A Simulation Study on the Sensitivity of Causal Machine Learning Methods to Violations of the Conditional Independence Assumption

Master Thesis

Author:

Pascal Yannick Homberger

19-615-376

Master of Arts in Economics (MEcon)

University of St. Gallen (HSG)

Supervisor:

Prof. Dr. Jana Mareckova

Swiss Institute for Empirical Economic Research (SEW)

Assistant Professor of Econometrics

November 17, 2024

Abstract

This paper investigates the sensitivity of Average Treatment Effect (ATE) estimation to violations of the Conditional Independence Assumption (CIA) within causal machine learning. While machine learning methods provide flexibility for handling complex data structures, fundamental identifying assumptions like the CIA must still hold to ensure unbiased estimates. Four machine learning approaches—Debiased Machine Learning (DML), T-Learner, X-Learner, and Accelerated Bayesian Causal Forest (XBCF)—are evaluated through a simulation study using different Data Generating Processes (DGPs). Sensitivity to CIA violations is tested by systematically omitting confounders to introduce bias. Results show that DML, when combined with XGBoost for propensity score estimation, exhibits unexpected behavior, including high bias and inconsistencies. XBCF and X-Learner perform most robustly, with X-Learner excelling in linear data structures and both methods showing resilience in non-linear settings. Additionally, the findings confirm that omitting confounders introduces significant bias across all methods, highlighting the importance of meeting the CIA in causal machine learning applications.

Contents

List of Tables

List of Figures

1 Introduction

From analyzing the long-term impacts of early educational interventions for children from low-income families (Garces, Thomas, & Currie, 2002) to studying the effects of physical activity on longevity (Paffenbarger, Hyde, Wing, & Hsieh, 1986), research papers investigating causal effects are extensive and varied. These two examples represent only a small segment of a vast body of research aimed at understanding causal relationships across diverse contexts. Accurate estimation of these effects is crucial, as it can influence policy decisions and behavioral changes that have large impacts on society. For instance, if early intervention is shown to significantly improve educational outcomes, scaling up such programs could be a strategic approach to increase social mobility. Likewise, confirming the health benefits of physical activity could bolster public health initiatives. However, the utility of these insights depend on ensuring that effect estimations are both accurate and unbiased.

Achieving accurate causal inference requires that certain "identifying assumptions" hold. These assumptions include the Stable Unit Treatment Value Assumption (SUTVA), common support, exogeneity of confounders, and, importantly, the Conditional Independence Assumption (CIA) (see for example: imbens, 2000; Lechner, 2001; Yao, Chu, Li, Li, Gao, & Zhang, 2021). Where the later 3 are more focused in observational studies. Each of these foundational assumptions will be detailed in the methodology section of this paper.

This study focuses on investigating the CIA, specifically examining how violations of this assumption affect the estimation of the Average Treatment Effect (ATE). A second key element of this research is the application of machine learning methods. In recent years, machine learning techniques have gained space in causal effect estimation (see for example: Brand, Xu, Koch, & Geraldo, 2021; Brand, Zhou, & Xie, 2023). One major advantage of these methods is their flexibility regarding functional form assumptions, allowing them to handle more complex data structures than traditional approaches like Linear Regression.

By integrating the exploration of CIA violations with an evaluation of machine learning methods, this study addresses the following research question: In the context of causal inference, how sensitive is the estimate of the ATE to violations of the CIA when assessed using various causal machine learning methods?

To address the research question, a simulation study is conducted using several machine learning methods: Doubly/Debiased Machine Learning (DML) (Chernozhukov et al., 2018), T-Learner, X-Learner (Künzel, Sekhon, Bickel, & Yu, 2019), and Accelerated Bayesian Causal Forest (XBCF) (Krantsevich, Jingyu, & Richard, 2023). Additionally, linear regression estimated by Ordinary Least Squares (OLS) is included as a benchmark. Two distinct DGPs are employed—one reflecting linear relationships and another constructed to represent more complex, non-linear environments. These DGPs are intentionally designed to introduce confounding, and CIA violations are simulated by systematically omitting confounders to examine the resulting bias in ATE estimates. Scenarios are created in which varying proportions of confounders are omitted. A specific metric, referred to here as the "correlation measure" is employed to quantify the strength of confounding in each scenario.

The results of this simulation have the potential to improve empirical practices in causal research. By better understanding how machine learning methods perform under CIA violations, researchers can make informed decisions about study design. For instance, if a study is likely to encounter strong confounding, it might prompt consideration of alternative approaches, such as instrumental variable setups, rather than a standard regression. Additionally, insights from the results could guide researchers in selecting appropriate machine learning methods when suspecting similar data structures to the Data Generating Processes (DGPs) explored in this study. While there are many simulation studies testing machine learning algorithms against each other (see for example: Dorie, Hill, Shalit, Scott, & Cervone, n.d.; Hahn, Murray, & Carvalho, 2020; McConnell & Lindner, 2019; Schuler & Rose, 2017), there are not many combining it with the CIA violations, which is why this paper does add to the current research environment.

Results of the simulation show that in the Linear DGP, Linear Regression performed consistently well, aligning with theoretical expectations, while the X-Learner also showed surprising accuracy. The DML method, however, exhibited less predictable behavior, occasionally producing better estimates when certain confounders were omitted—a counterintuitive outcome.

For the Non-Linear DGP, Linear Regression struggled due to misspecification. DML showed instability and irregular behavior, with bias shifting significantly depending on which specific confounder was omitted. This instability is suspected to stem from the XGBoost algorithm used to estimate the nuisance function. Among the machine learning estimators, the X-Learner and XBCF provided stable, accurate estimates, with XBCF slightly outperforming others in configurations with stronger confounding. The T-Learner did not outperform the X-Learner or XBCF in any scenario.

These results offer some insights to apply for empirical work. Notably, omitting even a single strong confounder can lead to severe bias, underscoring that even with state-of-the-art machine learning methods, researchers must be confident that the CIA holds. However, if a researcher is reasonably confident that all relevant confounders are included in the dataset, the results of this study can help guide the choice of an appropriate estimator.

The structure of this paper begins with a literature review on the use of machine learning in causal inference, followed by the methodology, which provides a theoretical overview of the potential outcomes framework and discusses the key identifying assumptions, including the CIA. Next, the strategy for simulating CIA violations is briefly outlined together with the perfomance measurements used to assess the estimation methods. The DGPs used in the simulation are then described, along with the machine learning methods (DML, T-Learner, X-Learner, XBCF). Finally, the results are presented, followed by a discussion and conclusion on the implications of the findings.

2 Literature Review

Since this paper evaluates different machine learning algorithms for causal estimation, it is useful to show the main contributions and foundational papers related to methods for estimating the ATE. The methods discussed here primarily focus on estimating effects within observational data settings, where confounding is present, and treatment and control groups are not perfectly randomized. Notably, some of these methods were originally developed to estimate heterogeneous treatment effects. However, these can still lead to ATE estimation by aggregating Conditional Average Treatment Effect (CATE) or Individual Average Treatment Effect (IATE) appropriately. This section will not show technical details, which are instead addressed in the methodology section.

For a comprehensive overview of recent advancements in causal inference and machine learning-based estimation methods, refer to Brand, Zhou, and Xie, 2023, Yao, Chu, Li, Li, Gao, and Zhang, 2021, and Athey and Imbens, 2019. One well-established method for estimating the ATE is Inverse Probability Weighting (IPW), which estimates the probability of treatment assignment for each unit, known as the propensity score, and uses this score to weight observations (Rosenbaum & Rubin, 1983). The propensity score helps balance treated and untreated groups, mitigating selection bias (Imbens, 2004). This approach can incorporate machine learning algorithms to estimate propensity scores, as demonstrated by several studies (Brand, Xu, Koch, & Geraldo, 2021; Lee, Lessler, & Stuart, 2010; McCaffrey, Ridgeway, & Morral, 2004; Westreich, Lessler, & Funk, 2010).

Another estimation approach is the Targeted Maximum Likelihood Estimator (TMLE), introduced by van der Laan and Rubin, 2006. TMLE is a doubly robust estimator, meaning it can yield unbiased estimates if either the outcome model or the treatment assignment model (propensity score) is correctly specified. TMLE also includes a ”targeting” step to optimize the trade-off between bias and variance, making it an asymptotically efficient estimator when both models are well-specified. Additionally, as a substitution estimator, TMLE is particularly resilient to issues such as outliers and data sparsity. TMLE’s adaptability to complex data structures increases when combined with machine learning techniques, potentially reducing bias and improving performance compared to traditional parametric estimators like OLS regression (van der Laan & Rose, 2011).

(Chernozhukov et al., 2018) introduced another doubly robust estimator known as the Doubly/Debiased Machine Learner (DML). Parts of this approach build on the ATE estimator initially developed by Robins and Rotnitzky, 1995, which combines propensity score estimation with outcome estimation. The result is a doubly robust framework, meaning that consistency for the ATE is achieved if either the propensity score or the outcome model is correctly specified. Chernozhukov et al., 2018 extended this framework by incorporating machine learning algorithms to estimate the propensity function and outcome function, referred to as nuisance functions. Additionally, they introduced "cross-fitting" methods, where the data is split to estimate the nuisance functions on a training set, while the remaining data is used to estimate nuisance parameters. These parameters are then integrated into the doubly robust estimator. Another important feature of their method is that it only includes so-called Neyman-orthogonal scores, which help to mitigate bias from regularization and model selection within machine learning.

In a related work, Chernozhukov, Newey, and Singh, 2018 presented a method called automatic debiased machine learning addressing the challenge of bias in machine learning methods that are typically optimized for prediction tasks. This method enables ATE estimation while minimizing regularization bias and is flexible enough to be applied to various regression learners, including neural networks, random forests, Lasso, and boosting.

Another broad category of estimators, sometimes referred to as meta-learners includes methods such as the S-learner (Künzel, Sekhon, Bickel, & Yu, 2019), T-learner (Künzel, Sekhon, Bickel, & Yu, 2019), R-learner (Nie & Wager, 2021), and X-learner (Künzel, Sekhon, Bickel, & Yu, 2019), which can estimate causal effects. While these learners were primarily developed to estimate heterogeneous treatment effects, they can also be used to estimate the ATE by aggregating individual treatment effects. These learners share the common feature of incorporating what they call base learners which serve as estimation functions that can utilize a range of algorithms, from simple parametric linear regression models to advanced machine learning models like Bayesian Additive Regression Trees (BART) or XGBoost (Yao, Chu, Li, Li, Gao, & Zhang, 2021). The term meta-learner is used variably in the literature, and some studies do not categorize these methods as meta-learners.

Athey and Imbens, 2019 introduced the Generalized Random Forest (GRF), which can be applied across various causal inference settings, including the estimation of heterogeneous treatment effects. Their approach builds on the classic random forest method but uses adapted weighting functions to capture parameter heterogeneity. Another related method, called Causal Forest (Wager & Athey, 2018), also builds on random forests with adapted splitting rules. This approach is based on Athey and Imbens, 2016, who introduced the concept of causal trees which estimate causal effects at the tree leaves. They introduced the condition of ”honesty” for causal inference, requiring that the data used for splits in the trees differ from that used to estimate causal effects at the leaves. Examples of Causal Forest applications in empirical research include Athey and Imbens, 2016; Davis and Heller, 2017; Miller, 2020.

Another machine learning method that can be utilized for causal estimation Bayesian Additive Regression Trees (BART), as introduced by Chipman, George, and McCulloch, 2010. While BART is a powerful method, it is computationally intensive with large datasets, leading to the development of XBART for increased speed (He, Yalov, & Hahn, 2019). Since BART was originally designed for prediction rather than causal inference, He and Hahn, 2020 developed the Bayesian Causal Forest (BCF) to address confounding issues specifically. Similar to BART, BCF can be slow on large datasets, which led to the development of Accelerated Bayesian Causal Forest (XBCF) with enhanced computational efficiency (Krantsevich, Jingyu, & Richard, 2023). BCF was used in some empirical research papers(Bail et al., 2020; Bryan, Yeager, & O'Brien, 2019; King et al., 2019)

In terms of conducting simulation studies, several sources provide foundational guidance. Morris, White, and Crowther, 2019 offer a comprehensive overview of simulation study design, drawing on a review of 100 articles that included simulations to illustrate best practices. Additionally, Burton, Altman, Royston, and Holder, 2006 provide practical guidance on key components of simulation studies, such as defining study objectives, constructing data-generating processes, determining the number of simulations, and other essential considerations.

Relatively few studies directly focus on testing the sensitivity of causal inference methods to violations of assumptions through simulations. However, some literature addresses strategies for testing the robustness of treatment effect estimates to assumption violations in empirical research. Although this is distinct from the focus of this paper—where machine learning methods are tested in simulations with known treatment effects—these strategies relate to the broader topic of assessing the robustness of the CIA. One such strategy is Rosenbaum's bounding approach (Rosenbaum, 2002), which evaluates the extent to which an unobserved confounder would impact the treatment effect, potentially altering its significance. Becker and Caliendo, 2007 tested this approach in the context of matching estimators, and Caliendo and Kopeinig, 2008 provided a summary of sensitivity testing strategies for CIA violations specifically within propensity score matching frameworks.

3 Methdology

3.1 Notation Overview

An overview of the characters used in the paper:

Treatment effect	θ
Nuisance outcome	μ
Nuisance propensity	p
Outcome	Y
Treatment	D
Covariates or Confounders	X

Unit/Observation A unit or observation i is a particular entity, such as a person or an object. There are N units in a dataset. Both terms unit and observation are used interchangeably.

Outcome The outcome is the dependent variable and is represented as a vector Y_i for each unit i.

Treatment In this paper, the treatment is a variable D_i that takes a binary form $(0, 1)$ and is assigned to each unit i.

Confounders Confounders are variables that correlate with both the outcome and the treatment simultaneously. Multiple confounders j can be assigned to each unit i.

Potential outcomes The potential outcomes for each unit i are defined as Y_{i1} and Y_{i0}. Each unit i has two potential outcomes: Y_{i1} is the outcome if it receives the treatment, and Y_{i0} is the outcome if it does not receive the treatment.

Observed outcome The observed outcome Y_i is the outcome actually observed in the data for a specific treatment and each unit i.

Counterfactual For each observed outcome, there is an opposite potential outcome, known as the counterfactual outcome. In other words, it is the outcome if unit i had been assigned to the opposite treatment.

ATE The Average Treatment Effect (ATE) is the average of all individual treatment effects in the population of interest: $E[Y_1 - Y_0]$.

Nuisance function A function that estimates components necessary for treatment effect estimation. In this paper, two nuisance functions are used: one for propensity and one for outcome.

Propensity score The propensity score is the conditional probability of a unit receiving the treatment, given some background information variables (Rosenbaum & Rubin, 1983). Background information variables are variables that are observed apart from the treatment.

3.2 Theoretical Setting

In this paper, we adopt the potential outcome framework, which originates from the foundational ideas of Neyman (1990) and Fisher (1935) and has been further developed in subsequent works (Rubin, 1974, 1977). This framework is applicable to several causal inference settings, including randomized trials and observational studies. The fundamental principle is that for every outcome Y, there exists an observed outcome and an unobserved counterfactual outcome.

To illustrate this with an example: suppose one wants to determine the effect of receiving a certain medicine (referred to as the treatment) on the survival rate (referred to as the outcome). For each individual, only one outcome is observed—the actual outcome. For instance, if a person receives the treatment, we observe the survival outcome for that person under treatment. However, to know the true causal effect, it would be ideal to also know the unobserved outcome, which would be the survival rate had the person not received the treatment. The difference between these two outcomes would represent the causal effect. The challenge, however, is that it is impossible for the same individual to both receive and not receive the treatment under identical circumstances; thus, one of these outcomes will always remain unobserved. Therefore, the potential outcome framework is employed, which conceptualizes the outcomes as if the individual had either received or not received the treatment.

More technically, let us define D as a vector of treatments assigned to each unit i in the sample. Within the potential outcome framework, where treatment is binary, each unit i could either receive the treatment ($D_i = 1$) or not receive the treatment ($D_i = 0$). The potential outcomes for unit i are denoted as Y_{i1} (if treated) and Y_{i0} (if untreated). Analogous to the earlier example, the causal effect of the treatment for unit i would then be the difference between these potential outcomes: $Y_{i1} - Y_{i0}$.

In the potential outcome framework setting there are some assumptions that must hold true when claiming a causal effect to hold. We assume that the actual outcome, or observed outcome, Y, which is also a vector for i units, is determined by the potential outcomes and the treatment assignment. Specifically, the observed outcome for unit i is given by:

$$Y_i = D_i Y_{i1} + (1 - D_i) Y_{i0} \tag{1}$$

For this relationship to hold, the Stable Unit Treatment Value Assumption (SUTVA) must be satisfied. SUTVA, introduced by Rubin (1980), requires: first, the potential outcome for any unit should not be influenced by the treatment assignments of other units (Cox, 1958); and second, there should be no hidden variations of the treatment, meaning that the potential outcome remains unchanged regardless of how unit i receives the treatment (Rubin, 2005).

After stating the crucial assumption of SUTVA, which must hold when making claims about a causal effect, we define the Average Treatment Effect (ATE) that will be investigated in this paper:

$$ATE = E[Y_1 - Y_0] \tag{2}$$

ATE represents the average effect of the treatment on the outcome for the population of interest.

To correctly identify the ATE with the data at hand, there must be no confounding. Confounding is absent only if no characteristics of the units correlate with both the treatment and the outcome simultaneously. The problem of confounding can be addressed through various strategies, one of which is by conducting an experimental study Fisher (1935), where the treatment and control groups are perfectly randomized and accurately represent the population of interest. This leads to the independence of the potential outcomes from the treatment assignment, denoted as $Y_1, Y_0 \perp D$, where $\perp$ denotes statistical independence.

However, since experimental studies are not always feasible due to high costs or ethical concerns, researchers often rely on observational data where the treatment and control groups are not perfectly randomized, and confounding is usually present. This study aims to investigate within the context of observational data. In this setting, additional assumptions beyond SUTVA must be made to correctly identify the ATE. From here on the assumptions, including SUTVA will be addressed as identifying assumptions.

First, there is the ignorability assumption, also known as unconfoundedness or conditional indipendence assumption (CIA)(Lechner, 1999, 2002), defined as:

$$(Y_1, Y_0) \perp D \mid X \tag{3}$$

This assumption states that the potential outcomes (Y_1, Y_0) are conditionally independent of the treatment assignment D given the covariates X. Conditional independence holds if all confounding variables X are accounted for.

