in Missing Data and Causal Inference Models. *Biometrics*, *61*(4), 962–973. https://doi.org/10.1111/j.1541-0420.2005.00377.x

Brand, J. E., Zhou, X., & Xie, Y. (2023). Recent Developments in Causal Inference and Machine Learning. *Annual Review of Sociology*, *49*(1), 81–110. https://doi.org/10.1146/annurev-soc-030420-015345

Cassel, C. M., Särndal, C. E., & Wretman, J. H. (1976). Some results on generalized difference estimation and generalized regression estimation for finite populations. *Biometrika*, *63*(3), 615–620. https://doi.org/10.1093/biomet/63.3.615

Chatton, A., Le Borgne, F., Leyrat, C., Gillaizeau, F., Rousseau, C., Barbin, L., Laplaud,

D., Léger, M., Giraudeau, B., & Foucher, Y. (2020). G-computation, propensity score-based methods, and targeted maximum likelihood estimator for causal inference with different covariates sets: A comparative simulation study. *Scientific Reports*, *10*(1), 9219. https://doi.org/10.1038/s41598-020-65917-x

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. https://doi.org/10.1145/2939672.2939785

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., & Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, *21*(1), C1–C68. https://doi.org/10.1111/ectj.12097

Chernozhukov, V., Cinelli, C., Newey, W., Sharma, A., & Syrgkanis, V. (2022). *Long Story Short: Omitted Variable Bias in Causal Machine Learning* (No. arXiv:2112.13398). arXiv. https://doi.org/10.48550/arXiv.2112.13398

Cousineau, M., Verter, V., Murphy, S. A., & Pineau, J. (2022). Estimating causal effects with optimization-based methods: A review and empirical comparison. *European Journal of Operational Research*, S0377221722000844. https://doi.org/10.1016/j.ejor.2022.01.046

Dehejia, R. H., & Wahba, S. (1999). Causal Effects in Nonexperimental Studies: Reevaluating the Evaluation of Training Programs. *Journal of the American Statistical Association*, *94*(448), 1053–1062. https://doi.org/10.2307/2669919

Díaz, I. (2020). Machine learning in the estimation of causal effects: Targeted minimum loss-based estimation and double/debiased machine learning. *Biostatistics*, *21*(2), 353–358. https://doi.org/10.1093/biostatistics/kxz042

Dorie, V., Hill, J., Shalit, U., Scott, M., & Cervone, D. (2019). Automated versus Do-It-Yourself Methods for Causal Inference: Lessons Learned from a Data Analysis Competition. *Statistical Science*, *34*(1), 43–68. https://doi.org/10.1214/18-STS667

Dukes, O., Vansteelandt, S., & Whitney, D. (2022). *On doubly robust inference for double machine learning* (No. arXiv:2107.06124). arXiv. https://doi.org/10.48550/arXiv.2107.06124

Farbmacher, H., Huber, M., Laffers, L., Langen, H., & Spindler, M. (2022). Causal mediation

analysis with double machine learning. *The Econometrics Journal*, *25*(2), 277–300. https://doi.org/10.1093/ectj/utac003

Friedman, J., Hastie, T., Tibshirani, R., Narasimhan, B., Tay, K., Simon, N., & Qian, J. (2021). Package "glmnet." *CRAN R Repositary*, *595*.

Frisch, R., & Waugh, F. V. (1933). Partial Time Regressions as Compared with Individual Trends. *Econometrica*, *1*(4), 387–401. https://doi.org/10.2307/1907330

Funk, M. J., Westreich, D., Wiesen, C., Stürmer, T., Brookhart, M. A., & Davidian, M. (2011). Doubly Robust Estimation of Causal Effects. *American Journal of Epidemiology*, *173*(7), 761–767. https://doi.org/10.1093/aje/kwq439

Glynn, A. N., & Quinn, K. M. (2010). An Introduction to the Augmented Inverse Propensity Weighted Estimator. *Political Analysis*, *18*(1), 36–56. https://doi.org/10.1093/pan/mpp036

Gruber, S., & Laan, M. van der. (2009). Targeted Maximum Likelihood Estimation: A Gentle Introduction. *U.C. Berkeley Division of Biostatistics Working Paper Series*.

Gruber, S., & Laan, M. van der. (2012). Tmle: An R Package for Targeted Maximum Likelihood Estimation. *Journal of Statistical Software*, *51*, 1–35. https://doi.org/10.18637/jss.v051.i13

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. Springer.

