Essays on Using Machine Learning for Causal Inference in Social Science

Dissertation
zur Erlangung des Doktorgrades
der Wirtschafts- und Sozialwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen

vorgelegt von

Jonathan Bernhard Fuhr

Tübingen

2024

Tag der mündlichen Prüfung:	**05.05.2025**
Dekanin und Dekan:	**Prof. Dr. Taiga Brahm &** **Prof. Dr. Dominik Papies**
1. Gutachter:	**Prof. Dr. Dominik Papies**
2. Gutachter:	**Jun.-Prof. Dr. Michael Knaus**

Danksagung

Auf dieser Arbeit steht mein Name, aber ohne eine Vielzahl anderer Personen wäre ich nie dazu gekommen, eine Promotion anzustreben, geschweige denn bis zur Abgabe durchzuhalten.

An erster Stelle möchte ich meinem Betreuer Prof. Dr. Dominik Papies danken, dessen Lehre und Forschung in mir die Leidenschaft geweckt hat, Daten gewissenhaft und nachvollziehbar zu analysieren, damit andere auf dieser Grundlage fundierte Entscheidungen treffen können. Du hast mich schon im Studium, aber erst recht in der Promotion intellektuell gefordert und gefördert, mir wiederholt viel zugetraut, aber gleichzeitig auch immer die Möglichkeit für Diskussion und Rücksprache gegeben. Auch zwischenmenschlich hast du wesentlich zu einer positiven Atmosphäre am Lehrstuhl beigetragen, sodass ich viel lieber im Büro als daheim gearbeitet habe. "Es wäre super, wenn" du die gemeinsame Zeit genauso positiv in Erinnerung behalten würdest wie ich!

Zahlreiche andere Kollegen haben meine Zeit an der Universität und darüber hinaus geprägt und bereichert, von denen ich im Folgenden einige stellvertretend nennen möchte. Dani, David und Leo: Danke, dass ihr mich in einer verrückten und schweren Zeit so herzlich aufgenommen habt! Dank euch habe ich mich jeden Tag auf's Büro gefreut, konnte viel mit euch lachen, aber auch in mancher Forschungskrise unsere Selbsthilfegruppe einberufen. David, ich danke dir ganz besonders für dein beständiges Dasein als Kollege und Freund, für die zahlreichen Diskussionen, Ermutigungen und unumstrittenen Codenames-Siege. Alex, danke, dass du mir auch nach allen Veränderungen am Lehrstuhl mit deinem offenen Ohr, deiner Fröhlichkeit und vielen hilfreichen Ideen und Zweitmeinungen zur Seite standest! Ebenfalls dankend erwähnen möchte ich Moni, Stefan und Aseem, die zu einem ausgezeichnet harmonierenden Team beigetragen haben. Außerdem danke ich Prof. Dr. Philipp Berens, der in seiner Rolle als Zweitbetreuer manche Themen aus einer für mich neuen Perspektive hilfreich hinterfragte, sowie Jun.-Prof. Dr. Michael Knaus für inhaltliche Diskussionen und seine sofortige Bereitschaft, die Dissertation zu begutachten. Zuletzt bedanke ich mich bei allen Verantwortlichen des Exzellenzclusters "Maschinelles Lernen: Neue Perspektiven für die Wissenschaft" für die Förderung meiner Doktorandenstelle und die Chance, von der interdisziplinären Community zu lernen.

Weil Freunde außerhalb des Kollegenkreises ebenfalls unweigerlich von den Nebenwirkungen einer Promotion betroffen sind, möchte ich mich namentlich bei Linda, Hari, Tobi, Melli, Lydi und der Familie Lindauer bedanken, die mit mir ausgehalten und mich immer wieder ermutigt haben. Auch Alex W., Albrecht W., Matthias S. und Andrew P. haben mich ein Stück des Weges begleitet und mir mit ihrer wertvollen Perspektive, guten Worten und gutem Rat zur Seite gestanden.