Second, the positivity or common support assumption must hold, which states that for every unit, the treatment assignment is not deterministic:

$$0 < P(D = d \mid X = x) < 1 \tag{4}$$

This particular definition is taken from Yao, Chu, Li, Li, Gao, and Zhang (2021). It means that for every combination of X in the population of interest, there should not be a deterministic probability to receive or not receive the treatment. Otherwise, estimating an effect would be meaningless because there is no counterfactual. In practice, this can be problematic when data is sparse. Even if the assumption holds for the population, it may be violated in the available data. This assumption can be partially tested. If a violation occurs, or if some combinations of X have very high or low propensity scores, there are different ways to address this issue such as cutting out the units with highest and lowest propensity score based on share of percentage (Stürmer, Rothman, Avorn, & Glynn, 2010) or percentile (Crump, Hotz, Imbens, & Mitnik, 2009)

Lastly, there is an assumption regarding the exogeneity of confounders, meaning treatment D shouldn't influence confounders X if they have an effect on Y. Further assumptions on the data's structure, such as linearity, can be significantly relaxed due to the use of newer machine learning algorithms. However, even with advanced algorithms that can capture complex data structures, the identifying assumptions of SUTVA, CIA, common support and exogeneity of confounders must still hold. If these assumptions are violated, bias may be introduced.

3.3 Data Generating Process

There will be two different Data Generating Processes (DGPs), referred to as: (a) Linear and (b) Non-Linear. The first DGP is linear in covariates. Additionally, there are several parameters that can vary, including λ, γ, and the number of observations (N). The structure of the Linear DGP is as follows:

$$
\begin{aligned}
Y_i &= \theta D_i + \gamma \pi_i + u_i \\
D_i &= f(D_i^c) \\
D_i^c &= \lambda \cdot (\sum_{j=1}^{P} X'_{ij}\beta_j) + v_i \qquad (5) \\
\pi_i &= \sum_{j=1}^{P} X'_{ij}\beta_j \\
\text{where} \quad \beta_j &= 1 - \frac{j}{P} \quad j = \{1, ..., P\}
\end{aligned}
$$

In this setup, the term $\gamma \cdot \pi_i$ is there to introduce confounding since they are based π_i which is influenced by the covariate X_{ij}. γ is there to influence the strength of π_i. u_i is an error term with $N(0, 1)$. θ represents the treatment effect, which is always set to 1. D_i is a binary treatment variable constructed through an intermediate step that transforms it from a continuous D_i^c to a binary variable using a threshold function, which returns 1 if the value of D_i^c is above its median m, and 0 if it is below.

$$
f(D_i^c) = \begin{cases} 1 & \text{if } D_i^c \geq m \\ 0 & \text{if } D_i^c < m \end{cases}
$$

D_i^c is calculated from β_j and the underlying random variables X_{ij} which are sampled from a standard normal distribution $N(0, 1)$. The subscript i represents the observations i, where $i = \{1, ..., N\}$, and j represents the confounder or independent variable, where $j = \{1, ..., P\}$. β_j is constructed in such a way that it introduces sparsity, meaning the higher the position j, the weaker the relationship between the random variable X_{ij} and the outcome Y_i. λ controls the overlap between treated and untreated observations. A lower λ increases the influence of the error term $v_i \sim N(0, 1)$. This setup has the advantage that the propensity score for each observation can be directly calculated using the following formula:

$$\begin{aligned}
\Pr(\text{treatment} = 1 \mid X) &= \Pr(h(X_{ij}) + v_i > m) \\
&= \Pr(u_i > m - h(X_{ij})) \\
&= 1 - F(m - h(X_{ij})) \\
&= F(h(X_{ij}) - m)
\end{aligned}$$

Where Pr() is the probability, $h(X_{ij})$ represents $D_i^c - v_i$, and $F()$ is the cumulative distribution function of the standard normal distribution. A lower λ in equation 5 leads to better overlap in the propensity scores between treated and untreated groups. In other words, the probability of receiving treatment becomes more similar for both the treated and untreated groups. This first DGP was designed to be able to examine how methods perform when the Conditional Independence Assumption (CIA) is violated, by assuring that covariates influence both the outcome and the treatment variable. In addition, this first DGP should create a simple linear setting where both covariates and parameters enter the outcome model and the link function in propensity model linearly.

The next DGP is non-linear in covariates and is therefore called the Non-Linear DGP. Some parts like the treatment assignment function as well as the role of γ and λ stay the same as in the Linear DGP and therefore the DGP stays linear in parameters. The structure is as followed:

$$
\begin{aligned}
Y_i &= \theta D_i + \gamma \pi_i + u_i \\
D_i &= f(D_i^c) \\
D_i^c &= \lambda * \left(3 \cdot \frac{D_i^*}{\sigma_{D_i^*}}\right) && (6) \\
D_i^* &= \sum_{j=1}^{P} |X_{ij} - r_j| && (7) \\
&\quad + X_{i1} X_{i(\frac{P}{2})} X_{iP} + \Big((X_{i2} - 3)(X_{i(\frac{P}{2})} + 3)\Big)^2 \\
&\quad + \ln\left(|X_{i3}|\right)\left(X_{i(P-1)} - 3\right) + v_i \\
\pi_i &= \sum_{j=1}^{P} |X_{ij} - r_j| \\
&\quad + X_{i1} X_{i(\frac{P}{2})} X_{iP} + \Big((X_{i2} - 3)(X_{i(\frac{P}{2})} + 3)\Big)^2 \\
&\quad + \ln\left(|X_{i3}|\right)\left(X_{i(P-1)} - 3\right)
\end{aligned}
$$

The final equation to construct Y_i remains the same as in the linear case, but the construction of the treatment variable D_i and the confounding part π_i changes. First, D_i^* is designed, which depends on the absolute term of $|X_{ij} - r_j|$, that decreases for each confounder due to an increase in r_j. Here, r_j is a vector defined as $r_j = 1 - (1/P) \cdot k$, with $k = \{1, 2, \ldots, P\}$, and P is the number of confounders. The absolute term is followed by interaction terms, where the subscripts indicate which confounders are used (e.g., X_{i1} refers to the first confounder, and $X_{i(P/2)}$ refers to the confounder in the middle of the list, such as X_{i10} if there are 20 confounders). D_i^c is then calculated by multiplying D_i^* by 3 and dividing by the standard deviation σ of D_i^*. Finally, like in the linear DGP, D_i^c undergoes a binary transformation. π_i is constructed in a similar manner but without the binary transformation and the transformation in 6. Each X_i is drawn from a standard normal distribution $N(0, 1)$.

The purpose of these two DGPs is to explore variability across different settings and provide insights for further research if anything noteworthy is discovered. Throughout the simulations, certain parameters can be adjusted, such as γ, which controls the strength of confounding. Additionally, the number of observations (N) and λ, which influences the overlap between the propensity scores of treated and untreated units as well as the strength of confounding, can also be varied. The combinations of these parameters for the linear and non-linear DGPs are: $\gamma = [0.1, 0.5]$, $N = [2000, 8000]$, and $\lambda = [0.1, 0.5, 1.2]$. The number of confounders is kept at 20 for all simulations. The combinations are displayed in Table 1 with all the varying parameters.

Table 1: Combinations of Parameters for the Simulation.

linear	N obs	gamma	lambda	θ	P conf
False	2000	0.5	0.5	1	20
False	2000	0.5	1.2	1	20
False	2000	0.1	0.1	1	20
False	8000	0.5	0.5	1	20
False	8000	0.5	1.2	1	20
False	8000	0.1	0.1	1	20
True	2000	0.5	0.1	1	20
True	2000	0.5	0.5	1	20
True	2000	0.1	0.1	1	20
True	8000	0.5	0.1	1	20
True	8000	0.5	0.5	1	20
True	8000	0.1	0.1	1	20

Column Linear displays true if Linear DGP and false if Non-Linear DPG. N obs stands for number of observations and P conf stands for number of confounders.

Furthermore, the number of Monte Carlo simulations conducted varies with the DGPs and the number of confounders. Either 500 or 300 iterations were conducted, depending on whether N was 2000 or 8000. This difference stems from the idea that with a higher number of observations, the estimation becomes more accurate and less variable.

This paper does not include a DGP with treatment effect heterogeneity, as the focus is on the impact of CIA violations on the ATE, which reflects the effect across the entire population.

Since this paper tests for violations of the CIA, it is important to track the level of confounding within the dataset. Confounding arises when X_{ij} influences both Y_i and D_i. Two methods are used to assess the level of confounding, where the first is stated in this section and will give an visualized overview of the correlations. In addition, to giving insights into correlation it has the additional benefit of of visualizing the data structure. The second approach to track confounding will be introduced in section 3.4. As for the visualization of confounding in Figure 1, a Pearson correlation for each variable X_i with Y_i and D_i is displayed. These correlations represent the mean over 500 iterations for each DGP. An extensive graph displaying all the confounders is attached in the appendix 5.

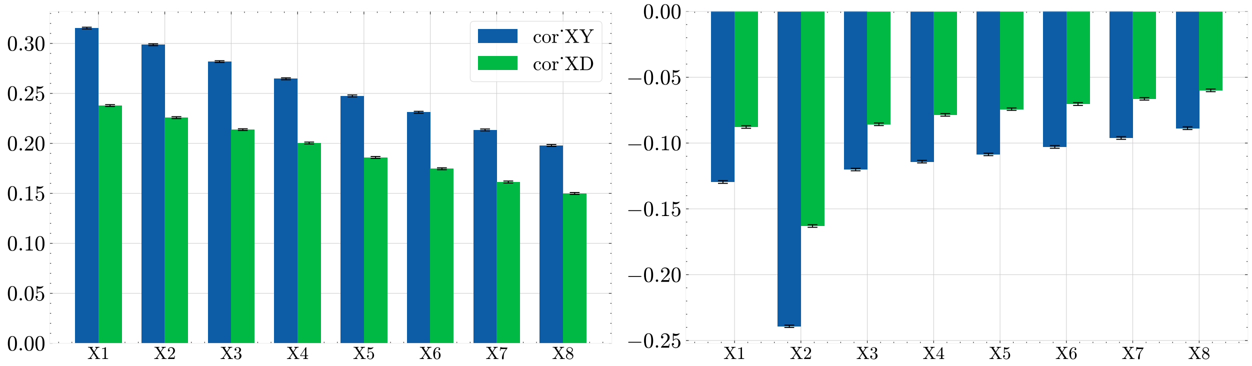


Figure 1: Correlation between X, D and X, Y.
X-axis displays the confounders. For each X_j, the correlation between X and Y (blue) and X and D (green) is shown. The correlations are the mean aggregated over an iteration of 500 simulations. Black spreads at the top of the bars represent the standard error over the iterations. The left plot represents Linear DGP and the right Non-Linear DGP. Specifications for the DGP are: Observations = 2000, $\gamma = 0.5$, confounders = 20, θ is always 1, and λ is 0.5 for the Linear DGP and 1.2 for the Non-Linear DGP.

The results show that, for the Linear DGP, the correlation between X and Y is higher than that between X and D, which could be due to the additional transformation as well as the λ value. However, the correlation is likely still sufficient to impact ATE estimation and introduce bias, when a confounder is omitted. The correlation decreases as the position of the confounder increases, which is due to the sparsity introduced in the DGP. For the Non-Linear DGP, the relationship between X and Y, and X and D, shows a decreasing negative correlation as the position of X_j increases, aligning with the DGP's design as seen from the absolute value reduction effect (i.e., Equation 13). Some outliers such as X_2 indicate variables involved in interaction terms. Here again, the difference in the strength of the correlation between X and Y, and X and D, could be due to λ as well as the additional transformations in Equation 13.

Since this paper focuses on sensitivity to CIA violations, the other identifying assumptions must hold. First, SUTVA is typically valid in simulation contexts like this, as there are no hidden treatments (all treatments are either 0 or 1), and one unit's treatment assignment does not affect another unit's potential outcome.

The common support assumption requires further investigation. A good way to assess this property is to plot the propensity scores for both treated and untreated observations (Garrido et al., 2014). The propensity score represents the probability of being treated given some independent variables, $\Pr(\text{treatment} = 1 \mid X)$. If the propensity scores exhibit good overlap in the graph—meaning the probability of being treated is similar for both treated and untreated groups—then the common support assumption is likely to hold. As explained above, the DGP of this paper is constructed such that the propensity score is directly accessible. Different levels of the parameter λ lead to varying degrees of overlap (see Table 1 for the different levels of λ). Two example propensity score graphs in Figure 2 are shown for the Non-Linear DGP. They illustrate that a higher λ results in worse overlap, as previously explained. The overlap for $\lambda = 1.2$ is significantly lower than for $\lambda = 0.5$, suggesting that the common support assumption might be violated to some extent. The Propensity score visualizations for all DGPs and λ can be found in the appendix 5.

Even though the main goal of the paper is to investigate the violation of the CIA, and not common support, it is still interesting to observe how the methods react to these different specifications. Furthermore, a lower overlap in this setting aligns with a stronger confounding effect, as the underlying X variables have a greater influence on treatment selection when λ is increased. This ties in with the goal of investigating CIA violations.

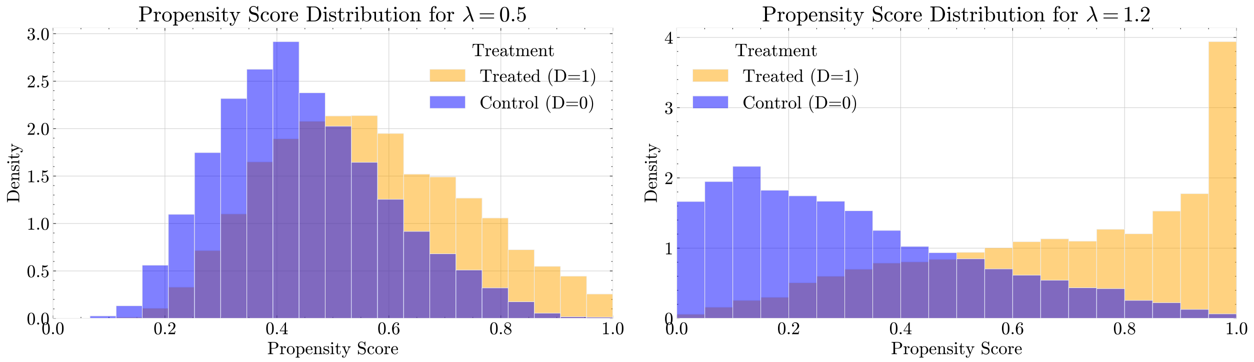


Figure 2: Visualization of Propensity Scores

Display of propensity scores for treated and untreated groups. The Y-axis displays the density. These plots are based on the Non-Linear DGP with different λ values.

The CIA is assumed to hold for the full dataset. Since all variables used to construct Y and D are present in the initial dataset, the CIA should hold. Lastly, the exogeneity assumption for the confounders is satisfied, as D does not influence any of the confounders in the DGP.

3.4 Violation of CIA

After presenting the DGP, the detailed methodology used to assess the impact of violating the Conditional Independence Assumption (CIA) is described. As noted, the CIA holds when all confounders are present in the dataset. To simulate a violation, we manipulate the data by omitting some confounders. The manipulated datasets are then used to evaluate the performance of the machine learning methods.

In an attempt to approximate and compare the strength of the impact caused by omitting confounders, the following measure is introduced: the correlation between the error term in the DGP and the remaining term that determines the dependent variable, Y. These calculations are important for this paper, serving as a bridge between the simulation results and the level of confounding strength. The construction of the correlation measure proceeds as follows. Consider the DGP with all underlying random variables, where the function $f(X, D)$ encapsulates all the factors influencing the dependent variable except for the error term ϵ:

$$Y = f(X, D) + \epsilon \tag{8}$$

When no variables are omitted, the correlation between $f(X, D)$ and ϵ should be zero, given that $\epsilon \sim N(0, 1)$. However, when some variables are omitted, there will be missing parts in $f(X, D)$, leading to a new expression $f(\tilde{X}, D)$. The missing part must be included in the new error term $\tilde{\epsilon}$, as all information is still within the dependent variable Y. The new equation with omitted variables is:

$$\begin{aligned} Y &= f(\tilde{X}, D) + \tilde{\epsilon} \\ &= f(\tilde{X}, D) + \epsilon + \text{missing part of } f(X, D) \end{aligned}$$

The key step is now to calculate the correlation between $\tilde{\epsilon}$ and $f(\tilde{X}, D)$. Since the structure of the missing part of $f(X, D)$ is known from the DGP function, and ϵ is also known, one can derive $\tilde{\epsilon}$. Then, $f(\tilde{X}, D)$ is derived by simply subtracting $\tilde{\epsilon}$ from the dependent variable Y. This leads us to the final measure of correlation:

$$\text{correlation measure} = \text{corr}(f(\tilde{X}, D), \tilde{\epsilon}) \tag{9}$$

Next, we apply this correlation measure to in this paper called scenarios, where each scenario represents a case in which certain confounders are omitted. For example, one scenario involves omitting a single strong confounder, while in another, 10 relatively weak confounders are omitted. The scenarios differ slightly between the linear and non-linear DGPs. However, for the later analysis, these differences do not have a significant impact. Table 2 presents the correlation measure just described in combination with the scenarios. All scenarios, except the one where all confounders are omitted, are used in the simulation. It is important to once again note that this measure is just an approximation, and in the Non-Linear DGP case, it may not be the most accurate indicator of true confounding strength.

In Table 2, we observe that the correlation measure is almost zero when all confounders are included, which is expected. Additionally, when omitting all confounders, the correlation is significantly higher. When omitting individual confounders, the correlation is not particularly high, but differences between confounders are evident. Also note that there are multiple single omitted confounders which are labeled with S for strong or W for weak. This translates into picking three different confounders as representatives of strong or weak. For the Linear DGP there is a Table 18 in the appendix.

In addition to omitting confounders according to the predefined scenarios, another simulation approach was conducted where confounders were randomly omitted. Different shares of confounders were omitted, ranging from 10% to 80% (i.e., 0.1, 0.2, 0.4, 0.6, 0.8). For example, if 10% of the confounders were omitted out of 20 confounders, 2 confounders would be omitted in each iteration of the Monte Carlo simulation, and the bias would be measured. Since these simulations were

Table 2: Different Scenarios of Confounding Omitting

Omitted Confounders	Scenario	Correlation
	No Omit	0.000475
X1	S	0.012211
X2	S	0.030348
X3	S	0.056560
X17	W	0.004613
X18	W	0.004675
X2, X18	S W	0.033531
X1, X2	S S	0.038831
X17, X18	W W	0.008506
X8-X9, X11-X18	10W	0.037891
X19, X1-X4, X14-X18	5S 5W	0.696785
X19, X1-X9, X19	10S	0.705076
X1-X20	All	0.830386
X1-X2, X4-X18, X20	All but 2S	0.254961

Values in Omitted Confounders correspond to confounders position Xj where j = {1,. . . ,P}. In Scenario, S and W correspond to Strong and Weak. The numbers in front of letters indicate the number of variables omitted. I.e. 5S meaning 5 strong variables.

conducted earlier in the thesis process, the DGP and other settings differ slightly. Therefore, the results will only be displayed in the Appendix 5.