Hazlett, C., & Shinkre, T. (2024). *Understanding and avoiding the "weights of regression": Heterogeneous effects, misspecification, and longstanding solutions* (No. arXiv:2403.03299). arXiv. https://doi.org/10.48550/arXiv.2403.03299

Hernán, M. A., & Robins, J. M. (2020). *Causal Inference: What If*. CRC Press.

Hines, O., Dukes, O., Diaz-Ordaz, K., & Vansteelandt, S. (2022). Demystifying Statistical Learning Based on Efficient Influence Functions. *The American Statistician*, *76*(3), 292–304. https://doi.org/10.1080/00031305.2021.2021984

Hünermund, P., Louw, B., & Caspi, I. (2023). Double Machine Learning and Automated Confounder Selection – A Cautionary Tale. *Journal of Causal Inference*, *11*(1), 20220078. https://doi.org/10.1515/jci-2022-0078

Jacob, D. (2021). CATE meets ML. *Digital Finance*, *3*(2), 99–148. https://doi.org/10.1007/s42521-021-00033-7

Jung, Y., Tian, J., & Bareinboim, E. (2021). Estimating Identifiable Causal Effects through Double Machine Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(13), 12113–12122. https://doi.org/10.1609/aaai.v35i13.17438

Kang, J. D. Y., & Schafer, J. L. (2007). Demystifying Double Robustness: A Comparison of Alternative Strategies for Estimating a Population Mean from Incomplete Data. *Statistical Science*, *22*(4), 523–539. https://doi.org/10.1214/07-STS227

Keil, A. P., Mooney, S. J., Jonsson Funk, M., Cole, S. R., Edwards, J. K., & Westreich, D. (2018). Resolving an Apparent Paradox in Doubly Robust Estimators. *American Journal of Epidemiology*, *187*(4), 891–892. https://doi.org/10.1093/aje/kwx385

Kennedy, E. H. (2023). *Semiparametric doubly robust targeted double machine learning: A review* (No. arXiv:2203.06469). arXiv. https://doi.org/10.48550/arXiv.2203.06469

Knaus, M. C. (2022). Double machine learning-based programme evaluation under unconfoundedness. *The Econometrics Journal*, *25*(3), 602–627. https://doi.org/10.1093/ectj/utac015

LaLonde, R. J. (1986). Evaluating the Econometric Evaluations of Training Programs with Experimental Data. *The American Economic Review*, *76*(4), 604–620. https://www.jstor.org/stable/1806062

Lin, W. (2013). Agnostic notes on regression adjustments to experimental data: Reexamining Freedman's critique. *Annals of Applied Statistics*, *7*(1), 295–318. https://doi.org/10.1214/12-AOAS583

Lovell, M. C. (1963). Seasonal Adjustment of Economic Time Series and Multiple Regression Analysis. *Journal of the American Statistical Association*, *58*(304), 993–1010. https://doi.org/10.1080/01621459.1963.10480682

Lundberg, I., Brand, J. E., & Jeon, N. (2022). Researcher reasoning meets computational capacity: Machine learning for social science. *Social Science Research*, *108*, 102807. https://doi.org/10.1016/j.ssresearch.2022.102807

Luque-Fernandez, M. A., Schomaker, M., Rachet, B., & Schnitzer, M. E. (2018). Targeted

maximum likelihood estimation for a binary treatment: A tutorial. *Statistics in Medicine*, *37*(16), 2530–2546. https://doi.org/10.1002/sim.7628

Morgan, S. L., & Winship, C. (2015). *Counterfactuals and causal inference: Methods and principles for social research* (2nd ed.). Cambridge University Press.

Nie, X., & Wager, S. (2021). Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, *108*(2), 299–319. https://doi.org/10.1093/biomet/asaa076

Okui, R., Small, D. S., Tan, Z., & Robins, J. M. (2012). Doubly Robust Instrumental Variable Regression. *Statistica Sinica*, *22*(1), 173–205. https://www.jstor.org/stable/24310144

Polley, E., LeDell, E., Kennedy, C., Lendle, S., & Laan, M. van der. (2023). *SuperLearner: Super Learner Prediction*.