Meinen Eltern: Danke für alles! Ihr habt mir überhaupt erst Studium und Promotion ermöglicht und mich bedingungslos in allem unterstützt. Gemeinsam haben wir die letzten Jahre irgendwie gemeistert, und wir werden auch die kommenden Jahre zusammenhalten, egal was kommt. Ihr seid die Besten und ich bin sehr froh, euch zu haben!

Mein größter Dank gehört Gott, der ultimativen Ursache aller guten Dinge und damit auch aller meiner Gaben und Fähigkeiten, die in diese Arbeit geflossen sind. Praise the Maker!

Zusammenfassung

Die Verfügbarkeit enormer Datenmengen und die Entwicklung leistungsfähiger Algorithmen für maschinelles Lernen (ML) haben viele wissenschaftliche Disziplinen erheblich verändert. Die primär datengetriebenen Methoden des maschinellen Lernens sind jedoch nur begrenzt dazu geeignet, *kausale* Fragen anhand von Beobachtungsdaten zu beantworten, da hierfür üblicherweise Annahmen erforderlich sind, die durch wissenschaftliche Theorien gestützt werden. Dennoch haben Forscher in den vergangenen Jahren mehrere Ansätze vorgeschlagen, die ML einsetzen, um einige der gängigen Annahmen für kausale Inferenz abzuschwächen. Die grundlegende statistische Theorie für viele dieser Ansätze wurde zwar entwickelt, aber für anwendungsorientierte Forscher ist oft nicht unmittelbar ersichtlich, wie gut diese Methoden in realistischen Situationen funktionieren, unter welchen Umständen sie scheitern könnten und wie plausibel die zugrundeliegenden Annahmen sind. Um diese Lücke zu schließen, arbeitet diese Dissertation einige dieser neuen Entwicklungen aus einer sozialwissenschaftlichen Perspektive auf, untersucht sie empirisch, erweitert sie für komplexere Situationen und wendet sie an. Der erste Aufsatz (Kapitel 2) evaluiert, wie die populäre Methode des Double/Debiased Machine Learning (DML) es ermöglicht, bei der Schätzung von kausalen Effekten flexibel für andere Einflussgrößen zu kontrollieren. Gleichzeitig zeigt dieses Kapitel die Auswirkungen verschiedener Forscherentscheidungen bei Anwendung der Methode auf und liefert konkrete Empfehlungen für die bestmögliche Umsetzung in der Praxis. Der zweite Aufsatz (Kapitel 3) erörtert und analysiert die Herausforderungen bei der Anpassung von DML für Situationen, in denen Paneldaten verfügbar sind und unbeobachtete Heterogenität vorliegen könnte. Letztendlich schlägt dieser Beitrag einen Ansatz basierend auf Correlated Random Effects vor, der sowohl mit unbeobachteter Heterogenität als auch mit nichtlinearen beobachtbaren Störeinflüssen umgehen kann. Nach einem Überblick über die grundlegenden Konzepte auf dem Forschungsgebiet Causal Discovery wird im dritten Beitrag (Kapitel 4) anhand von Simulationen und Anwendungen gezeigt, dass das Erlernen von kausaler Struktur aus Beobachtungsdaten in den Sozialwissenschaften eine große Herausforderung darstellt und starke Annahmen erfordert, die schwer zu überprüfen sind. Zusammenfassend zeigt diese Dissertation, wie sich kausale Inferenz und datengetriebene ML-Ansätze in den Sozialwissenschaften ergänzen können, wenn man sie angemessen einsetzt, die zugrundeliegenden und nicht überprüfbaren Annahmen transparent darlegt, und diese anhand von Theorie und Fachkenntnissen für konkrete Anwendungen begründet. Damit bietet diese Arbeit anwendungsorientierten Forschern einen Leitfaden für die Einschätzung und Anwendung neuartiger Methoden, die maschinelles Lernen für kausale Fragen nutzen.