3.5 Estimation Methods

In this section the 5 methods that are used to estimate the ATE are introduced and technically described (Linear Regression, DML, T-learner, X-learner, XBCF). It is important to mention that some estimation methods have underlying nuisance functions such as outcome and propensity function. Since the goal was to keep them as simple as possible regarding the configurations some nuisance functions differ from one another, in particular the nuisance function for propensity score of the X-Learner uses elastic net instead of XGBoost. The differences are made clear in Table 3

Some of the estimators are primarily designed to estimate heterogeneous effects. This paper's DGP and research question focus on the ATE and therefore do not utilize the additional capabilities of these estimators for CATE estimation. However, it is still interesting to see how these estimators perform with homogeneous effects, as usually the underlying data structure in empirical research is not known.

Table 3: Comparison of Machine Learning Methods.

Method	**Description**	**Python Package**	**Underlying Model**
Linear Regression OLS	A basic linear model to estimate treatment effects.	statsmodels	Ordinary Least Squares (OLS)
DML	Double machine learning using cross-fitting and two nuisance functions combined in a final model.	doubleml	Outcome model: XGBoost, Propensity score model: XGBoost
T-learner	Estimates separate outcome nuisance function for treated and control groups.	causalml	Outcome model: XGBoost
X-learner	Multiple step approach with imputation of treatment effects.	causalml	Outcome model: XGBoost, Propensity score model: Elastic Net
XBCF	Bayesian causal forest method with BART model as foundation and elements that enhance computational time.	xbcausalforest	Bayesian regression trees

Ordinary Least Squares (OLS)

The first method is not a machine learning estimation method but rather a simple linear regression estimated by OLS out of the textbook. The python package statsmodels is used for the implementation(Seabold, Skipper, & Perktold, 2010).

Debiased Machine Learner (DML)

This method involves the application of the DML approach introduced by Chernozhukov et al. (2018). This paper presents a new method for estimating the treatment effect in high-dimensional settings. Traditional machine learning techniques, while effective in prediction tasks, utilize regularization to mitigate overfitting and bias. However, when these methods are naively applied to estimate θ, they can introduce significant bias and fail to achieve $N^{1/2}$ consistency.

To overcome these limitations, the authors propose two key components: (1) the use of Neyman-orthogonal moments or scores, and (2) the application of cross-fitting, a robust data-splitting technique. The authors demonstrate that their approach results in point estimators that are $N^{1/2}$ consistent and asymptotically normally distributed. The score function employed in this paper, which is the function for the estimation of ATE, is based on the doubly robust augmented inverse probability weighting (AIPW) method, first introduced by Robins, Rotnitzky, and Zhao (1994) and Robins and Rotnitzky (1995). Chernozhukov et al. (2018) also recommend this score function, with the primary innovation being the incorporation of cross-fitting. The method is described more technically by the following function:

$$\widehat{ATE} = \frac{1}{N}\sum_{i=1}^{N} \hat{\mu}_{-i}(1, X_i) - \hat{\mu}_{-i}(0, X_i) + d_i \frac{Y_i - \hat{\mu}_{-i}(1, X_i)}{\hat{p}_{-i}(X_i)} - (1 - d_i)\frac{Y_i - \hat{\mu}_{-i}(0, X_i)}{1 - \hat{p}_{-i}(X_i)} \tag{10}$$

The two nuisance functions, for estimating outcome μ and propensity p are estimated with machine learning methods. The $-i$ shows the cross fitting process in which all data but i (over k fold data) is used to estimate the nuisance functions. Data i is then used as input to the score function which estimates the ATE. The estimation of nuisance and score function is repeated over k folds and finally the ATE is averaged out. In this paper a 3 fold cross-fit approach was used.

This method was implemented in a python environment with a package called DoubleML (Bach, Chernozhukov, Kurz, & Spindler, 2022).

T-learner

Next, a simpler method known as the T-learner is applied, which is sometimes categorized under the group of meta-learners. This method estimates the ATE as follows:

$$\widehat{ATE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{\mu}_1(X_i) - \hat{\mu}_0(X_i)) \tag{11}$$

In here only the nuisance function for the outcome $\hat{\mu}(x)$ is modeled and the propensity score does not come into play. Two outcome functions are estimated: one for the treatment group $\hat{\mu}_1(X) = \hat{E}[Y|D = 1, X]$ and another for the control group $\hat{\mu}_0(X) = \hat{E}[Y|D = 0, X]$. These functions are estimated using the subsamples of treatment and outcome groups. In other words the treatment groups outcome function is trained separately from the control groups. The final ATE estimation function however, uses all the observations combined. Any arbitrary estimation method such as XGboost or OLS can be employed for the estimation of these nuisance functions. (Künzel, Sekhon, Bickel, & Yu, 2019)

To implement this method the causalML package in python was used.

X-learner

This method also belongs to the group of meta-learners and was proposed in Künzel, Sekhon, Bickel, and Yu (2019). The method consists of a three step approach beginning similar to the T-learner with estimating two outcome functions with the subsamples of treatment and control group:

$$\hat{\mu}_1(X) = \hat{E}[Y|D = 1, X]$$
$$\hat{\mu}_0(X) = \hat{E}[Y|D = 0, X]$$

Next, these nuisance functions are used to impute the treatment effect for the treated group by taking the difference between the observed outcomes of the treated group and the estimated nuisance function for the untreated group, applied to the treated group data, $(Y_i, X_i)_{D_{i=1}}$. The author calls $\hat{B}_i$ the imputed treatment effects. The same approach is used to estimate the imputed treatment effect for the untreated group, by switching the roles of the data and the nuisance functions, $(Y_i, X_i)_{D_{i=0}}$.

$$\hat{B}_{1i} := Y_i - \hat{\mu}_0(X_i)$$

$$\hat{B}_{0i} := \hat{\mu}_1(X_i) - Y_i$$

With the help of the imputed treatment effects and any arbitrary machine learning method one can estimate $\hat{\tau}_1(X)$ respectively $\hat{\tau}_0(X)$. Where $\hat{\tau}(X)$ is the treatment effect and is estimated using the imputed treatment effect as the outcome variable and X as the independent variables. Similar to the first step, $\hat{\tau}(X)$ are estimated with the subsample of treated respectively untreated groups.

Finally, obtain the CATE by building in a weighting function to each treatment effect estimated in the previous step. The weighting function ($g \in [0, 1]$) can for example be obtained by using the propensity score.

$$\widehat{CATE}(X_i) = g(X_i)\hat{\tau}_0(X_i) + (1 - g(X_i))\hat{\tau}_1(X_i))$$

By averaging out the CATE in the end we can obtain the ATE, which is the estimate this paper is interested in.

To implement this method the causalML package in python was used.

Accelerated Bayesian Causal Forest (XBCF)

The next model uses an adaptation of the Bayesian Additive Regression Tree (BART) as the underlying machine learner. Unlike the previously mentioned methods, such as the X-Learner, where the underlying machine learning methods are interchangeable, BART is fixed as the foundation. BART, introduced by Chipman, George, and McCulloch, 2010, is, in simple terms, a tree ensemble learner that combines multiple trees to estimate an outcome Y. BART employs so-called priors to ensure that each individual tree has only a small impact on the overall estimated effect. These priors can be interpreted as components that perform regularization within the trees.

Building on the BART estimator, Hahn, Murray, and Carvalho, 2020 proposed the Bayesian Causal Forest (BCF) to address some of BART's shortcomings in causal inference settings, as BART was originally designed for prediction tasks. The BCF model introduces several modifications to the standard BART model. It uses two separate BART models instead of a single one for estimating the outcome Y. Specifically, one tree ensemble estimates the prognostic score function, while the other estimates the treatment function. This approach allows the two models to be regularized individually through their respective priors. Additionally, the BCF model incorporates an estimate of the propensity score as an input into the prognostic score function and applies scaling factors to both the prognostic and treatment functions.

Krantsevich, Jingyu, and Richard, 2023 introduced further refinements to the BCF model, resulting in the XBCF model, represented as follows:

$$\begin{aligned} Y_i &= a\varphi(X_i, \hat{p}_i) + b_{z_i}\tilde{\tau}(X_i) + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma_{z_i}^2) \\ a &\sim N(0,1), \quad b_0, b_1 \sim N(0, 1/2) \end{aligned}$$

Here, $\varphi(X_i, \hat{p}_i)$ represents the prognostic function, scaled by factor a, while $\tilde{\tau}(X_i)$ is the treatment function. $\hat{p}_i$ is the mentioned propensity score. The treatment function $\tilde{\tau}(X_i)$ is scaled by factor b_{z_i}. This scaling factor improves the accuracy of the treatment effect estimates. The actual treatment effects are calculated as $(b_1 - b_0)\hat{\tau}(X)$, and the ATE is obtained by averaging these treatment effects.

A key distinction between BCF and XBCF lies in the error term ϵ and its standard deviation

σ_{z_i}. In XBCF, σ_{z_i} allows the standard deviation to differ between the control and treatment groups, whereas in BCF, the standard deviation is shared across both groups (i.e., σ).

Additionally, Krantsevich, Jingyu, and Richard, 2023 improved the computational performance of BCF by incorporating advancements from the adaptation of the BART model to the XBART model (He & Hahn, 2020). These improvements include a more efficient tree-building approach called recursive partitioning, where each tree is built recursively based on a stochastic process. While the number of trees remains unchanged, the speed of the ensemble process increases significantly.

To implement this method the xbcausalforest package in python was used

3.6 Performance Measures

To compare the performance of the algorithms, several evaluation measures need to be introduced. These measures are essential for assessing the accuracy, stability, and sensitivity of the ATE estimates (denoted as θ) produced by the different algorithms. Specifically, we will focus on bias, absolute bias and root mean squared error (RMSE). While coverage rate is often included in similar simulation studies, it has been excluded here because confidence intervals are not well-established for some of the machine learning methods under consideration. Additionally, bootstrapping is not a viable option to get the confidence intervals, since the simulation study would be too computational intensive with this approach.

The definitions of these performance measures are based on the work of Morris, White, and Crowther (2019), where each measure is explained in terms of its Definition, Estimate, and Monte Carlo Standard Error (SE) of the Estimate. The Definition refers to the theoretical value of a measure, such as bias. The Estimate is the sample-based approximation of this measure, calculated across all Monte Carlo iterations. The Monte Carlo SE of Estimate represents the standard error associated with these approximations. An overview is given in Table 4 where M is the number of Monte Carlo simulations. In there $\overline{\text{Bias}}$ and $\overline{\text{Abs Bias}}$ stand for the average of the measures estimated over all iterations $r = \{1, \ldots, M\}$.

Table 4: Summary of Measures used in the Analysis.

Measure	**Definition**	**Estimate**	**Monte Carlo SE of Estimate**
Bias	$E[\hat{\theta}] - \theta$	$\frac{1}{M}\sum_{r=1}^{M}\hat{\theta}_r - \theta$	$\sqrt{\frac{1}{M(M-1)}\sum_{r=1}^{M}(\hat{\theta}_r - \overline{\text{Bias}})^2}$
Abs Bias	$E[\left\|\hat{\theta}\right\|] - \theta$	$\frac{1}{M}\sum_{r=r}^{M}\left\|\hat{\theta}_r - \theta\right\|$	$\sqrt{\frac{1}{M(M-1)}\sum_{r=1}^{M}\left(\left\|\hat{\theta}_r - \theta\right\| - \overline{\text{Abs Bias}}\right)^2}$
RMSE	$\sqrt{E[(\hat{\theta} - \theta)^2]}$	$\sqrt{\frac{1}{M}\sum_{r=r}^{M}(\hat{\theta}_r - \theta)^2}$	—

4 Result

4.1 Results

First, we examine how the correlation measure, which partially defines the scenarios, influences the bias observed in the simulation results. In Figure 3, a clear positive relationship between the correlation measure and bias with the exception of DML is evident. The plot displays each estimation method in a different color, with all simulation configurations combined into a single plot (all possible configurations are listed in Table 1).

However, it has to be noted that correlation measuere is only an approximation for the strength of the confounding and the relationship between the measure and bias is not linear. I.e. for the Non-Linear DGP when omitting the strong confounder X_{i3} then the bias is significantly higher compared to the other two strong confounders omitted (X_{i1}, X_{i2}) even though the correlation measure does not have that much of a difference. A good way to explore this is to look at the two tables of correlation measure 2 and a simulation 16 with $N_{\text{Obs}} = 30'000$ (note that this is the only simulation conducted with this many observations).

The visualization also provides insight into the performance of each estimation method in the simulation. For example, in many cases, DML tends to underestimate the treatment effect, resulting in a negative bias. Additionally, Linear Regression exhibits a very high bias in some cases, likely due to its misspecification for the Non-Linear DGP. To further interpret the simulation results, additional tables are presented in this section, covering both the Linear and Non-Linear DGPs.

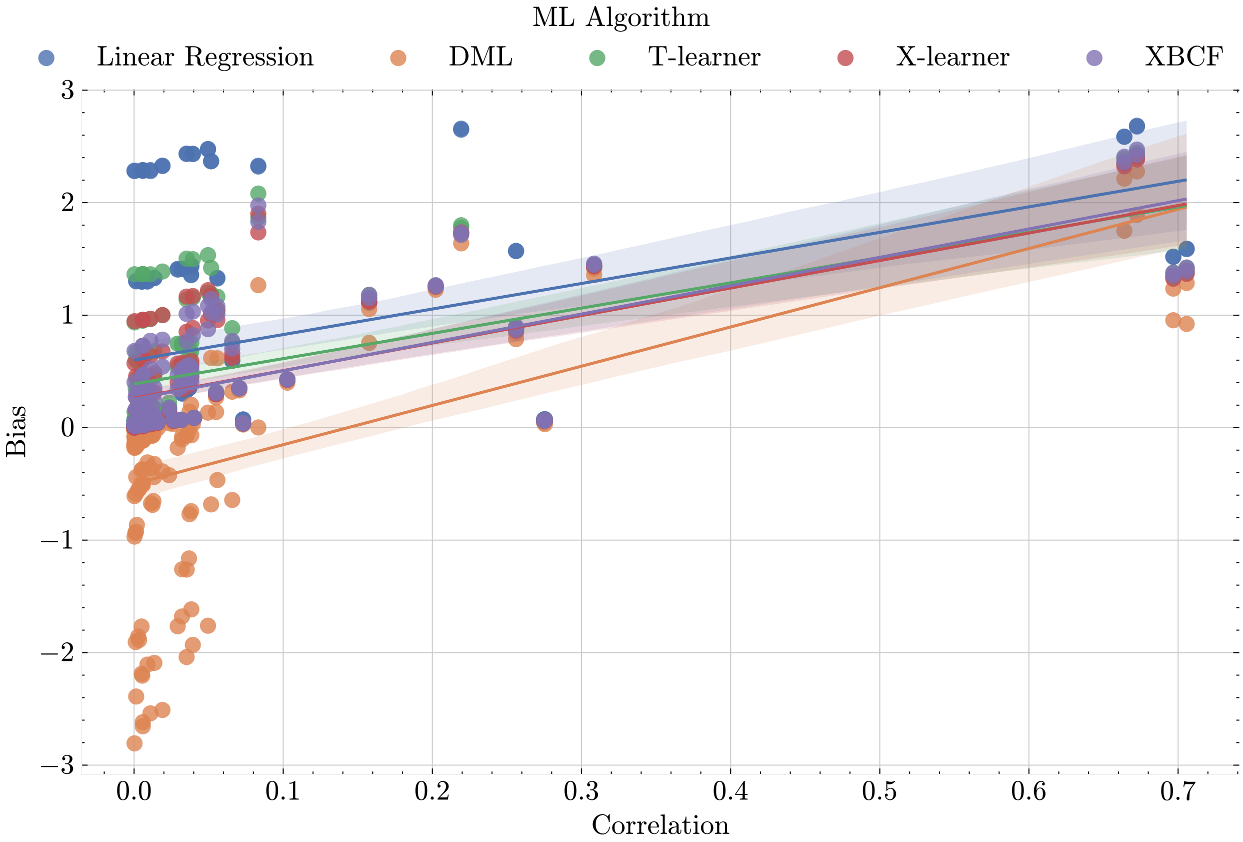


Figure 3: Relationship between Correlation Measure and Bias
The X-axis represents the correlation measure, while the Y-axis shows the bias. The lines on the plot correspond to regressions. Each dot corresponds to a possible specification of the DGP, an estimation method as well as a different scenario.

Table 5: Bias for Linear DGP.

		Bias					Total
	Scenario	No Omit	S_X1	W_X15	10W	All but	2S
N Obs	Method						
2000	Linear Regression	-0.000	0.070	0.010	0.077	0.352	0.508
	DML	-0.968	-0.687	-0.924	-0.439	0.331	3.348
	T-learner	0.141	0.181	0.146	0.148	0.353	0.969
	X-learner	0.026	0.091	0.034	0.086	0.351	0.588
	XBCF	0.051	0.115	0.057	0.105	0.352	0.681
8000	Linear Regression	0.002	0.072	0.003	0.077	0.349	0.504
	DML	-0.180	-0.071	-0.162	-0.007	0.347	0.766
	T-learner	0.079	0.128	0.075	0.113	0.349	0.745
	X-learner	0.011	0.077	0.011	0.081	0.348	0.527
	XBCF	0.032	0.098	0.032	0.094	0.351	0.607
specification: lambda = 0.1 gamma = 0.5							
2000	Linear Regression	0.002	0.331	0.027	0.360	1.248	1.968
	DML	-1.906	-1.261	-1.887	-0.769	1.226	7.050
	T-learner	0.601	0.743	0.597	0.648	1.261	3.849
	X-learner	0.157	0.427	0.167	0.409	1.256	2.417
	XBCF	0.272	0.548	0.282	0.501	1.266	2.869
8000	Linear Regression	0.005	0.339	0.023	0.359	1.248	1.974
	DML	-0.590	-0.059	-0.547	0.142	1.260	2.597
	T-learner	0.350	0.573	0.354	0.520	1.257	3.054
	X-learner	0.043	0.370	0.060	0.388	1.256	2.116
	XBCF	0.162	0.471	0.172	0.441	1.258	2.503
specification: lambda = 0.5 gamma = 0.5							
2000	Linear Regression	0.002	0.011	0.000	0.016	0.070	0.099
	DML	-0.166	-0.115	-0.128	-0.066	0.071	0.546
	T-learner	0.023	0.027	0.019	0.024	0.070	0.163
	X-learner	0.007	0.015	0.005	0.016	0.070	0.112
	XBCF	0.030	0.035	0.027	0.031	0.068	0.191
8000	Linear Regression	-0.002	0.011	0.002	0.016	0.069	0.100
	DML	-0.023	-0.010	-0.016	0.006	0.069	0.124
	T-learner	0.007	0.018	0.011	0.020	0.068	0.124
	X-learner	-0.000	0.012	0.003	0.016	0.068	0.100
	XBCF	0.017	0.028	0.020	0.027	0.070	0.162
specification: lambda = 0.1 gamma = 0.1							

The first Table 5 displays results for the Linear DGP. Only a handful of scenarios are shown due to space limitations; a more comprehensive view of the results is provided in Appendix 5. In this table, we observe that Linear Regression performs very well in estimating the treatment effect. When no confounders are omitted, it is the most accurate estimation method. Even as omitted confounding strength increases across scenarios, Linear Regression consistently remains the most reliable estimator. This finding is confirmed by the aggregated bias across all displayed scenarios.