Ratkovic, M. (2023). Relaxing Assumptions, Improving Inference: Integrating Machine Learning and the Linear Regression. *American Political Science Review*, *117*(3), 1053–1069. https://doi.org/10.1017/S0003055422001022

Robins, J. M. (1986). A new approach to causal inference in mortality studies with a sustained exposure period—application to control of the healthy worker survivor effect. *Mathematical Modelling*, *7*(9), 1393–1512. https://doi.org/10.1016/0270-0255(86)90088-6

Robins, J. M., Rotnitzky, A., & Zhao, L. P. (1994). Estimation of Regression Coefficients When Some Regressors are not Always Observed. *Journal of the American Statistical Association*, *89*(427), 846–866. https://doi.org/10.1080/01621459.1994.10476818

Rotnitzky, A., Robins, J. M., & Scharfstein, D. O. (1998). Semiparametric Regression for Repeated Outcomes with Nonignorable Nonresponse. *Journal of the American Statistical Association*, *93*(444), 1321–1339. https://doi.org/10.2307/2670049

Rubin, D. B. (1973). The Use of Matched Sampling and Regression Adjustment to Remove Bias in Observational Studies. *Biometrics*, *29*(1), 185–203. https://doi.org/10.2307/2529685

Rubin, D. B. (1974). Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology*, *66*(5), 688–701. https://doi.org/10.1037/h0037350

Rubin, D. B. (1980). Randomization Analysis of Experimental Data: The Fisher Randomiza-

tion Test Comment. *Journal of the American Statistical Association*, *75*(371), 591–593. https://doi.org/10.2307/2287653

Sant'Anna, P. H. C., & Zhao, J. (2020). Doubly robust difference-in-differences estimators. *Journal of Econometrics*, *219*(1), 101–122. https://doi.org/10.1016/j.jeconom.2020.06.003

Scharfstein, D. O., Rotnitzky, A., & Robins, J. M. (1999). Adjusting for Nonignorable Drop-Out Using Semiparametric Nonresponse Models. *Journal of the American Statistical Association*, *94*(448), 1096–1120. https://doi.org/10.1080/01621459.1999.10473862

Schuler, M. S., & Rose, S. (2017). Targeted Maximum Likelihood Estimation for Causal Inference in Observational Studies. *American Journal of Epidemiology*, *185*(1), 65–73. https://doi.org/10.1093/aje/kww165

Semenova, V., & Chernozhukov, V. (2021). Debiased machine learning of conditional average treatment effects and other causal functions. *The Econometrics Journal*, *24*(2), 264–289. https://doi.org/10.1093/ectj/utaa027

Słoczyński, T., & Wooldridge, J. M. (2018). A general double robustness result for estimating average treatment effects. *Econometric Theory*, *34*(1), 112–133. https://doi.org/10.1017/S0266466617000056

Snowden, J. M., Rose, S., & Mortimer, K. M. (2011). Implementation of G-Computation on a Simulated Data Set: Demonstration of a Causal Inference Technique. *American Journal of Epidemiology*, *173*(7), 731–738. https://doi.org/10.1093/aje/kwq472

Tran, L., Yiannoutsos, C., Wools-Kaloustian, K., Siika, A., Laan, M. van der, & Petersen, M. (2019). Double Robust Efficient Estimators of Longitudinal Treatment Effects: Comparative Performance in Simulations and a Case Study. *The International Journal of Biostatistics*, *15*(2). https://doi.org/10.1515/ijb-2017-0054

van der Laan, M. J., Polley, E. C., & Hubbard, A. E. (2007). Super Learner. *Statistical Applications in Genetics and Molecular Biology*, *6*(1). https://doi.org/10.2202/1544-6115.1309

van der Laan, M. J., & Rose, S. (2011). *Targeted Learning: Causal Inference for Observational and Experimental Data*. Springer New York. https://doi.org/10.1007/978-1-4419-9782-1

Van Der Laan, M. J., & Rose, S. (2018). *Targeted Learning in Data Science: Causal*

Inference for Complex Longitudinal Studies. Springer International Publishing. https://doi.org/10.1007/978-3-319-65304-4

van der Laan, M. J., & Rubin, D. (2006). Targeted Maximum Likelihood Learning. *The International Journal of Biostatistics*, *2*(1). https://doi.org/10.2202/1557-4679.1043

Wang, L., & Tchetgen Tchetgen, E. (2018). Bounded, Efficient and Multiply Robust Estimation of Average Treatment Effects Using Instrumental Variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *80*(3), 531–550. https://doi.org/10.1111/rssb.12262

Yu, Z., & van der Laan, M. (2006). Double robust estimation in longitudinal marginal structural models. *Journal of Statistical Planning and Inference*, *136*(3), 1061–1089. https://doi.org/10.1016/j.jspi.2004.08.011

Zhong, Y., Kennedy, E. H., Bodnar, L. M., & Naimi, A. I. (2021). AIPW: An R Package for Augmented Inverse Probability–Weighted Estimation of Average Causal Effects. *American Journal of Epidemiology*, *190*(12), 2690–2699. https://doi.org/10.1093/aje/kwab207