Summary

The availability of vast amounts of data and the development of powerful machine learning (ML) algorithms have had a major impact on many scientific disciplines. However, mainly data-driven ML methods are limited in their ability to answer *causal* questions from observational data, for which researchers traditionally rely on assumptions substantiated by scientific theory. Nevertheless, researchers have recently suggested several approaches that utilize ML to relax some of the conventional assumptions in causal inference. While the general statistical theory for many of these approaches is established, it is not immediately apparent to applied researchers how these methods perform in realistic settings, under which conditions they might fail, and how plausible the underlying assumptions are. To address this gap, this thesis reviews, empirically evaluates, extends, and applies some of these new developments from a social science perspective. The first paper (Chapter 2) evaluates how the popular double/debiased machine learning (DML) approach enables flexible covariate adjustment when estimating causal effects, demonstrates the impact of various researcher decisions in the implementation process, and provides actionable best practice recommendations for the application of the method. The second paper (Chapter 3) discusses and assesses the challenges of extending DML to settings with panel data and unobserved heterogeneity, finally suggesting that a strategy with predictors based on the correlated random effects approach can handle both the unobserved heterogeneity and nonlinear observed confounding. After reviewing the basic concepts of the causal discovery field, the third paper (Chapter 4) through simulation and application finds that learning causal structure from observational data in social science is very challenging and requires strong assumptions that are difficult to assess. In sum, this dissertation demonstrates how causal inference and data-driven ML approaches can complement each other in social science, provided that researchers use them within appropriate frameworks, clarify the underlying untestable assumptions, and justify these from theory and domain knowledge for specific applications. In doing so, this thesis offers guidance to applied researchers for how to evaluate and apply novel methods that use machine learning for causal questions.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Parts of Chapter 1 are based on the introduction of the working paper "Fuhr, J., Berens, P., and Papies, D. (2024). Estimating Causal Effects with Double Machine Learning – A Method Evaluation. arXiv:2403.14385 [cs, econ, stat]."

In the past two decades, two developments have substantially influenced both society and the scientific domain: the availability of "Big Data", and the design of new methods and algorithms that can analyze this kind of data. First, technological advances like the internet, smartphones, e-commerce, sensors, IoT devices, etc. have led to an enormous amount, frequency, and variety of data. This data is available to governments, companies, scientists, and others, and analyzing the information contained has a tremendous potential for improved decision-making (McAfee and Brynjolfsson, 2012). Second, methodological advances in artificial intelligence (AI), and, more specifically, machine learning (ML), have made it possible to find complex patterns in such large amounts of data and thus solve increasingly difficult tasks. Examples range from product recommendations to image recognition, fraud detection (e.g., Mayer-Schönberger and Cukier, 2013), and, more recently, chatbots such as ChatGPT based on large language models (LLMs) (e.g., OpenAI et al., 2024). In science, researchers have successfully used these ML methods to, for example, accurately predict protein structure (Jumper et al., 2021), analyze medical images (Ronneberger et al., 2015), or to predict credit repayment from mobile phone usage (Björkegren and Grissen, 2020).

The fruitful combination of Big Data with new data-driven methods in science has caused an enthusiastic optimism in some circles (see Kitchin, 2014). Already in 2008, the editor-in-chief of the *Wired* magazine published a provocative but influential article with the title "The End of Theory: The Data Deluge Makes the Scientific Method Obsolete" (Anderson, 2008). In it, Chris Anderson claims that massive data and statistical algorithms can replace scientific models and theory, stating that given these new circumstances, "Correlation supersedes causation, and science can advance even without coherent models, unified theories, or really any mechanistic explanation at all", and "With enough data, the numbers speak for themselves". While certainly an extreme position, others have also been very optimistic about the potential of Big Data for science. Despite being critical of Anderson's article, in a well-cited book, Mayer-Schönberger and Cukier (2013, Chapter 4) argue that correlational analyses are "good enough" in many settings, asserting that Big Data can often replace causal research or even disprove causal claims.