The DML method tends to underestimate the treatment effect in many scenarios, leading to a negative bias. It also exhibits the highest aggregated bias in many cases, as indicated in the final "Total" column. Moreover, DML does not show a clear relationship between increased confounding omission and bias, in contradiction to other observed methods. For instance, in Scenario S, where one strong confounder is omitted, the DML estimation is more accurate than when no confounders or only a weak confounder are omitted. This behavior is consistent across all simulation specifications. Additionally, when 10 weak confounders are omitted, the DML estimation becomes more accurate. This aligns with initial insights from the plot above, where many accurate DML estimates appear when the correlation measure is between 0 and 0.1.

The T-Learner, on the other hand, ranks second to last in terms of estimator accuracy. It significantly underperforms relative to the X-Learner, XBCF, and Linear Regression estimators. The X-Learner demonstrates high accuracy, generally outperforming other methods (except for Linear Regression) in most cases. When the number of observations increases to 8000, the X-Learner's performance approaches almost that of Linear Regression.

Overall, most estimators show improved accuracy with a larger number of observations, which is expected. Additionally, the parameters γ and λ affect accuracy, with higher values generally reducing estimator accuracy.

Table 6 displays the results for the Non-Linear DGP. A different pattern emerges here regarding the Linear Regression estimator, which is clearly misspecified, as it struggles to estimate the true effect accurately even with all confounders present and a relatively high number of observations. It performs the worst in estimating the treatment effect. Interestingly, the bias for Linear Regression does not increase substantially when comparing the "No Omit" scenario to "All but 2S." For example, in the scenario where $\lambda = 0.5$, $\gamma = 0.5$, and $N_{\text{Obs}} = 8000$, the bias increases by approximately 0.27, while for the same configuration, the bias for XBCF increases by 0.7.

The DML estimation method shows mixed results. In many cases, it outperforms other machine learning estimators especially with higher numbers of observations. However, DML appears somewhat unstable, as its bias fluctuates quickly from negative to positive. For instance, by omitting a single strong confounder, DML's bias shifts from very negative to positive in the case where $N_{\text{Obs}} = 8000$ and $\gamma = 0.5$. Additionally, it is noteworthy that omitting strong confounders sometimes leads to a more accurate estimate compared to including all confounders (e.g., see Tables 14 and 9 for $\lambda = 0.5$, $\gamma = 0.5$, and $N_{\text{Obs}} = 8000$ in scenarios such as S_X1, S_X2, etc.). Similar patterns are not observed with other ML methods, indicating that this instability may be unique to the DML estimator and its interaction with this DGP rather than a general characteristic of the DGP itself.

Another interesting observation regarding DML is that it occasionally performs better with a lower number of observations. For instance, with $\lambda = 1.2$ and $\gamma = 0.5$, the bias increases from 0.002 to 1.2 in scenario S. When comparing this with other scenarios, this pattern only seems to occur when the variable X_{i3} is omitted. This trend is also observed in the more detailed tables in the appendix: as soon as X_{i3} is omitted, such as when 10 strong confounders are omitted, the bias increases with a higher number of observations. However, if X_{i3} is not omitted, the more intuitive behavior is observed, with bias decreasing as the number of observations increases. It is worth noting that the variable X_{i3} is not always omitted in scenario where a strong confounder is omitted (for instance, in scenario S S only X_{i1} and X_{i2} are omitted). This specific pattern does not appear in other machine learning methods.

An additional observation about DML is found in Table 10 for scenario S_X3 (where $\lambda = 1.2$, $\gamma = 0.5$, and $N_{\text{Obs}} = 2000$). Here, the absolute bias is 0.4, which is significantly higher than the average bias of 0.002. This suggests that the DML method has considerable variance in estimating the treatment effect. As a final point, DML's precision improves significantly with an increased number of observations compared to other methods, going from one of the poorest performers to one of the best as sample size increases.

Next, the T-Learner generally performs worse than the other methods (except for Linear Regression). The X-Learner, on the other hand, appears to perform better, showing results very similar to XBCF. Both XBCF and T-Learner display similar sensitivity to the omission of confounders; however, there are cases where XBCF shows a greater increase in bias with the omission of certain confounders (e.g., in scenario S), while in other scenarios, the X-Learner reacts more (e.g., in scenario W). In the aggregated results marked ”Total” in the table, XBCF performs slightly better than the other estimators.

Overall, it can be observed once again that the parameters λ and γ significantly impact the performance of the methods. When these parameters are set to very low values (i.e., $\lambda = 0.1$ and $\gamma = 0.1$), all estimators perform relatively well, and the omission of confounders has a low impact on estimation accuracy.

The Monte Carlo Standard Errors (SE) for bias, summarized in Table 7, indicate that DML consistently has the highest Monte Carlo SE across both DGPs for the bias measure, suggesting greater variability in its performance.

In terms of computational efficiency measures in seconds, Linear Regression is by far the fastest method, with the T-Learner being the second fastest but still approximately 70 times slower in the Linear DGP. The X-Learner and DML take a similar amount of time to estimate the ATE, while XBCF requires the longest computation time. These results are summarized in Table 8, which presents the computational times for each method. The built-in checks in the simulation code confirm that all calculations for each ML method were executed as expected, with no missing data reported in any of the simulations.

Table 6: Bias for Non-Linear DGP.

		Bias					Total
	Scenario	No Omit	S_X3	W_X17	10W	All but 2S	
N Obs	Method						
2000	Linear Regression	1.304	1.327	1.300	1.355	1.571	6.857
	DML	-2.390	-0.466	-2.204	-0.743	0.787	6.589
	T-learner	0.672	1.164	0.665	0.674	0.898	4.073
	X-learner	0.459	1.050	0.459	0.550	0.875	3.393
	XBCF	0.309	1.079	0.329	0.535	0.894	3.146
8000	Linear Regression	1.301	1.332	1.304	1.360	1.570	6.867
	DML	-0.438	0.617	-0.371	0.204	0.838	2.468
	T-learner	0.432	1.032	0.438	0.540	0.876	3.317
	X-learner	0.267	0.956	0.283	0.462	0.863	2.832
	XBCF	0.169	1.006	0.199	0.468	0.871	2.713
specification: lambda = 0.5 gamma = 0.5							
2000	Linear Regression	2.281	2.328	2.283	2.369	2.649	11.910
	DML	-2.806	0.002	-2.618	-0.682	1.638	7.746
	T-learner	1.363	2.081	1.366	1.420	1.799	8.030
	X-learner	0.946	1.902	0.960	1.172	1.736	6.716
	XBCF	0.682	1.975	0.729	1.143	1.734	6.264
8000	Linear Regression	2.284	2.322	2.288	2.368	2.658	11.920
	DML	-0.609	1.267	-0.508	0.619	1.725	4.727
	T-learner	0.936	1.865	0.952	1.189	1.776	6.719
	X-learner	0.572	1.736	0.608	1.009	1.730	5.655
	XBCF	0.404	1.830	0.469	1.022	1.713	5.438
specification: lambda = 1.2 gamma = 0.5							
2000	Linear Regression	0.058	0.058	0.062	0.065	0.073	0.316
	DML	-0.081	-0.005	-0.088	-0.027	0.028	0.229
	T-learner	0.027	0.049	0.031	0.032	0.039	0.177
	X-learner	0.019	0.044	0.024	0.028	0.038	0.153
	XBCF	0.022	0.049	0.026	0.030	0.038	0.164
8000	Linear Regression	0.062	0.060	0.059	0.063	0.072	0.316
	DML	-0.009	0.036	-0.016	0.006	0.034	0.100
	T-learner	0.018	0.046	0.016	0.023	0.036	0.139
	X-learner	0.013	0.044	0.011	0.021	0.036	0.124
	XBCF	0.020	0.049	0.018	0.026	0.038	0.150
specification: lambda = 0.1 gamma = 0.1							

Table 7: Monte Carle SE for Bias

DGP	Scenario Method	No Omit	S	W	10W	All but 2S	Total
Linear	Linear Regression	0.002	0.002	0.002	0.002	0.002	0.010
	DML	0.010	0.010	0.010	0.010	0.007	0.047
	T-learner	0.002	0.002	0.002	0.002	0.002	0.010
	X-learner	0.002	0.002	0.002	0.002	0.002	0.011
	XBCF	0.002	0.002	0.002	0.002	0.002	0.010
Non-Linear	Linear Regression	0.003	0.003	0.003	0.003	0.003	0.014
	DML	0.014	0.014	0.014	0.013	0.008	0.063
	T-learner	0.002	0.003	0.002	0.003	0.003	0.013
	X-learner	0.002	0.003	0.002	0.003	0.003	0.013
	XBCF	0.002	0.003	0.003	0.003	0.003	0.013

Monte Carlo SE aggregated by mean for all specifications DGPs grouped by Linear and Non-Linear DGP.

Table 8: Execution Time for Methods

DGP Method	Linear	Non-Linear
Linear Regression	0.007	0.005
DML	2.045	2.068
T-learner	0.469	0.496
X-learner	2.306	2.396
XBCF	4.215	4.983

Time in seconds aggregated by mean for all specification DGPs.

4.2 Discussion

As observed, there are unexpected behaviors in the results when it comes to DML. One notable observation is that, in many cases, the estimator performs better when some confounders are omitted compared to when no confounders are omitted. This result is counterintuitive. One potential reason for this could be that there are simply too few observations for DML to achieve consistency, leading to these results. This hypothesis is quickly tested by increasing the number of observations to 30,000 for the Non-Linear DGP scenario with $\lambda = 0.5$ and $\gamma = 0.5$. The results show that with an increased number of observations, this behavior no longer appears, and the algorithm estimates most accurately when all confounders are included (see Appendix 16).

A second possible explanation could be that for this specific DGPs, DML encounters issues relative to other estimators, as it is the only method that incorporates the propensity score as a foundational element in its estimate. Although the X-Learner also has the option to use the propensity score, as discussed in the methodology section, and does so in this study, the propensity score may have less impact on the overall estimate of the X-Learner. Additionally, the X-Learner uses a different machine learning method, namely elastic net, to estimate the propensity score; this choice is due to it being the default in the package used. A quick simulation with DML alone shows that, indeed, switching the nuisance function (for propensity score) to random forest or elastic net results in more stable and reasonable estimates. In the version where elastic net is used, the results even outperform all other estimators in this study (see Appendix 17).

A third possible reason for these unexpected behaviors could be that the hyperparameters for the DML method and its underlying machine learning models used to estimate the nuisance function are not adequately tuned. The XGBoost method used in this study can be fine-tuned in various ways, and more nuanced parameter settings than the default could lead to improved results. This consideration applies to the other machine learning estimators in this paper as well, as they also rely on machine learning-based nuisance function estimations.

To translate these findings to empirical applications, if one suspects similar data structure conditions to those in this study's DGP, caution should be exercised when implementing DML in combination with the XGBoost classifier (without hyperparameter tuning) used here, particularly when the sample size is small. However, as the number of observations increases, DML can yield highly accurate results, even outperforming other estimators.

Other than the DML the rest of the estimator behaved as expected in becoming more accurate in its estimate when in increasing the number of observations. In addition, the bias increased as stronger and more confounders where omitted.

Another insight from the results is that Linear Regression performed best in the linear case. This aligns with the theory that, when dependencies in the data are linear, Ordinary Least Squares (OLS) Linear Regression (as used in this paper) is the optimal linear estimator, or in other words it is the Best-Linear-Unbiased-Estimator (BLUE).

We also observe that when many confounders are omitted, machine learning methods tend to converge to a similar bias. This is intuitive, as omitting a large number of confounders leaves only limited information, reducing the ability of the estimators to diverge significantly. Additionally, the complexity of the data structure to estimate decreases when only a few confounders remain.

Lastly, the results offer some more guidance for empirical research. However, it is important to note that applying these findings to real-world data is challenging, as the true DGP of real-world data is unknown and can only be approximated. What have seen is that even with only one strong confounder omitted the bias can increase significantly. So if the researcher is not very confident that the undrlying assumption of CIA holds and that there might be some omitted confounder variables he might think about alternitive settups with instrumental variables or RDD to name two examples. However, if the reasearcher commits to the idea that the CIA holds that there are no omitted confounders he can look at this results as an guidance. Keeping in mind that the reasearcher must also have an idea about the structure of his DGP, the recommendation would be going into XBCF or X-learner when the setting is similar to the non linear DGP in this paper. Especially with a low number of observations these two seem to to good. If the DGP is strongly suspected to be linear there Linear Regression method is the right call. But, as shown with a higher number of observations X-learner does also do not bad in the Linear DGP case, so if there is doubt about linear relationships X-learner is a valuable option.

5 Conclusion

We began by presenting related research, focusing on machine learning methods used in causal effect estimation. Next, the theoretical foundations of the study were laid out, followed by a description of the two Data Generating Processes (DGPs)—Linear and Non-Linear. The data generated from these DGPs was examined, and the underlying assumption of common support was tested. Subsequently, the method of introducing violations to the CIA by omitting confounders was explained. In this context, the approach of measuring the confounding level through the correlation between the error term and the remaining function was introduced. The machine learning methods used for estimating the ATE were briefly outlined, and the measures employed to assess the performance of the estimators were presented.

In the results and conclusions section, the DML estimator exhibited some unexpected behavior when certain variables were omitted, particularly when using XGBoost as the nuisance function for propensity score estimation and when the sample size was small. Possible reasons for this behavior include the reliance on the propensity score, the low number of observations, and the absence of hyperparameter tuning. Linear Regression performed the worst in the Non-Linear setting but excelled in the Linear setting, where it was closely followed by the X-Learner. In contrast, XBCF and X-Learner performed comparably well in the Non-Linear DGP, consistently outperforming the T-Learner.

Recommendations from this simulation study include exercising caution when implementing DML with XGBoost for propensity score estimation, especially with small sample sizes. Despite these challenges, DML remains a valuable method, as it occasionally outperforms other estimators.

Another key insight is that omitting variables can lead to significant bias, reinforcing the importance of ensuring the CIA holds, even when using advanced machine learning methods. For scenarios where the CIA is expected to hold and the data structure resembles the DGPs used in this study, Linear Regression is the recommended approach in Linear settings, while XBCF or X-Learner are preferable in Non-Linear settings.

Future research could explore the behavior of XGBoost as a propensity score estimation method within DML more profoundly, particularly with hyperparameter tuning, which was beyond the computational scope of this study. Additionally, extending the simulation to incorporate heterogeneous treatment effects could provide further valuable insights.

References

Athey, S., & Imbens, G. (2016). Recursive partitioning for heterogeneous causal effects. *Proceedings of the National Academy of Sciences of the United States of America*, *113*(27), 7353–7360. https://doi.org/10.1073/pnas.1510489113

Athey, S., & Imbens, G. W. (2019). Machine learning methods that economists should know about. *Annual Review of Economics*, *11*(1), 685–725. https://doi.org/10.1146/annurev-economics-080217-053433

Bach, P., Chernozhukov, V., Kurz, M. S., & Spindler, M. (2022). Doubleml - an object-oriented implementation of double machine learning in python. *Journal of Machine Learning Research*, *23*(53), 1–6.

Bail, C. A., Guay, B., Maloney, E., Combs, A., Hillygus, D. S., Merhout, F., Freelon, D., & Volfovsky, A. (2020). Assessing the russian internet research agency's impact on the political attitudes and behaviors of american twitter users in late 2017. *Proceedings of the National Academy of Sciences of the United States of America*, *117*(1), 243–250. https://doi.org/10.1073/pnas.1906420116

Becker, S. O., & Caliendo, M. (2007). Sensitivity analysis for average treatment effects. *The Stata Journal: Promoting communications on statistics and Stata*, *7*(1), 71–83. https://doi.org/10.1177/1536867X0700700104

Brand, J. E., Xu, J., Koch, B., & Geraldo, P. (2021). Uncovering sociological effect heterogeneity using tree-based machine learning. *Sociological methodology*, *51*(2), 189–223. https://doi.org/10.1177/0081175021993503

Brand, J. E., Zhou, X., & Xie, Y. (2023). Recent developments in causal inference and machine learning. *Annual review of sociology*, *49*, 81–110. https://doi.org/10.1146/annurev-soc-030420-015345

Bryan, C. J., Yeager, D. S., & O'Brien, J. M. (2019). Replicator degrees of freedom allow publication of misleading failures to replicate. *Proceedings of the National Academy of Sciences of the United States of America*, *116*(51), 25535–25545. https://doi.org/10.1073/pnas.1910951116

Burton, A., Altman, D. G., Royston, P., & Holder, R. L. (2006). The design of simulation studies in medical statistics. *Statistics in medicine*, *25*(24), 4279–4292. https://doi.org/10.1002/sim.2673

Caliendo, M., & Kopeinig, S. (2008). Some practical guidance for the implementation of propensity score matching. *Journal of Economic Surveys*, *22*(1), 31–72. https://doi.org/10.1111/j.1467-6419.2007.00527.x

Chernozhukov, V., Newey, W. K., & Singh, R. (2018). Automatic debiased machine learning of causal and structural effects. https://doi.org/10.48550/arXiv.1809.05224

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

Chipman, H. A., George, E. I., & McCulloch, R. E. (2010). Bart: Bayesian additive regression trees. *The Annals of Applied Statistics*, *4*(1). https://doi.org/10.1214/09-AOAS285

Cox, D. R. (1958). *Planning of experiments*. Wiley.