Such optimism has generated significant opposition, primarily pointing out the need for theory and models when the goal are *causal* explanations that can answer the "why" questions ubiquitous across scientific disciplines (e.g., Mazzocchi, 2015; Naimi and Westreich, 2014). However, these articles explicitly do not pit data-driven and theory-based approaches against each

other, but suggest that the combination of both can lead to new opportunities and progress in science (e.g., Leavitt et al., 2021).

Since the earlier enthusiasm, there has been continuous progress in the development of data-driven methods. As researchers in more and more scientific disciplines are applying ML, it is natural to ask where the limits for learning directly from data really are. More specifically, since many critics have identified causal questions as one of the main limitations of data-driven approaches, what can these data-driven methods contribute to the field of causal inference?

Causal questions are not a marginal issue, they are at the heart of most research in the natural and social sciences (Pearl et al., 2016). Across scientific disciplines, researchers attempt to learn about causal relationships (Imbens and Rubin, 2015), i.e., they ask how one variable of interest (the "treatment") causally affects another variable (the "outcome"). For example, labor economists care about the effect of education on wages (e.g., Card, 1999), physicians want to know whether smoking causes lung cancer (e.g., Cornfield et al., 2009), and marketing researchers want to know how a price change affects demand (e.g., Bijmolt et al., 2005). For many of these questions, researchers have to rely on observational data because experimental interventions may be infeasible, unethical, or simply too costly to obtain (e.g., Athey and Imbens, 2017). However, prior research on causal inference has emphatically argued that without experimental variation, identifying and estimating causal effects is not possible without making several assumptions (e.g., Pearl, 2009). These assumptions can be strong and are generally not testable from data in observational studies, irrespective of the data size, so researchers need to substantiate them from theory and domain knowledge (e.g., Imbens and Rubin, 2015). Justifying the necessary assumptions is one of the biggest challenges in real-world applications of causal inference (e.g., Hernán and Robins, 2020). As a consequence, researchers are interested in developing and applying methods that work under weaker or more plausible assumptions.

Figure 1.1 depicts the main components of a classical causal inference workflow: causal structure, causal identification, and causal estimation (see also Dang et al., 2023; Feuerriegel et al., 2024). Each of these components relies on assumptions that researchers typically make and assess based on theory and domain knowledge of the specific field.

The first step is specifying the causal structure of the problem, which is a description of the assumed data-generating process behind a particular problem. Such a causal model describes the qualitative causal relationships between different variables and can be encoded in a causal graph or via mathematical equations (e.g., Imbens and Rubin, 2015; Pearl et al., 2016). Researchers

usually derive the causal structure from expert knowledge and current theory in the literature of the specific causal question.

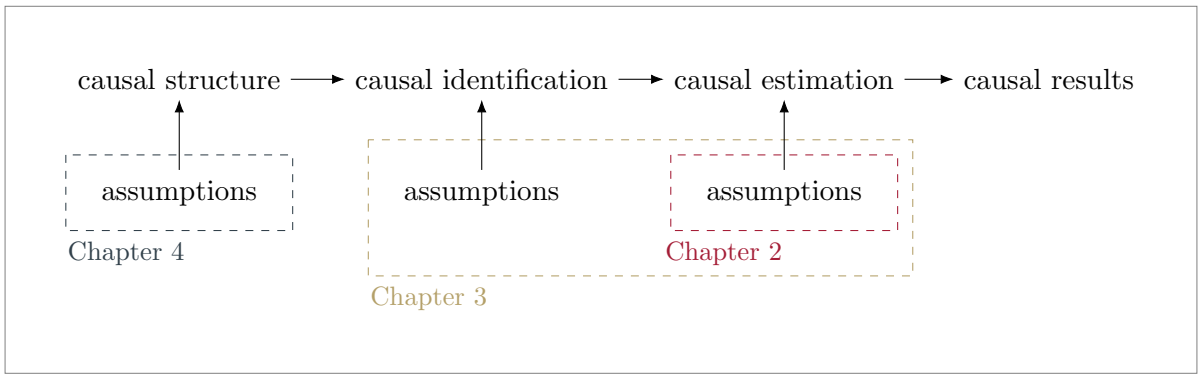


Figure 1.1: A simplified representation of a typical causal inference workflow. Each of the three main steps relies on assumptions. The dashed boxes indicate at which step(s) each chapter addresses conventional assumptions.