Crump, R. K., Hotz, V. J., Imbens, G. W., & Mitnik, O. A. (2009). Dealing with limited overlap in estimation of average treatment effects. *Biometrika*, *96*(1), 187–199. https://doi.org/10.1093/biomet/asn055

Davis, J. M., & Heller, S. B. (2017). Using causal forests to predict treatment heterogeneity: An application to summer jobs. *American Economic Review*, *107*(5), 546–550. https://doi.org/10.1257/aer.p20171000

Dorie, V., Hill, J., Shalit, U., Scott, M., & Cervone, D. (n.d.). Automated versus do-it-yourself methods for causal inference: Lessons learned from a data analysis competition. http://arxiv.org/pdf/1707.02641

Fisher, R. A. (1935). *The design of experiments*. Oliver & Boyd.

Garces, E., Thomas, D., & Currie, J. (2002). Longer-term effects of head start. *American Economic Review*, *92*(4), 999–1012. https://doi.org/10.1257/00028280260344560

Garrido, M. M., Kelley, A. S., Paris, J., Roza, K., Meier, D. E., Morrison, R. S., & Aldridge, M. D. (2014). Methods for constructing and assessing propensity scores. *Health services research*, *49*(5), 1701–1720. https://doi.org/10.1111/1475-6773.12182

Hahn, P. R., Murray, J. S., & Carvalho, C. M. (2020). Bayesian regression tree models for causal inference: Regularization, confounding, and heterogeneous effects (with discussion). *Bayesian Analysis*, *15*(3). https://doi.org/10.1214/19-BA1195

He, J., & Hahn, P. R. (2020). Stochastic tree ensembles for regularized nonlinear regression. *Journal of the American Statistical Association*, *118*(541), 551–570. https://doi.org/10.1080/01621459.2021.1942012

He, J., Yalov, S., & Hahn, P. R. (2019). Xbart: Accelerated bayesian additive regression trees. *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, *89*. http://arxiv.org/pdf/1810.02215

imbens, G. W. (2000). The role of the propensity score in estimating dose-response functions. *Biometrika*, *87*(3), 706–710.

Imbens, G. W. (2004). Nonparametric estimation of average treatment effects under exogeneity: A review. *Review of Economics and Statistics*, *86*(1), 4–29. https://doi.org/10.1162/003465304323023651

King, C. R., Escallier, K. E., Ju, Y.-E. S., Lin, N., Palanca, B. J., McKinnon, S. L., & Avidan, M. S. (2019). Obstructive sleep apnoea, positive airway pressure treatment and postoperative delirium: Protocol for a retrospective observational study. *BMJ open*, *9*(8), e026649. https://doi.org/10.1136/bmjopen-2018-026649

Krantsevich, N., Jingyu, H., & Richard, H. P. (2023). Stochastic tree ensembles for estimating heterogeneous effects. *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, *206*, 6120–6131.

Künzel, S. R., Sekhon, J. S., Bickel, P. J., & Yu, B. (2019). Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences of the United States of America*, *116*(10), 4156–4165. https://doi.org/10.1073/pnas.1804597116

Lechner, M. (2001). Identification and estimation of causal effects of multiple treatments under the conditional independence assumption. In F. Pfeiffer & M. Lechner (Eds.), *Econometric evaluation of active labour market policies*. Physica.

Lechner, M. (1999). Earnings and employment effects of continuous off-the-job training in east germany after unification. *Journal of Business & Economic Statistics*, *17*(1), 74–90.

Lechner, M. (2002). Program heterogeneity and propensity score matching: An application to the evaluation of active labor market policies. *Review of Economics and Statistics*, *84*(2), 205–220. https://doi.org/10.1162/003465302317411488

Lee, B. K., Lessler, J., & Stuart, E. A. (2010). Improving propensity score weighting using machine learning. *Statistics in medicine*, *29*(3), 337–346. https://doi.org/10.1002/sim.3782

McCaffrey, D. F., Ridgeway, G., & Morral, A. R. (2004). Propensity score estimation with boosted regression for evaluating causal effects in observational studies. *Psychological methods*, *9*(4), 403–425. https://doi.org/10.1037/1082-989X.9.4.403

McConnell, K. J., & Lindner, S. (2019). Estimating treatment effects with machine learning. *Health services research*, *54*(6), 1273–1282. https://doi.org/10.1111/1475-6773.13212

Miller, S. (2020). Causal forest estimation of heterogeneous and time-varying environmental policy effects. *Journal of Environmental Economics and Management*, *103*, 102337. https://doi.org/10.1016/j.jeem.2020.102337

Morris, T. P., White, I. R., & Crowther, M. J. (2019). Using simulation studies to evaluate statistical methods. *Statistics in medicine*, *38*(11), 2074–2102. https://doi.org/10.1002/sim.8086

Neyman, J. S. (1990). On the application of probability theory to agricultural experiments. essay on principles. section 9. *Statistical Science*, *5*(4), 465–472.

Nie, X., & Wager, S. (2021). Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, *108*(2), 299–319. http://arxiv.org/pdf/1712.04912

Paffenbarger, R. S., Hyde, R. T., Wing, A. L., & Hsieh, C. C. (1986). Physical activity, all-cause mortality, and longevity of college alumni. *The New England journal of medicine*, *314*(10), 605–613. https://doi.org/10.1056/NEJM198603063141003

Robins, J. M., & Rotnitzky, A. (1995). Semiparametric efficiency in multivariate regression models with missing data. *Journal of the American Statistical Association*, *90*(429), 122. https://doi.org/10.2307/2291135

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. https://doi.org/10.2307/2290910

Rosenbaum, P. R. (2002). *Observational studies* (2nd ed.). Springer.

Rosenbaum, P. R., & Rubin, D. B. (1983). The central role of the propensity score in observational studies for causal effects. *Biometrika*, *70*(1), 41. https://doi.org/10.2307/2335942

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. (1977). Assignment to treatment group on the basis of a covariate. *Journal of Educational Statistics*, *2*(1), 1. https://doi.org/10.2307/1164933

Rubin, D. B. (1980). Randomization analysis of experimental data: The fisher randomization test comment. *Journal of the American Statistical Association*, *75*(371), 591. https://doi.org/10.2307/2287653

Rubin, D. B. (2005). Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, *100*(469), 322–331.

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

Seabold, Skipper, & Perktold, J. (2010). *Statsmodels: Econometric and statistical modeling with python: 9th python in science conference.*

Stürmer, T., Rothman, K. J., Avorn, J., & Glynn, R. J. (2010). Treatment effects in the presence of unmeasured confounding: Dealing with observations in the tails of the propensity score distribution–a simulation study. *American journal of epidemiology*, *172*(7), 843–854. https://doi.org/10.1093/aje/kwq198

van der Laan, M. J., & Rose, S. (2011). *Targeted learning: Causal inference for observational and experimental data.* Springer.

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

Wager, S., & Athey, S. (2018). Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, *113*(523), 1228–1242. https://doi.org/10.1080/01621459.2017.1319839

Westreich, D., Lessler, J., & Funk, M. J. (2010). Propensity score estimation: Neural networks, support vector machines, decision trees (cart), and meta-classifiers as alternatives to logistic regression. *Journal of clinical epidemiology*, *63*(8), 826–833. https://doi.org/10.1016/j.jclinepi.2009.11.020

Yao, L., Chu, Z., Li, S., Li, Y., Gao, J., & Zhang, A. (2021). A survey on causal inference. *ACM Transactions on Knowledge Discovery from Data*, *15*(5), 1–46. https://doi.org/10.1145/3444944

Appendix

Appendix I: Extended main Result Tables

In this Appendix Section all simulation result are at display in a table form. Note that the scenarios name are now slightly different due to readability. For example since we have multiple scenarios for a strong confounder S (with different X_j omitted) they are displayed as "S_Xj". For an exact mapping of the scenarios with the confounders, please refer to Table 18

Table 9: Appendix I: Bias, Absolute Bias, and RMSE for Non-Linear DGP 1

	ScenarioX	No Omit			S_X1		
N Obs	Method	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE
2000	Linear Regression	1.304	1.304	1.308	1.329	1.329	1.332
	DML	-2.390	2.390	2.458	-2.091	2.091	2.164
	T-learner	0.672	0.672	0.676	0.679	0.679	0.683
	X-learner	0.459	0.459	0.464	0.484	0.484	0.489
	XBCF	0.309	0.309	0.316	0.355	0.355	0.363
8000	Linear Regression	1.301	1.301	1.302	1.332	1.332	1.332
	DML	-0.438	0.438	0.453	-0.324	0.324	0.341
	T-learner	0.432	0.432	0.433	0.460	0.460	0.461
	X-learner	0.267	0.267	0.269	0.314	0.314	0.316
	XBCF	0.169	0.169	0.172	0.234	0.234	0.236
specification: lambda = 0.5 gamma = 0.5							
2000	Linear Regression	2.281	2.281	2.282	2.326	2.326	2.328
	DML	-2.806	2.806	2.873	-2.509	2.509	2.580
	T-learner	1.363	1.363	1.365	1.388	1.388	1.390
	X-learner	0.946	0.946	0.949	1.001	1.001	1.004
	XBCF	0.682	0.682	0.687	0.782	0.782	0.787
8000	Linear Regression	2.284	2.284	2.285	2.329	2.329	2.329
	DML	-0.609	0.609	0.631	-0.389	0.389	0.418
	T-learner	0.936	0.936	0.937	1.000	1.000	1.000
	X-learner	0.572	0.572	0.573	0.678	0.678	0.679
	XBCF	0.404	0.404	0.407	0.542	0.542	0.544
specification: lambda = 1.2 gamma = 0.5							
2000	Linear Regression	0.058	0.065	0.077	0.063	0.068	0.080
	DML	-0.081	0.212	0.266	-0.078	0.206	0.261
	T-learner	0.027	0.046	0.056	0.032	0.047	0.057
	X-learner	0.019	0.043	0.054	0.025	0.045	0.055
	XBCF	0.022	0.043	0.054	0.027	0.044	0.055
8000	Linear Regression	0.062	0.062	0.065	0.061	0.061	0.065
	DML	-0.009	0.037	0.047	-0.010	0.039	0.050
	T-learner	0.018	0.022	0.028	0.017	0.023	0.029
	X-learner	0.013	0.021	0.026	0.013	0.021	0.027
	XBCF	0.020	0.024	0.029	0.018	0.024	0.030
specification: lambda = 0.1 gamma = 0.1							

Table 10: Appendix I: Bias, Absolute Bias, and RMSE for Non-Linear DGP 2

	ScenarioX	S_X2			S_X3			W_X17			W_X18		
		Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE
N Obs	Method												
2000	Linear Regression	1.412	1.412	1.415	1.327	1.327	1.330	1.300	1.300	1.303	1.298	1.298	1.301
	DML	-1.765	1.766	1.861	-0.466	0.599	0.744	-2.204	2.204	2.274	-2.187	2.187	2.262
	T-learner	0.747	0.747	0.750	1.164	1.164	1.167	0.665	0.665	0.669	0.669	0.669	0.672
	X-learner	0.570	0.570	0.574	1.050	1.050	1.054	0.459	0.459	0.464	0.460	0.460	0.464
	XBCF	0.470	0.470	0.476	1.079	1.079	1.083	0.329	0.329	0.336	0.326	0.326	0.333
8000	Linear Regression	1.405	1.405	1.406	1.332	1.332	1.333	1.304	1.304	1.305	1.301	1.301	1.301
	DML	-0.179	0.181	0.205	0.617	0.617	0.627	-0.371	0.371	0.386	-0.375	0.375	0.391
	T-learner	0.538	0.538	0.539	1.032	1.032	1.032	0.438	0.438	0.439	0.439	0.439	0.440
	X-learner	0.403	0.403	0.404	0.956	0.956	0.957	0.283	0.283	0.285	0.283	0.283	0.285
	XBCF	0.345	0.345	0.347	1.006	1.006	1.007	0.199	0.199	0.202	0.200	0.200	0.203
specification: lambda = 0.5 gamma = 0.5													
2000	Linear Regression	2.436	2.436	2.437	2.328	2.328	2.330	2.283	2.283	2.285	2.285	2.285	2.286
	DML	-2.039	2.039	2.132	0.002	0.474	0.607	-2.618	2.618	2.698	-2.654	2.654	2.718
	T-learner	1.503	1.503	1.505	2.081	2.081	2.083	1.366	1.366	1.368	1.364	1.364	1.366
	X-learner	1.165	1.165	1.168	1.902	1.902	1.904	0.960	0.960	0.963	0.957	0.957	0.960
	XBCF	1.011	1.011	1.015	1.975	1.975	1.978	0.729	0.729	0.733	0.729	0.729	0.733
8000	Linear Regression	2.433	2.433	2.433	2.322	2.322	2.322	2.288	2.288	2.289	2.289	2.289	2.289
	DML	-0.073	0.137	0.171	1.267	1.267	1.275	-0.508	0.508	0.533	-0.497	0.497	0.521
	T-learner	1.139	1.139	1.140	1.865	1.865	1.865	0.952	0.952	0.953	0.953	0.953	0.953
	X-learner	0.853	0.853	0.854	1.736	1.736	1.736	0.608	0.608	0.609	0.608	0.608	0.609
	XBCF	0.766	0.766	0.767	1.830	1.830	1.830	0.469	0.469	0.472	0.466	0.466	0.468
specification: lambda = 1.2 gamma = 0.5													
2000	Linear Regression	0.066	0.069	0.082	0.058	0.064	0.075	0.062	0.067	0.079	0.057	0.064	0.076
	DML	-0.069	0.214	0.268	-0.005	0.204	0.257	-0.088	0.202	0.255	-0.089	0.212	0.267
	T-learner	0.033	0.047	0.059	0.049	0.057	0.069	0.031	0.047	0.059	0.025	0.044	0.055
	X-learner	0.027	0.045	0.057	0.044	0.054	0.066	0.024	0.045	0.056	0.018	0.042	0.053
	XBCF	0.029	0.046	0.057	0.049	0.057	0.068	0.026	0.045	0.055	0.021	0.042	0.053
8000	Linear Regression	0.065	0.065	0.070	0.060	0.060	0.065	0.059	0.059	0.064	0.059	0.059	0.063
	DML	0.004	0.038	0.047	0.036	0.050	0.062	-0.016	0.040	0.050	-0.010	0.040	0.048
	T-learner	0.022	0.027	0.033	0.046	0.047	0.053	0.016	0.024	0.030	0.015	0.021	0.026
	X-learner	0.018	0.025	0.030	0.044	0.045	0.051	0.011	0.022	0.027	0.010	0.019	0.024
	XBCF	0.023	0.027	0.033	0.049	0.049	0.055	0.018	0.025	0.031	0.017	0.022	0.028
specification: lambda = 0.1 gamma = 0.1													

Table 11: Appendix I: Bias, Absolute Bias, and RMSE for Non-Linear DGP 3

	ScenarioX	S W			S S			W W			10W		
		Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE
N Obs	Method												
2000	Linear Regression	1.413	1.413	1.416	1.441	1.441	1.444	1.299	1.299	1.302	1.355	1.355	1.359
	DML	-1.679	1.679	1.767	-1.615	1.615	1.702	-2.106	2.106	2.175	-0.743	0.765	0.907
	T-learner	0.747	0.747	0.750	0.757	0.757	0.761	0.662	0.662	0.666	0.674	0.674	0.679
	X-learner	0.575	0.575	0.579	0.594	0.594	0.599	0.463	0.463	0.468	0.550	0.550	0.556
	XBCF	0.489	0.489	0.495	0.506	0.506	0.512	0.348	0.348	0.355	0.535	0.535	0.540
8000	Linear Regression	1.409	1.409	1.410	1.432	1.432	1.433	1.301	1.301	1.302	1.360	1.360	1.361
	DML	-0.105	0.122	0.149	-0.068	0.103	0.128	-0.309	0.309	0.329	0.204	0.206	0.227
	T-learner	0.545	0.545	0.546	0.567	0.567	0.568	0.441	0.441	0.442	0.540	0.540	0.541
	X-learner	0.417	0.417	0.418	0.450	0.450	0.452	0.294	0.294	0.295	0.462	0.462	0.464
	XBCF	0.368	0.368	0.370	0.402	0.402	0.404	0.225	0.225	0.228	0.468	0.468	0.469
specification: lambda = 0.5 gamma = 0.5													
2000	Linear Regression	2.433	2.433	2.435	2.477	2.477	2.479	2.289	2.289	2.291	2.369	2.369	2.370
	DML	-1.931	1.931	2.025	-1.761	1.761	1.866	-2.540	2.540	2.614	-0.682	0.751	0.907
	T-learner	1.497	1.497	1.499	1.533	1.533	1.536	1.363	1.363	1.365	1.420	1.420	1.423
	X-learner	1.170	1.170	1.173	1.222	1.222	1.225	0.968	0.968	0.971	1.172	1.172	1.175
	XBCF	1.032	1.032	1.036	1.083	1.083	1.087	0.773	0.773	0.777	1.143	1.143	1.146
8000	Linear Regression	2.433	2.433	2.434	2.473	2.473	2.474	2.280	2.280	2.280	2.368	2.368	2.368
	DML	0.033	0.120	0.150	0.134	0.167	0.201	-0.360	0.361	0.391	0.619	0.619	0.635
	T-learner	1.156	1.156	1.157	1.201	1.201	1.202	0.965	0.965	0.966	1.189	1.189	1.190
	X-learner	0.886	0.886	0.887	0.952	0.952	0.953	0.640	0.640	0.641	1.009	1.009	1.009
	XBCF	0.821	0.821	0.823	0.875	0.875	0.876	0.524	0.524	0.526	1.022	1.022	1.023
specification: lambda = 1.2 gamma = 0.5													
2000	Linear Regression	0.067	0.070	0.081	0.067	0.071	0.083	0.061	0.066	0.078	0.065	0.070	0.083
	DML	-0.083	0.210	0.262	-0.056	0.205	0.261	-0.086	0.209	0.268	-0.027	0.194	0.250
	T-learner	0.034	0.047	0.057	0.034	0.048	0.059	0.030	0.045	0.056	0.032	0.049	0.061
	X-learner	0.028	0.044	0.054	0.028	0.046	0.056	0.023	0.043	0.054	0.028	0.048	0.060
	XBCF	0.029	0.044	0.055	0.031	0.046	0.056	0.025	0.042	0.053	0.030	0.048	0.059
8000	Linear Regression	0.065	0.065	0.069	0.066	0.066	0.070	0.059	0.059	0.065	0.063	0.063	0.068
	DML	-0.001	0.038	0.048	0.000	0.040	0.050	-0.008	0.041	0.052	0.006	0.036	0.045
	T-learner	0.022	0.026	0.033	0.022	0.028	0.034	0.016	0.024	0.030	0.023	0.028	0.034
	X-learner	0.018	0.025	0.031	0.019	0.026	0.033	0.012	0.023	0.029	0.021	0.026	0.032
	XBCF	0.024	0.027	0.034	0.024	0.028	0.035	0.018	0.025	0.031	0.026	0.030	0.035
specification: lambda = 0.1 gamma = 0.1													

Table 12: Appendix I: Bias, Absolute Bias, and RMSE for Non-Linear DGP 4

	ScenarioX	5S 5W			10S			All but 2S		
		Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE
N Obs	Method									
2000	Linear Regression	1.519	1.519	1.521	1.591	1.591	1.594	1.571	1.571	1.574
	DML	0.955	0.979	1.115	0.922	0.948	1.090	0.787	0.791	0.851
	T-learner	1.377	1.377	1.381	1.418	1.418	1.421	0.898	0.898	0.903
	X-learner	1.348	1.348	1.351	1.394	1.394	1.398	0.875	0.875	0.880
	XBCF	1.378	1.378	1.381	1.424	1.424	1.427	0.894	0.894	0.898
8000	Linear Regression	1.519	1.519	1.520	1.587	1.587	1.588	1.570	1.570	1.571
	DML	1.236	1.236	1.241	1.286	1.286	1.290	0.838	0.838	0.841
	T-learner	1.336	1.336	1.337	1.377	1.377	1.378	0.876	0.876	0.877
	X-learner	1.324	1.324	1.325	1.367	1.367	1.368	0.863	0.863	0.864
	XBCF	1.350	1.350	1.351	1.393	1.393	1.394	0.871	0.871	0.872
specification: lambda = 0.5 gamma = 0.5										
2000	Linear Regression	2.586	2.586	2.587	2.677	2.677	2.678	2.649	2.649	2.650
	DML	1.750	1.750	1.834	1.892	1.892	1.969	1.638	1.638	1.685
	T-learner	2.392	2.392	2.394	2.445	2.445	2.447	1.799	1.799	1.802
	X-learner	2.349	2.349	2.351	2.409	2.409	2.411	1.736	1.736	1.739
	XBCF	2.408	2.408	2.409	2.472	2.472	2.474	1.734	1.734	1.737
8000	Linear Regression	2.587	2.587	2.587	2.684	2.684	2.684	2.658	2.658	2.658
	DML	2.213	2.213	2.216	2.277	2.277	2.280	1.725	1.725	1.728
	T-learner	2.343	2.343	2.343	2.401	2.401	2.401	1.776	1.776	1.777
	X-learner	2.323	2.323	2.323	2.383	2.383	2.384	1.730	1.730	1.731
	XBCF	2.360	2.360	2.361	2.425	2.425	2.426	1.713	1.713	1.714
specification: lambda = 1.2 gamma = 0.5										
2000	Linear Regression	0.073	0.076	0.089	0.073	0.076	0.089	0.073	0.076	0.088
	DML	0.044	0.205	0.259	0.031	0.207	0.258	0.028	0.123	0.155
	T-learner	0.063	0.068	0.081	0.064	0.069	0.084	0.039	0.052	0.066
	X-learner	0.062	0.068	0.081	0.063	0.069	0.084	0.038	0.054	0.067
	XBCF	0.062	0.069	0.081	0.062	0.068	0.081	0.038	0.049	0.062
8000	Linear Regression	0.071	0.071	0.075	0.076	0.076	0.079	0.072	0.072	0.076
	DML	0.053	0.061	0.073	0.058	0.064	0.076	0.034	0.040	0.049
	T-learner	0.062	0.062	0.067	0.065	0.065	0.069	0.036	0.037	0.043
	X-learner	0.061	0.061	0.067	0.064	0.064	0.069	0.036	0.037	0.043
	XBCF	0.063	0.063	0.067	0.066	0.066	0.070	0.038	0.039	0.044
specification: lambda = 0.1 gamma = 0.1										

Table 13: Appendix I: Bias, Absolute Bias, and RMSE for Linear DGP 1

	ScenarioX	No Omit			S_X1			S_X2			W_X15		
		Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE
N Obs	Method												
2000	Linear Regression	-0.000	0.036	0.044	0.070	0.074	0.086	0.062	0.069	0.082	0.010	0.038	0.047
	DML	-0.968	0.969	1.030	-0.687	0.695	0.778	-0.673	0.683	0.767	-0.924	0.924	0.982
	T-learner	0.141	0.142	0.151	0.181	0.181	0.189	0.174	0.174	0.184	0.146	0.146	0.155
	X-learner	0.026	0.046	0.058	0.091	0.094	0.106	0.084	0.089	0.103	0.034	0.049	0.062
	XBCF	0.051	0.058	0.071	0.115	0.116	0.127	0.107	0.109	0.122	0.057	0.065	0.078
8000	Linear Regression	0.002	0.018	0.023	0.072	0.072	0.076	0.065	0.065	0.070	0.003	0.019	0.023
	DML	-0.180	0.180	0.191	-0.071	0.077	0.092	-0.074	0.082	0.099	-0.162	0.162	0.173
	T-learner	0.079	0.079	0.083	0.128	0.128	0.131	0.123	0.123	0.126	0.075	0.075	0.079
	X-learner	0.011	0.022	0.027	0.077	0.077	0.081	0.071	0.071	0.076	0.011	0.022	0.028
	XBCF	0.032	0.034	0.040	0.098	0.098	0.101	0.093	0.093	0.097	0.032	0.034	0.041
specification: lambda = 0.1 gamma = 0.5													
2000	Linear Regression	0.002	0.048	0.060	0.331	0.331	0.337	0.302	0.302	0.308	0.027	0.050	0.062
	DML	-1.906	1.906	1.953	-1.261	1.261	1.338	-1.259	1.259	1.347	-1.887	1.887	1.942
	T-learner	0.601	0.601	0.604	0.743	0.743	0.745	0.728	0.728	0.730	0.597	0.597	0.600
	X-learner	0.157	0.157	0.170	0.427	0.427	0.433	0.400	0.400	0.405	0.167	0.168	0.179
	XBCF	0.272	0.272	0.280	0.548	0.548	0.551	0.522	0.522	0.525	0.282	0.282	0.288
8000	Linear Regression	0.005	0.020	0.026	0.339	0.339	0.340	0.306	0.306	0.308	0.023	0.030	0.037
	DML	-0.590	0.590	0.607	-0.059	0.111	0.138	-0.078	0.126	0.159	-0.547	0.547	0.564
	T-learner	0.350	0.350	0.351	0.573	0.573	0.573	0.550	0.550	0.551	0.354	0.354	0.356
	X-learner	0.043	0.045	0.053	0.370	0.370	0.371	0.340	0.340	0.342	0.060	0.061	0.069
	XBCF	0.162	0.162	0.165	0.471	0.471	0.471	0.438	0.438	0.439	0.172	0.172	0.174
specification: lambda = 0.5 gamma = 0.5													
2000	Linear Regression	0.002	0.037	0.046	0.011	0.039	0.048	0.012	0.037	0.046	0.000	0.034	0.043
	DML	-0.166	0.247	0.305	-0.115	0.216	0.275	-0.117	0.235	0.294	-0.128	0.241	0.302
	T-learner	0.023	0.043	0.054	0.027	0.044	0.056	0.028	0.043	0.054	0.019	0.039	0.049
	X-learner	0.007	0.041	0.051	0.015	0.041	0.052	0.015	0.041	0.051	0.005	0.038	0.047
	XBCF	0.030	0.045	0.056	0.035	0.049	0.060	0.036	0.047	0.057	0.027	0.042	0.053
8000	Linear Regression	-0.002	0.018	0.022	0.011	0.020	0.025	0.011	0.019	0.024	0.002	0.018	0.024
	DML	-0.023	0.048	0.060	-0.010	0.042	0.053	-0.007	0.044	0.054	-0.016	0.043	0.055
	T-learner	0.007	0.020	0.024	0.018	0.024	0.030	0.018	0.024	0.029	0.011	0.021	0.027
	X-learner	-0.000	0.019	0.024	0.012	0.022	0.028	0.012	0.021	0.026	0.003	0.019	0.026
	XBCF	0.017	0.023	0.028	0.028	0.031	0.036	0.028	0.030	0.036	0.020	0.026	0.032
specification: lambda = 0.1 gamma = 0.1													

Table 14: Appendix I: Bias, Absolute Bias, and RMSE for Linear DGP 2

	ScenarioX	W_X16			S W			S S			W W		
		Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE
N Obs	Method												
2000	Linear Regression	0.004	0.036	0.046	0.076	0.080	0.092	0.136	0.136	0.146	0.011	0.037	0.047
	DML	-0.936	0.936	1.000	-0.652	0.656	0.736	-0.423	0.462	0.549	-0.864	0.864	0.928
	T-learner	0.141	0.141	0.150	0.183	0.183	0.191	0.221	0.221	0.229	0.141	0.141	0.150
	X-learner	0.029	0.048	0.059	0.094	0.096	0.110	0.152	0.152	0.163	0.032	0.049	0.061
	XBCF	0.052	0.060	0.072	0.120	0.122	0.133	0.175	0.175	0.184	0.058	0.065	0.077
8000	Linear Regression	0.002	0.018	0.023	0.077	0.077	0.081	0.139	0.139	0.141	0.011	0.020	0.026
	DML	-0.176	0.176	0.187	-0.059	0.070	0.087	0.038	0.061	0.076	-0.160	0.160	0.171
	T-learner	0.075	0.075	0.079	0.132	0.132	0.135	0.182	0.182	0.184	0.081	0.081	0.085
	X-learner	0.009	0.022	0.027	0.082	0.082	0.087	0.144	0.144	0.147	0.017	0.025	0.031
	XBCF	0.031	0.033	0.040	0.103	0.103	0.107	0.164	0.164	0.167	0.037	0.038	0.045
specification: lambda = 0.1 gamma = 0.5													
2000	Linear Regression	0.022	0.051	0.064	0.350	0.350	0.355	0.596	0.596	0.599	0.040	0.058	0.071
	DML	-1.856	1.856	1.907	-1.163	1.166	1.246	-0.643	0.675	0.791	-1.768	1.768	1.825
	T-learner	0.598	0.598	0.601	0.745	0.745	0.747	0.885	0.885	0.886	0.588	0.588	0.591
	X-learner	0.166	0.166	0.179	0.437	0.437	0.442	0.659	0.659	0.663	0.170	0.170	0.183
	XBCF	0.278	0.278	0.286	0.555	0.555	0.558	0.770	0.770	0.773	0.284	0.284	0.291
8000	Linear Regression	0.019	0.028	0.034	0.350	0.350	0.351	0.595	0.595	0.595	0.042	0.043	0.051
	DML	-0.552	0.552	0.567	-0.017	0.107	0.137	0.320	0.320	0.343	-0.499	0.499	0.521
	T-learner	0.352	0.352	0.353	0.575	0.575	0.576	0.756	0.756	0.757	0.357	0.357	0.359
	X-learner	0.056	0.057	0.065	0.382	0.382	0.383	0.621	0.621	0.622	0.076	0.076	0.084
	XBCF	0.168	0.168	0.171	0.476	0.476	0.477	0.704	0.704	0.704	0.184	0.184	0.187
specification: lambda = 0.5 gamma = 0.5													
2000	Linear Regression	-0.002	0.035	0.045	0.015	0.037	0.047	0.031	0.045	0.056	-0.001	0.038	0.047
	DML	-0.150	0.255	0.317	-0.115	0.228	0.285	-0.064	0.213	0.271	-0.122	0.238	0.301
	T-learner	0.016	0.039	0.049	0.030	0.045	0.056	0.042	0.053	0.064	0.016	0.041	0.052
	X-learner	0.001	0.039	0.048	0.018	0.042	0.052	0.032	0.047	0.058	0.002	0.040	0.050
	XBCF	0.025	0.041	0.052	0.038	0.049	0.060	0.051	0.057	0.069	0.026	0.045	0.055
8000	Linear Regression	0.002	0.019	0.023	0.016	0.022	0.027	0.029	0.031	0.036	0.002	0.019	0.025
	DML	-0.020	0.046	0.059	-0.004	0.043	0.054	0.018	0.042	0.051	-0.016	0.046	0.057
	T-learner	0.010	0.021	0.026	0.022	0.026	0.032	0.033	0.035	0.040	0.011	0.022	0.028
	X-learner	0.003	0.020	0.024	0.016	0.023	0.029	0.029	0.032	0.037	0.004	0.021	0.027
	XBCF	0.020	0.025	0.031	0.032	0.034	0.039	0.043	0.043	0.048	0.020	0.026	0.032
specification: lambda = 0.1 gamma = 0.1													

Table 15: Appendix I: Bias, Absolute Bias, and RMSE for Linear DGP 3

	ScenarioX	10W			5S 5W			10S			All but 2S		
		Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE	Bias	Abs Bias	RMSE
N Obs	Method												
2000	Linear Regression	0.077	0.080	0.092	0.296	0.296	0.302	0.424	0.424	0.429	0.352	0.352	0.357
	DML	-0.439	0.462	0.541	0.139	0.286	0.369	0.400	0.444	0.534	0.331	0.343	0.399
	T-learner	0.148	0.148	0.158	0.321	0.321	0.327	0.431	0.431	0.438	0.353	0.353	0.360
	X-learner	0.086	0.089	0.102	0.299	0.299	0.307	0.427	0.427	0.434	0.351	0.351	0.359
	XBCF	0.105	0.106	0.119	0.312	0.312	0.318	0.428	0.428	0.434	0.352	0.352	0.358
8000	Linear Regression	0.077	0.077	0.081	0.296	0.296	0.298	0.425	0.425	0.426	0.349	0.349	0.350
	DML	-0.007	0.046	0.059	0.269	0.269	0.276	0.421	0.421	0.427	0.347	0.347	0.351
	T-learner	0.113	0.113	0.116	0.308	0.308	0.309	0.427	0.427	0.428	0.349	0.349	0.350
	X-learner	0.081	0.081	0.085	0.298	0.298	0.299	0.425	0.425	0.426	0.348	0.348	0.350
	XBCF	0.094	0.094	0.098	0.308	0.308	0.310	0.429	0.429	0.431	0.351	0.351	0.352
specification: lambda = 0.1 gamma = 0.5													
2000	Linear Regression	0.360	0.360	0.365	1.108	1.108	1.110	1.429	1.429	1.431	1.248	1.248	1.250
	DML	-0.769	0.786	0.888	0.754	0.767	0.873	1.356	1.356	1.405	1.226	1.226	1.263
	T-learner	0.648	0.648	0.651	1.181	1.181	1.183	1.444	1.444	1.445	1.261	1.261	1.263
	X-learner	0.409	0.409	0.415	1.124	1.124	1.126	1.434	1.434	1.436	1.256	1.256	1.258
	XBCF	0.501	0.501	0.505	1.180	1.180	1.182	1.458	1.458	1.459	1.266	1.266	1.268
8000	Linear Regression	0.359	0.359	0.361	1.109	1.109	1.110	1.432	1.432	1.432	1.248	1.248	1.248
	DML	0.142	0.154	0.185	1.054	1.054	1.059	1.413	1.413	1.415	1.260	1.260	1.262
	T-learner	0.520	0.520	0.521	1.149	1.149	1.149	1.438	1.438	1.439	1.257	1.257	1.258
	X-learner	0.388	0.388	0.389	1.121	1.121	1.121	1.433	1.433	1.434	1.256	1.256	1.256
	XBCF	0.441	0.441	0.442	1.152	1.152	1.152	1.448	1.448	1.448	1.258	1.258	1.258
specification: lambda = 0.5 gamma = 0.5													
2000	Linear Regression	0.016	0.038	0.048	0.058	0.063	0.075	0.088	0.089	0.100	0.070	0.073	0.084
	DML	-0.066	0.214	0.274	0.031	0.203	0.255	0.087	0.192	0.241	0.071	0.141	0.178
	T-learner	0.024	0.042	0.053	0.061	0.066	0.079	0.088	0.090	0.102	0.070	0.074	0.085
	X-learner	0.016	0.041	0.051	0.059	0.065	0.079	0.087	0.088	0.101	0.070	0.074	0.087
	XBCF	0.031	0.045	0.057	0.065	0.069	0.081	0.088	0.089	0.100	0.068	0.072	0.082
8000	Linear Regression	0.016	0.024	0.029	0.061	0.061	0.066	0.086	0.086	0.089	0.069	0.069	0.072
	DML	0.006	0.039	0.051	0.054	0.060	0.072	0.086	0.088	0.099	0.069	0.070	0.080
	T-learner	0.020	0.027	0.032	0.062	0.062	0.067	0.086	0.086	0.089	0.068	0.068	0.072
	X-learner	0.016	0.026	0.031	0.061	0.061	0.066	0.085	0.085	0.089	0.068	0.068	0.072
	XBCF	0.027	0.031	0.037	0.067	0.067	0.072	0.088	0.088	0.091	0.070	0.070	0.073
specification: lambda = 0.1 gamma = 0.1													

Table 16: Appendix I: DML Results for 30000 Observations

ScenarioX	Bias	Abs Bias	RMSE
No Omit	0.0001	0.0182	0.0227
S_X1	0.0819	0.0819	0.0869
S_X2	0.1641	0.1641	0.1655
S_X3	0.8245	0.8245	0.8253
W_X17	0.0242	0.0301	0.0363
W_X18	0.0264	0.0287	0.0368
S W	0.2020	0.2020	0.2041
S S	0.2577	0.2577	0.2591
W W	0.0620	0.0627	0.0688
10W	0.3551	0.3551	0.3563
5S 5W	1.2973	1.2973	1.2976
10S	1.3385	1.3385	1.3389
All but 2S	0.8611	0.8611	0.8616

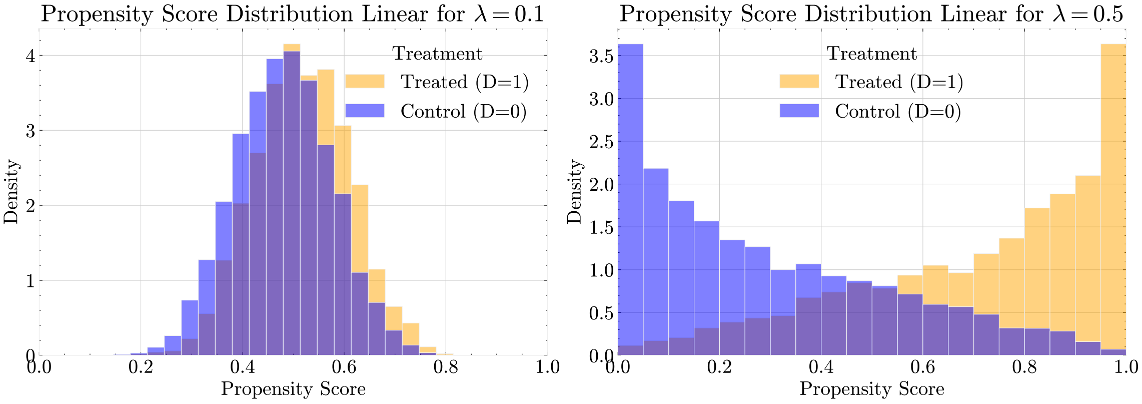
S and W correspond to Strong and Weak. Prop. Function stands for the underlying ML method used to estimate the nuisance function of propensity score. The DGP specifications are: Non-Linear DGP, lambda = 0.5, gamma = 0.5, N Obs = 30'000.