Given a causal structure, the next step is the assessment of causal identification, i.e., determining whether one can uniquely answer the causal question of interest with observable data (e.g., Hernán and Robins, 2020, Chapter 3). This step again critically hinges on assumptions, some of which can already be part of the causal structure. For example, the unconfoundedness assumption asserts that all variables influencing both the treatment and the outcome ("confounders") are part of the observed data and can be used for adjustment (e.g., Imbens and Wooldridge, 2009). Different approaches can rely on different identification assumptions, but if any of the identifying assumptions underlying a particular methodology are violated, the observed data is not sufficient to compute a causal effect using this approach (e.g., Hernán and Robins, 2020).

Thirdly, given a causal structure and a valid identification strategy, researchers can use observable data to *estimate* the causal effect of interest. This involves selecting a statistical model and estimator, which researchers also often base on assumptions. For example, if identification is possible by adjusting for observed confounders, researchers commonly use an ordinary least squares (OLS) regression of the outcome on the treatment and the other variables needed for adjustment, thereby imposing assumptions on specific (e.g., linear) functional forms (e.g., Wooldridge, 2010).

Only if the main assumptions of each step in the workflow hold, we can correctly interpret the resulting estimates as representing causal effects. Naturally, finding ways to relax some of these assumptions is an important goal of new developments in causal inference.

Indeed, researchers have recently suggested that we can relax some of the classically imposed assumptions by relying on new causal inference approaches based on data-driven methods and machine learning. Machine learning in its arguably most successful form, supervised ML, has established itself as a powerful tool for making *predictions* in complex, nonlinear settings. ML methods are capable of handling high-dimensional data, i.e., data where the number of variables or parameters may even exceed the number of observations (see, e.g., Hastie et al., 2009). However, the goal of achieving high predictive accuracy is fundamentally different from the main goal of causal inference, which typically is accurate parameter or effect estimation (Mullainathan and Spiess, 2017; Shmueli, 2010). For causal inference, establishing identification is essential, whereas for predictive ML, causal relationships are not of primary interest, as long as the model predicts well on unseen data. Also, researchers can evaluate the performance of predictive methods on data they did not use for training, but there is usually no ground truth available to evaluate causal inference methods, so we must assess the underlying assumptions on the basis of substantive arguments (Athey, 2019). Thus, in a sense, answering causal questions is often harder than solving classical machine learning problems (Peters et al., 2017, Preface). A consequence of the different goals of ML and causal inference is that directly using ML methods designed for prediction does not lead to parameter estimates that we should interpret as causal effects (Athey, 2019). Since there are other forms of ML beyond the predictive/supervised ML, for this thesis, I will define machine learning as a field developing algorithms that are primarily data-driven and often rely on only relatively weak assumptions (see also Athey, 2017). By contrast, the field of causal inference traditionally relies on rather strong assumptions, but has perhaps not always fully exploited the potential of learning from data (e.g., Varian, 2014).

Given these fundamental differences between the traditional approaches of machine learning and causal inference, researchers have studied what each field can learn from the other. More specifically – and this is the focus of this thesis – they have explored what we can learn about causality from observational data, and how ML might help to relax traditional assumptions in causal inference. Researchers have suggested contributions of ML in each of the three steps in the causal workflow outlined above: learning causal structure from data (e.g., Glymour et al., 2019; Peters et al., 2017), relaxing causal identification assumptions (e.g., Burauel, 2023; Wang and Blei, 2019), and relying on weaker assumptions in the estimation of causal effects (e.g., Chernozhukov et al., 2018; Wager and Athey, 2018). For many of these new methods, general statistical properties are established, but their performance in realistic settings, their boundary