Table 17: Appendix I: DML Results for different Propensity

ScenarioX	Prop. Function	Abs Bias	Bias	RMSE
No Omit	ElasticNet	0.225361	0.225361	0.227866
	RandomForestClassifier	0.358988	0.358988	0.360468
S_X1	ElasticNet	0.256522	0.256522	0.258532
	RandomForestClassifier	0.389590	0.389590	0.391343
S_X2	ElasticNet	0.338617	0.338617	0.340358
	RandomForestClassifier	0.472090	0.472090	0.473533
S_X3	ElasticNet	0.893702	0.893702	0.894873
	RandomForestClassifier	0.985740	0.985740	0.986895
W_X17	ElasticNet	0.236302	0.236302	0.238646
	RandomForestClassifier	0.367690	0.367690	0.369201
W_X18	ElasticNet	0.233168	0.233168	0.235316
	RandomForestClassifier	0.366340	0.366340	0.368038
S W	ElasticNet	0.346711	0.346711	0.348444
	RandomForestClassifier	0.482348	0.482348	0.483789
S S	ElasticNet	0.373900	0.373900	0.375547
	RandomForestClassifier	0.503520	0.503520	0.504937
W W	ElasticNet	0.241306	0.241306	0.243293
	RandomForestClassifier	0.380068	0.380068	0.381624
10W	ElasticNet	0.356556	0.356556	0.358689
	RandomForestClassifier	0.498933	0.498933	0.500553
5S 5W	ElasticNet	1.264461	1.264461	1.265634
	RandomForestClassifier	1.319079	1.319079	1.320172
10S	ElasticNet	1.310638	1.310638	1.311915
	RandomForestClassifier	1.361782	1.361782	1.362924
All but 2S	ElasticNet	0.744264	0.744264	0.745855
	RandomForestClassifier	0.827626	0.827626	0.835886

S and W correspond to Strong and Weak. Prop. Function stands for the underlying ML method used to estimate the nuisance function of propensity score. The DGP specifications are: Non-Linear DGP, lambda = 0.5, gamma = 0.5, N Obs = 8000.

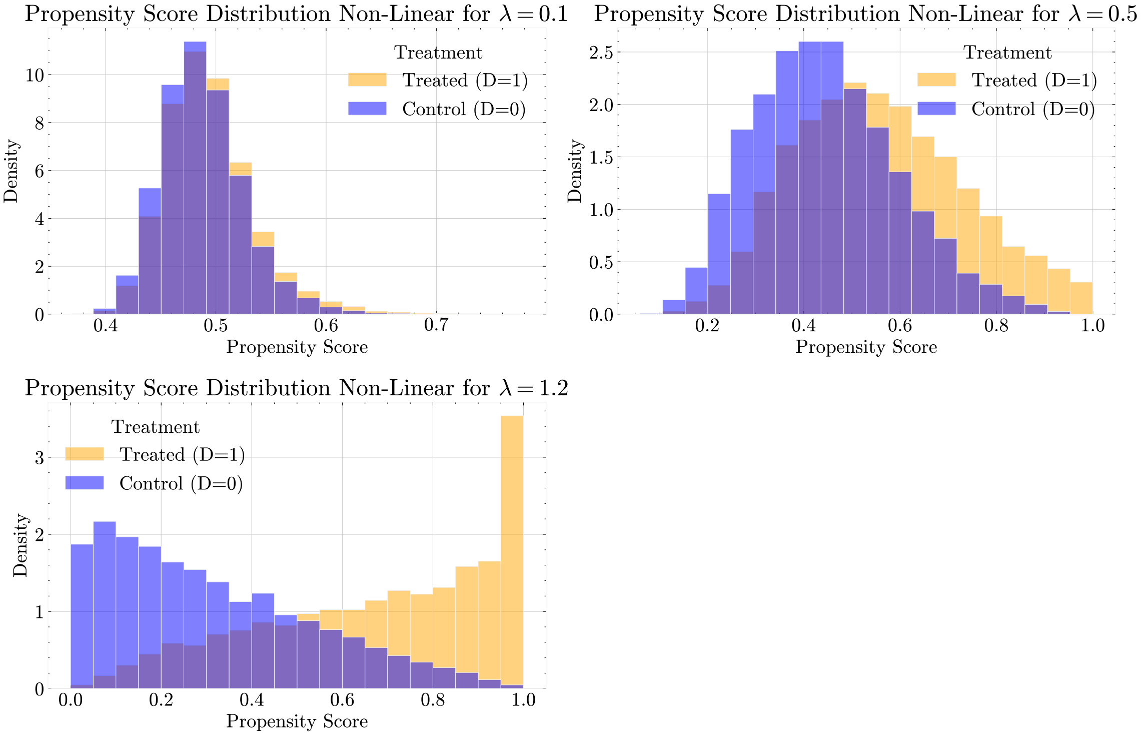
Appendix II: Propensity Scores



Propensity Score Distribution Linear for $\lambda = 0.1$

Propensity Score Distribution Linear for $\lambda = 0.5$

Figure 4: Appendix II: Propensity Score Linear DGP
Y-axis is the density. This propensity score is taken directly from the DGP with an N observation = 10'000.



Propensity Score Distribution Non-Linear for $\lambda = 0.1$

Propensity Score Distribution Non-Linear for $\lambda = 0.5$

Propensity Score Distribution Non-Linear for $\lambda = 1.2$

Figure 5: Appendix II: Propensity Score Non-Linear DGP
Y-axis is the density. This propensity score is taken directly from the DGP with an N observation = 10'000.

Appendix III: Extended Correlation Plot

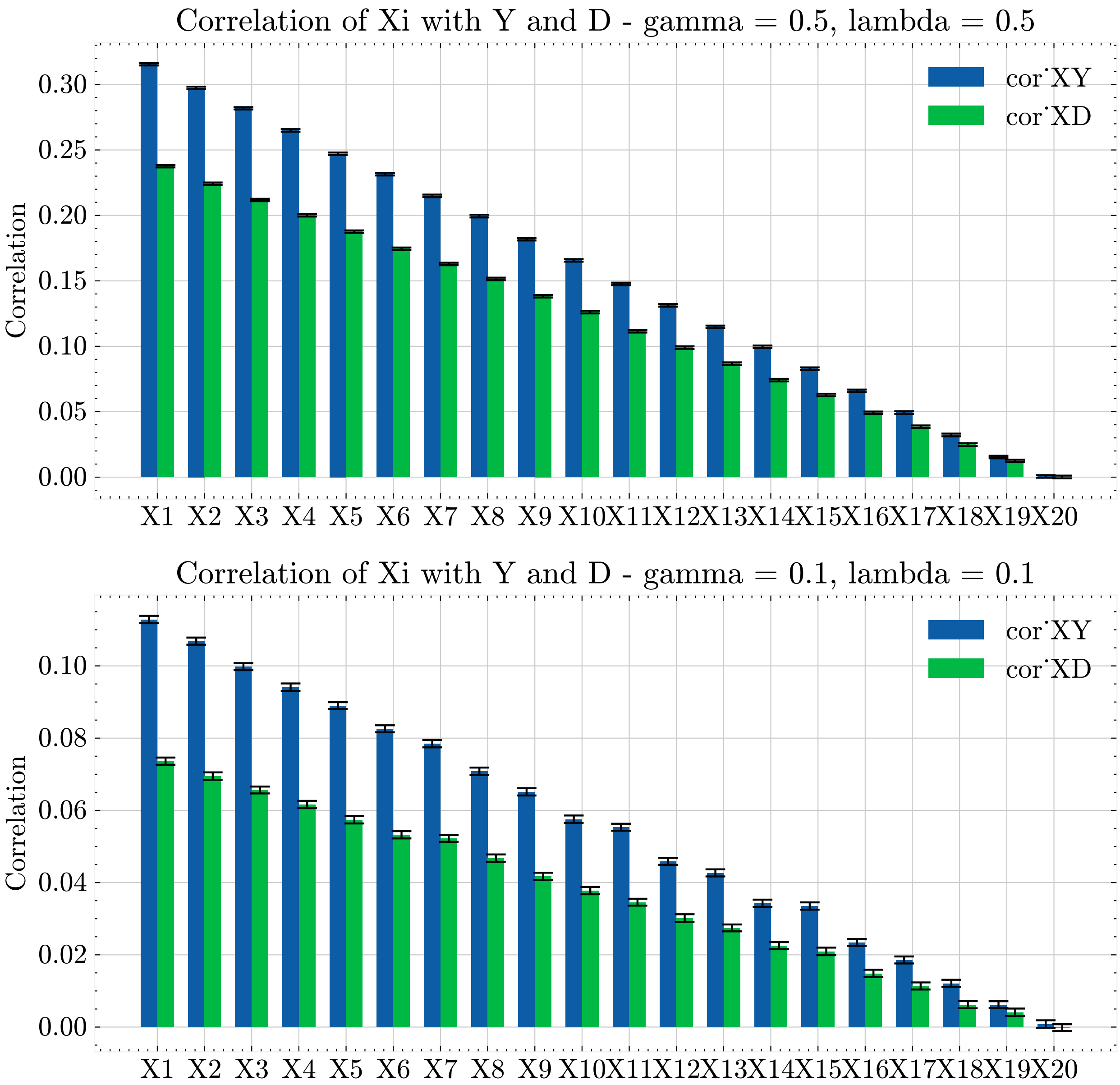
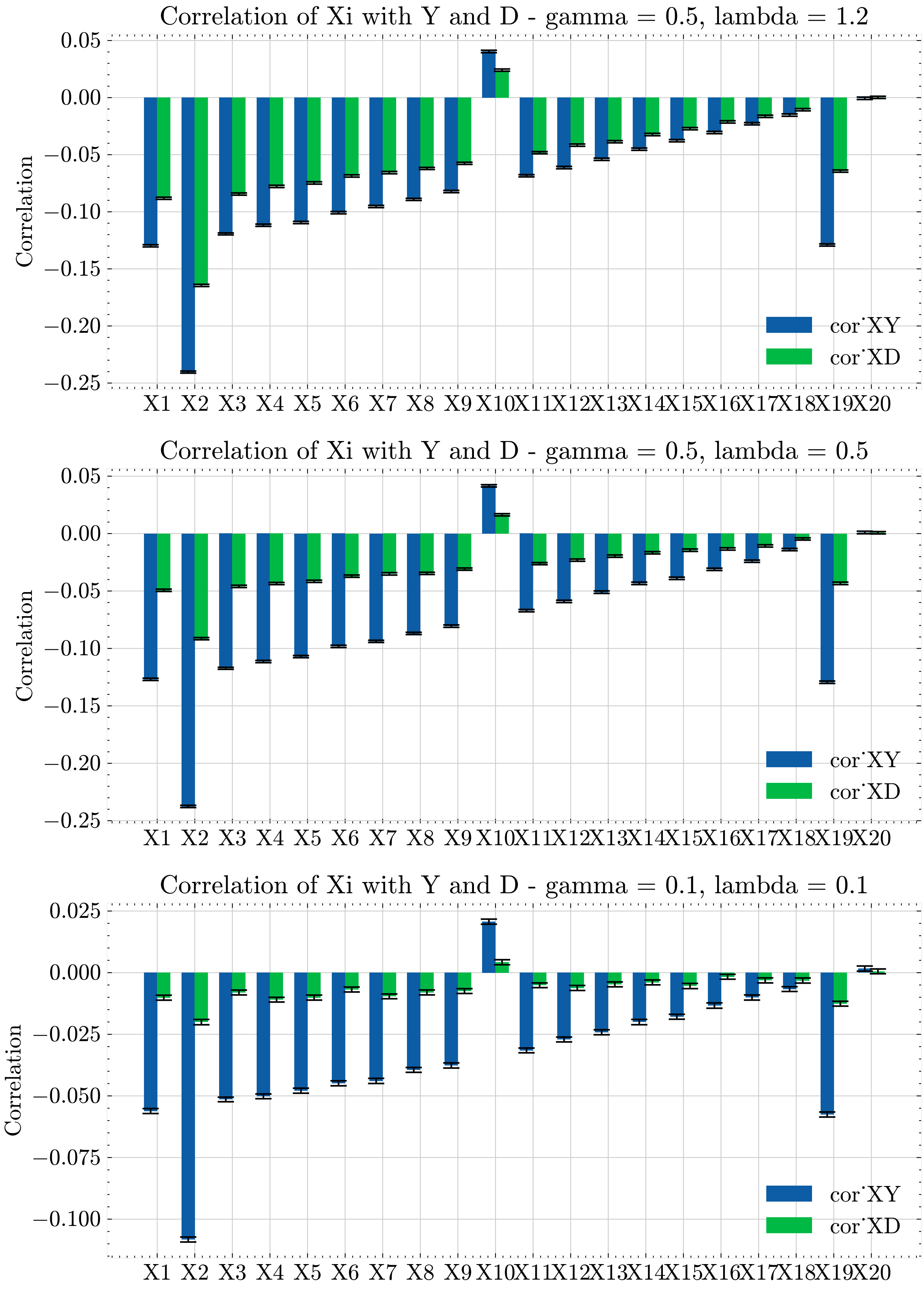


Figure 6: Appendix III: Correlation between X, D and X, Y Linear DGP
X-axis displays the confounders. For each X_j, the correlation between X and Y (blue) and X and D (green) is shown. The correlations are the mean aggregated over an iteration of 500 simulations. Black spreads at the top of the bars represent the standard error over the iterations. The specifications for the DGPs are: Linear DGP, N Obs = 2000, other parameters are displayed in title for each plot.



Correlation of Xi with Y and D - gamma = 0.5, lambda = 1.2

Correlation of Xi with Y and D - gamma = 0.5, lambda = 0.5

Correlation of Xi with Y and D - gamma = 0.1, lambda = 0.1

Figure 7: Appendix III: Correlation between X, D and X, Y Non-Linear DGP
X-axis displays the confounders. For each X_j, the correlation between X and Y (blue) and X and D (green) is shown. The correlations are the mean aggregated over an iteration of 500 simulations. Black spreads at the top of the bars represent the standard error over the iterations. The specifications for the DGPs are: Non-Linear DGP, N Obs = 2000, other parameters are displayed in title for each plot.

Table 18: Appendix III: Different Scenarios of Confounding Omitting

Linear DGP			**Non-Linear DGP**		
Omitted Conf.	Scenario	Corr	Omitted Conf.	Scenario	Corr
	No Omit	0.002861		No Omit	0.001674
X1	S	0.036756	X1	S	0.014206
X2	S	0.031259	X2	S	0.032060
			X3	S	0.055103
X15	W	0.00536	X17	W	0.007105
X16	W	0.004664	X18	W	0.004700
X1, X16	S W	0.038122	X2, X18	S W	0.033718
X1, X2	S S	0.064999	X1, X2	S S	0.040969
X15, X16	W W	0.007169	X17, X18	W W	0.009812
X10-X19	10W	0.037357	X8-X9, X11-X18	10W	0.036369
X1-X5, X15-X19	5S 5W	0.160913	X19, X1-X4, X14-X18	5S 5W	0.697172
X1-X10	10S	0.312133	X19, X1-X9, X19	10S	0.705027
X1-X20	All	0.486907	X1-X20	All	0.829484
X3-X20	All but 2S	0.202835	X1-X2, X4-X18, X20	All but 2S	0.250543

Values in Omitted Conf correspond to confounders position Xj where $j = \{1, \ldots, P\}$. In Scenario, S and W correspond to Strong and Weak. Corr. is correlelation measure The numbers in front of letters indicate the number of variables omitted. I.e. 5S meaning 5 strong variables. The specifications for both DGPs are: $\gamma = 0.5$, $\lambda = 0.5$.

Appendix IV: More Bias and MSE results from old Approach

In this Appendix the first approach for the simulation is displayed. It is not directly discussed in the main paper. However, the results might nontheless be of interest. The DGP differs only slightly from the Non-Linear DGP in the main paper. One significant difference is, that there is no λ included and furthermore, the threshold function to select the binary treatment variable does split the data into treated and non treated at the $median * 1.2$ instead of just the median. Additionally, the mean of all absolute terms $\overline{|(X_{ij} - r_j)|}$ is subtracted from the sum of absolute terms.

The DGP looks as followed:

$$\begin{align*}
Y_i &= \theta D_i + \gamma \pi_i + u_i \\
D_i &= f(D_i^c) \tag{12} \\
D_i^c &= \left(3 \cdot \frac{D_i^*}{\sigma_{D_i^*}}\right) \\
D_i^* &= \sum_{j=1}^{P} |X_{ij} - r_j| - \overline{\sum_{j=1}^{P} |X_{ij} - r_j|} \tag{13} \\
&\quad + X_{i1} X_{i(\frac{P}{2})} X_{iP} + \Big((X_{i2} - 3)(X_{i(\frac{P}{2})} + 3)\Big)^2 \\
&\quad + \ln\left(|X_{i3}|\right)\left(X_{i(P-1)} - 3\right) + v_i \\
\pi_i &= \sum_{j=1}^{P} |X_{ij} - r_j| - \overline{\sum_{j=1}^{P} |X_{ij} - r_j|} \tag{14} \\
&\quad + X_{i1} X_{i(\frac{P}{2})} X_{iP} + \Big((X_{i2} - 3)(X_{i(\frac{P}{2})} + 3)\Big)^2 \\
&\quad + \ln\left(|X_{i3}|\right)\left(X_{i(P-1)} - 3\right)
\end{align*}$$

What is also different is that instead of omitting variable in accordance with some scenarios the confounders are omitted as a share of all confounders. Regarding the result no method achieves accurate estimates with zero confounder omission. Linear Regression (LR) consistently underperforms across all stages and does not converge to similar bias levels at higher confounder omission percentages. The T-Learner and X-Learner follow, with XBCF initially displaying significantly lower bias. The DML method clearly outperforms all others, maintaining significantly lower bias and MSE across all levels of confounder omission. Both the bias and MSE of DML increase at a relatively stable rate with increased confounder omission, a pattern that also holds for the T-Learner, X-Learner, and LR. Same as in the main paper the DML method does perform better with lower number of observations, which is counterintuitive. The behavior that with confounders omitted the estimation becomes better can not be replicated here. Although, we have to keep in mind that the approach to omit confounders differs to the main paper.