conditions, and the plausibility of their underlying assumptions are often not obvious to applied researchers. The goal of this thesis is to bridge this gap by critically reviewing, evaluating, extending, and applying some of these new developments seemingly most relevant for applied causal research in the social sciences. This will contribute to clarify at which stages of the traditional causal inference workflow ML can support, and where it is unlikely that data-driven methods will replace theory and expert knowledge. While this thesis primarily takes a social science perspective on these issues, many of the results will also be relevant for other disciplines working with observational data.

The thesis is structured along the causal inference workflow in Figure 1.1. After the overall introduction in Chapter 1, each of the subsequent three chapters explores whether and how ML can relax assumptions at different stages in the causal workflow. I start at the final stage, where methods are most mature, and then move back in the workflow to more ambitious methods, addressing assumptions that might be more difficult to relax. The colored boxes in Figure 1.1 indicate at which stage(s) in the workflow each chapter investigates the potential to relax conventional assumptions.

Chapter 2 starts at the final step of the workflow, causal estimation, where some assumptions are often not based on strong theory and seem most feasible to relax. In this joint work with Philipp Berens and Dominik Papies, we focus on one of the most popular methods that uses ML for the estimation of causal effects, called "double/debiased machine learning" (DML) by Chernozhukov et al. (2018). By incorporating flexible ML methods in a specific estimation framework, DML enables researchers to relax assumptions about functional forms of confounding relationships. In our article, we review the method, empirically evaluate the performance of several possible implementation choices in many different settings of simulated data, and illustrate DML's application to real world data. Based on our overall findings, we derive specific recommendations for the many decisions applied researchers face when utilizing DML for their causal questions.

In settings with cross-sectional data, data-driven predictive methods within DML can help to relax causal estimation assumptions. However, in social science research, *panel data* is ubiquitous. Researchers often can exploit these repeated observations of identical units to remove unobserved heterogeneity between units, thereby relying on weaker causal identification assumptions. Thus, in Chapter 3 (joint work with Dominik Papies), we aim to adapt DML to settings with panel data and unobserved heterogeneity. Even though ML is not directly responsible for re-

laxing identification assumptions in this setting, such an adaptation would allow causal inference under both weaker estimation *and* identification assumptions. We first highlight the challenges of using DML for panel data, then discuss multiple intuitive ways of adapting DML to account for the panel data structure and unobserved heterogeneity, before assessing the performance of these different approaches in various simulation settings. Finally, we give recommendations for how applied researchers can use DML when they have access to panel data.

The single-authored Chapter 4 deals with assumptions in the first step of the causal inference workflow and explores what we can learn about *causal structure* directly from observational data. In social science, assumptions about causal structure almost exclusively originate from theory and domain knowledge. However, a literature from philosophy and computer science has developed algorithms that aim to *discover* such structure from observational data. If successful, this would relax the assumption that there is sufficient prior knowledge about the causal structure of the particular problem. Hence, in this article, I assess how relevant and useful some of the most popular of these *causal discovery* methods are for applied research in social science. After reviewing the essential methodology of the field, I focus on two algorithms with different underlying assumptions, evaluate their performance in simulations, and apply them to a classical social science dataset often used for new causal inference methods. I follow this analysis with a critical discussion of the utility of causal discovery for typical problems in social science.

The final Chapter 5 summarizes and connects the findings across the three projects, concluding with a discussion of the potential and limitations of data-driven methods for answering causal questions.

Overall, this thesis aims to contribute to the literature about machine learning for causal inference by critically evaluating the value and constraints of recently suggested approaches, extending their applicability beyond the original settings, and providing guidance for applied researchers about whether and how to adopt these novel developments.

Chapter 2

Estimating Causal Effects with Double Machine Learning – A Method Evaluation

Jonathan Fuhr, Philipp Berens, and Dominik Papies

Statement of contribution

Jonathan Fuhr conducted all literature review, data collection and simulation, implemented the methods and analyses, and wrote the first draft of the manuscript. Philipp Berens provided supervision, feedback, and revised the working paper. Dominik Papies conceived the initial idea, provided continuous supervision and feedback, and revised the working paper.

Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC number 2064/1 – Project number 390727645. The authors thank Michael Knaus for his excellent feedback on an earlier version of the working paper. Further, they acknowledge all participants of the Research Seminar Current Trends in Marketing Analytics 2021 in Cologne (Germany) and the European Marketing Academy Conference 2023 in Odense (Denmark) for their valuable comments. The authors acknowledge support by the state of Baden-Württemberg through bwHPC. Philipp Berens acknowledges support by the Hertie Foundation.

Chapter 2 is based on the working paper "Fuhr, J., Berens, P., and Papies, D. (2024). Estimating Causal Effects with Double Machine Learning – A Method Evaluation. arXiv:2403.14385 [cs, econ, stat]".

Abstract

The estimation of causal effects with observational data continues to be a very active research area. In recent years, researchers have developed new frameworks which use machine learning to relax classical assumptions typically imposed for the estimation of causal effects. In this paper, we review one of the most prominent methods - "double/debiased machine learning" (DML) - and empirically evaluate it by comparing its performance on simulated data relative to more traditional statistical methods, before applying it to real-world data. Our findings indicate that the application of suitably flexible machine learning algorithms within DML improves the adjustment for various nonlinear confounding relationships. This advantage enables a departure from traditional functional form assumptions typically made in causal effect estimation. However, we demonstrate that the method continues to critically depend on standard assumptions about causal structure and identification. When estimating the effects of air pollution on housing prices in our application, we find that DML estimates are consistently larger than estimates of less flexible methods. From our overall results, we provide actionable recommendations for specific choices researchers face when applying DML in practice.

2.1 Introduction

When estimating causal effects from observational data, researchers typically rely on several assumptions in the different stages of the causal inference workflow. One of the most utilized assumptions regarding causal structure and identification is that all relevant variables influencing both treatment and outcome are observed ("unconfoundedness") (e.g., Imbens, 2004). However, even given this or similar assumptions, researchers often impose additional assumptions to facilitate the estimation process. For example, a popular method to estimate causal effects under unconfoundedness is computing an OLS regression of the outcome variable on the treatment variable and the other covariates (e.g., Wooldridge, 2012). This approach assumes that the specified parametric model is correct, i.e., that it contains all variables with the appropriate interactions and functional forms. Conventionally, researchers include most variables linearly or with simple transformations (e.g., Wooldridge, 2012), even when there is little theoretical foundation to support these specific functional forms (Dang et al., 2023).

By contrast, the field of supervised machine learning (which we focus on in this chapter) has excelled in learning complex nonlinear functional forms from data, even in high-dimensional settings (e.g., Hastie et al., 2009). However, supervised ML does so in order to accurately *predict* an outcome variable from other covariates (or *features*) on unseen data. Predicting values of the outcome on an independent dataset is a fundamentally different task than estimating accurate causal effects (Shmueli, 2010), since the latter aims to predict what would happen under an intervention on the treatment variable (e.g., Peters et al., 2017). A consequence of these different goals is that directly using ML methods designed for prediction "off-the-shelf" can potentially lead to biased parameter estimates, even if the identifying assumptions hold (Athey, 2019). In other words, these methods do not usually provide unbiased estimates of causal effects, i.e., estimates that are in expectation equal to the true parameter value (Wooldridge, 2012). However, there may be a way to use "machine learning in the service of causal inference" (Mullainathan and Spiess, 2017): The core tenet underlying this idea is that, in addition to causal structure and identification assumptions, answering causal questions requires an estimation process that often contains a prediction part. For example, the first stage of instrumental variable estimation is a prediction task: We predict the treatment from the instrument(s) (Mullainathan and Spiess, 2017). Using data-driven ML for such predictive parts may facilitate a higher flexibility and may potentially allow for less restrictive functional form assumptions in the estimation process.