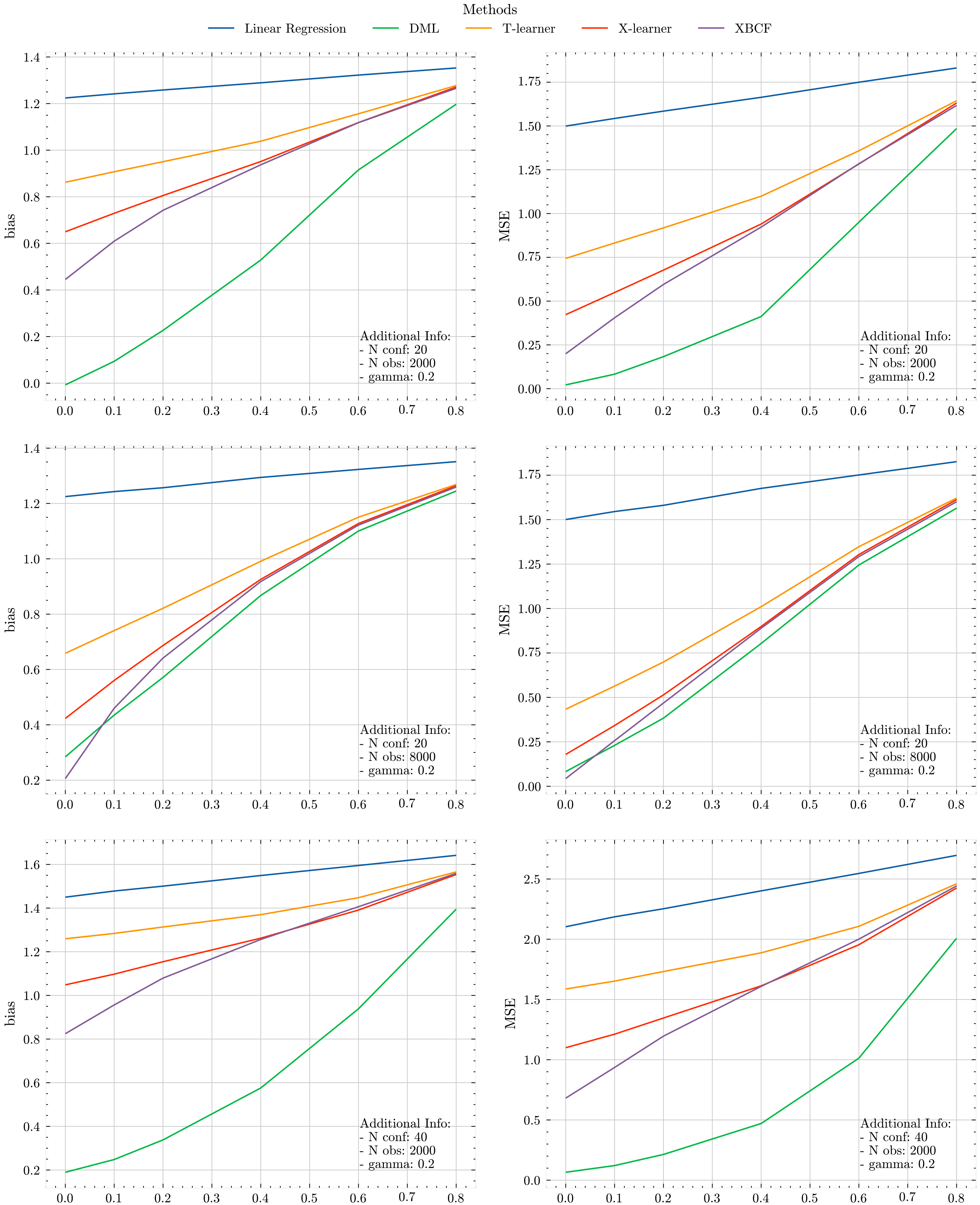


Figure 8: Appendix IV: Visualization of bias and MSE for all ML methods (old approach)
X-axis equals the share of omitted variables. Each row of visuals correspond to the same configuration of N confounders, N observations and gamma but differ by the measure.

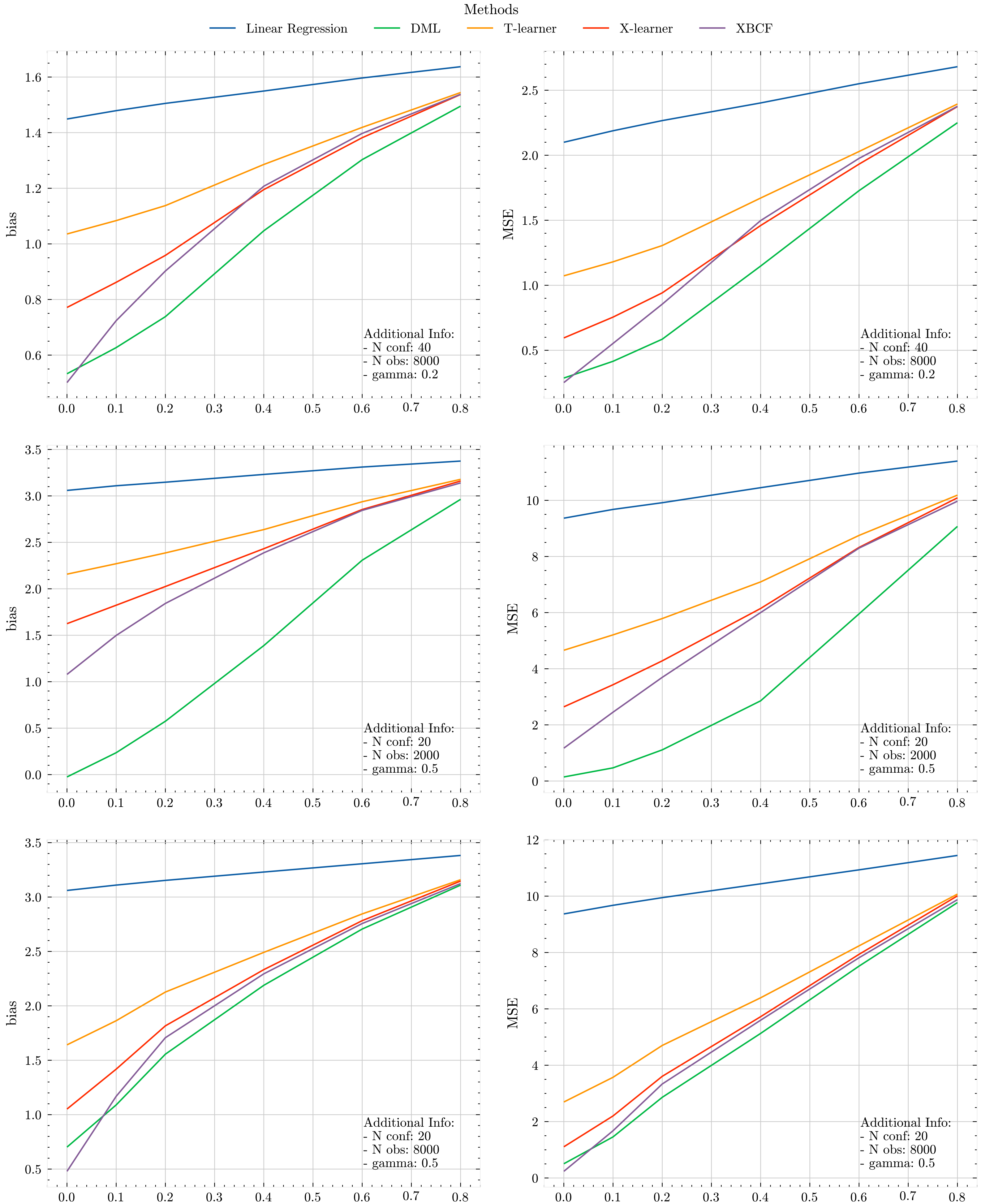


Figure 9: Appendix IV: Visualization of bias and MSE for all ML methods (old approach)
X-axis equals the share of omitted variables. Each row of visuals correspond to the same configuration of N confounders, N observations and gamma but differ by the measure.

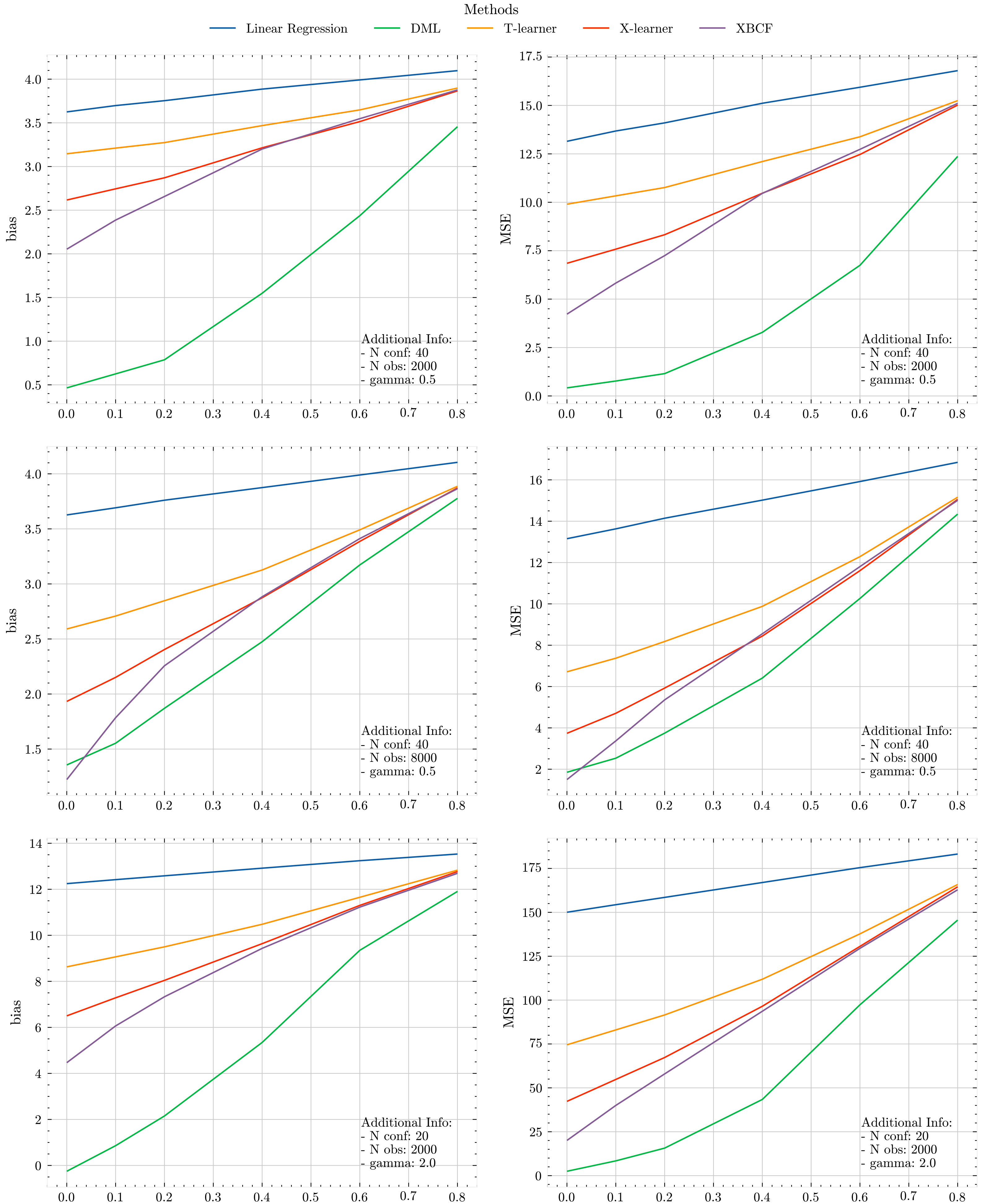


Figure 10: Appendix IV: Visualization of bias and MSE for all ML methods (old approach)
X-axis equals the share of omitted variables. Each row of visuals correspond to the same configuration of N confounders, N observations and gamma but differ by the measure.

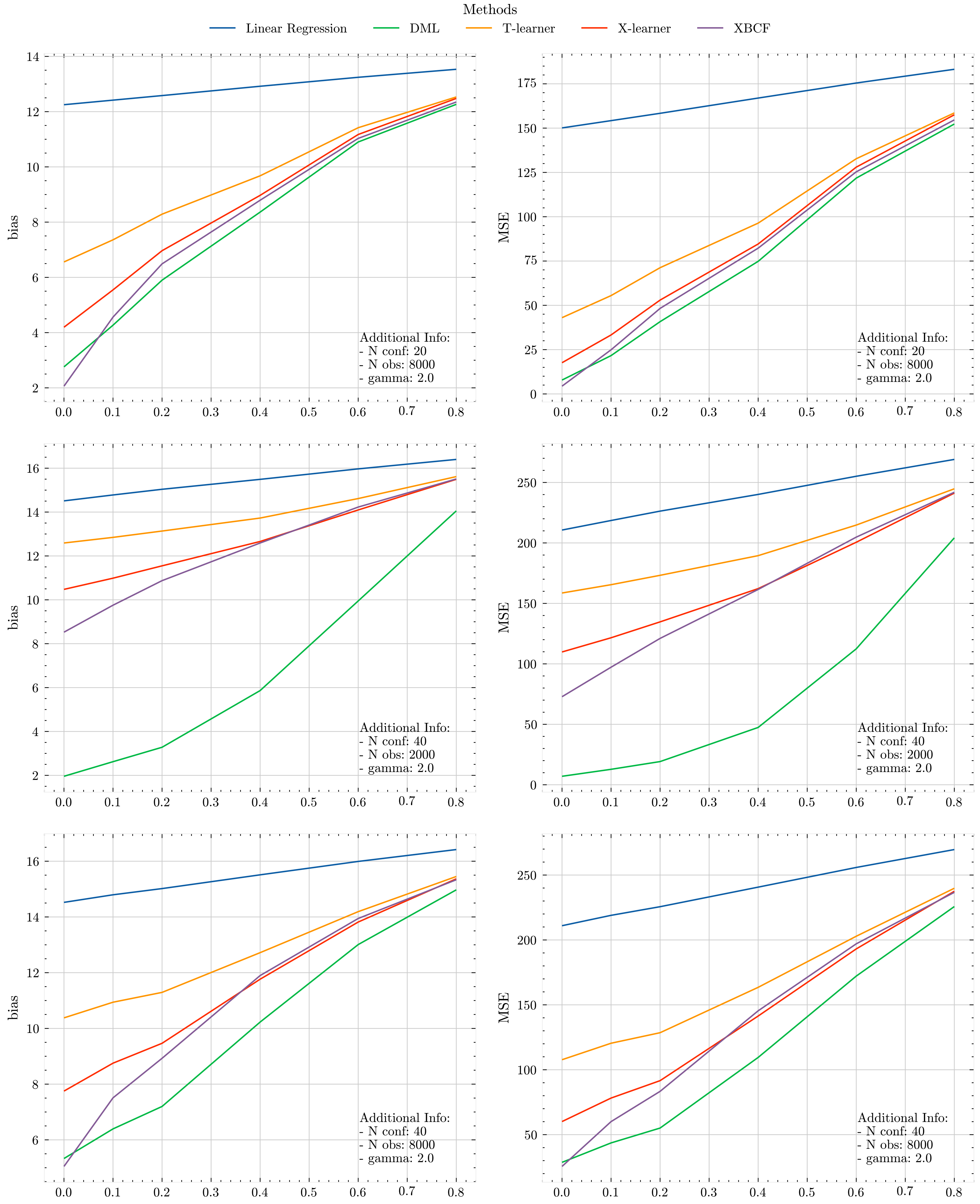


Figure 11: Appendix IV: Visualization of bias and MSE for all ML methods (old approach)
X-axis equals the share of omitted variables. Each row of visuals correspond to the same configuration of N confounders, N observations and gamma but differ by the measure.

Aid Specification

- OpenAI's ChatGPT-4 for coding assistance (Python), text editing, and support in LaTeX formatting.
- Visual Studio Code as the code editor and compiler.
- Python programming language for simulation implementation.
- GitHub's Copilot for coding assistance. Citavi for literature management.

Declaration of Authorship

I hereby declare:

- that I have written this thesis independently;
- that I have written the thesis using only the aids specified in the index;
- that all parts of the thesis produced with the help of aids have been declared;
- that I have handled both input and output responsibly when using AI. I confirm that I have therefore only read in public data or data released with consent, and that I have checked, declared, and comprehensibly referenced all results and/or other forms of AI assistance in the required form, and that I am aware that I am responsible if incorrect content, violations of data protection law, copyright law, or scientific misconduct (e.g., plagiarism) have also occurred unintentionally;
- that I have mentioned all sources used and cited them correctly according to established academic citation rules;
- that I have acquired all immaterial rights to any materials I may have used, such as images or graphics, or that these materials were created by me;
- that the topic, the thesis, or parts of it have not already been the object of any work or examination of another course unless this has been expressly agreed with the faculty member in advance and is stated as such in the thesis;
- that I am aware of the legal provisions regarding the publication and dissemination of parts or the entire thesis and that I comply with them accordingly;
- that I am aware that my thesis can be electronically checked for plagiarism and for third-party authorship of human or technical origin, and that I hereby grant the University of St.Gallen the copyright according to the Examination Regulations as far as it is necessary for administrative actions;
- that I am aware that the University will prosecute any violation of this Declaration of Authorship and that disciplinary as well as criminal consequences may result, which may lead to expulsion from the University or to the withdrawal of my title.

By uploading this academic term paper, I confirm through my conclusive action that I am submitting the Declaration of Authorship, that I have read and understood it, and that it is true.

17.11.24

Date and signature