In this paper, we focus on "double/debiased machine learning" (DML) by Chernozhukov et al. (2018), which is arguably one of the most prominent examples of a method using ML for causal inference. The basic promise of DML is that we can use ML methods to flexibly adjust for observed confounding variables. With that, researchers can still obtain unbiased estimates, even in settings with potentially many confounders and complex functional forms. By using DML with flexible ML algorithms instead of specifying a parametric model, researchers may be able to relax assumptions about how (which variables, which functional forms) they adjust for observed confounding. This is important because flexibly adjusting for a large number of covariates can increase the plausibility of the assumption that all relevant confounding variation has been considered (Belloni et al., 2016).

It is important that applied researchers seeking to use DML fully appreciate all potential benefits, pitfalls, and assumptions of this method. While certain properties of DML might seem obvious to some readers, history has shown the need for transparent evaluation and communication of new methodological developments to prevent flawed applications resting on implausible assumptions (e.g., Rossi, 2014). In addition, when using DML, researchers face a variety of choices and must answer questions such as: "Which variables should enter the estimation? Which ML algorithms should I use? Do these decisions depend on the sample size? How influential are these choices?" With these considerations and questions in mind, we see five important aspects missing from the literature at the time of writing (March 2023): (1) an extensive review and discussion of DML, focused on an intuitive understanding and an assessment of the necessary assumptions, (2) a review of published applications using DML, documenting substantial heterogeneity in implementation choices, (3) an empirical evaluation of the method's performance, i.e., its ability to recover causal effects in a wide range of simulated settings mimicking causal problems encountered in applied fields, (4) specific guidance for the many decisions researchers have to make when applying DML to their causal questions, and (5) an application of various implementations of DML to real-world data. To address these voids, we review and evaluate the method and compare it to more traditional statistical methods in both simulations and an application, from which we provide specific and actionable best-practice guidance to applied researchers.

To preview the results, a first finding is that the functional form of the confounding as well as the number of confounders strongly affect the suitability of specific ML algorithms for DML. More specifically, while many applications in the past have used lasso regression in the context

of DML, we caution against its use in this context because DML using lasso without manual variable transformations produces biased estimates in the presence of nonlinear confounding. Second, the results suggest that the main advantage of DML with flexible ML methods is its ability to adjust for nonlinear confounding without knowing the underlying functional forms, rather than adjusting for a very large number of important confounders simultaneously. Third, we find that gradient boosting (XGBoost) performs very well across a broad range of settings in our analyses, which is why we recommend it as a baseline or default method within DML. Further, the results also show that DML continues to critically hinge on researchers' input about causal structure and is no automatic remedy for unobserved confounding or bad controls (Cinelli et al., 2024). Finally, to support researchers in their choice of a suitable ML algorithm, we present a simple metric that researchers can use to aid their selection of ML algorithms.

We structure the rest of the paper as follows. In Section 2.2, we review the current literature around DML. Section 2.3 provides a mostly non-technical review of the method, focusing on the partially linear model. In Section 2.4, we assess DML in a variety of simulation settings, where we compare its performance to more traditional methods and evaluate strengths and weaknesses. Section 2.5 applies DML to real-world data, which leads to a discussion of the plausibility of the results and the remaining assumptions. Section 2.6 briefly outlines the applicability of DML in settings beyond the partially linear model. In Section 2.7, we derive recommendations of best practice for applied researchers about when and how to apply this method, and finally conclude our discussion.

2.2 Literature review

We review the literature related to double/debiased machine learning (Chernozhukov et al., 2018) from three perspectives: (1) the history and purpose of DML, (2) a quantitative analysis of published applications of DML, and (3) method evaluations considering DML.

(1) First, one central purpose of DML is relaxing assumptions about model specification and functional forms of covariates. Even in cases where all confounding variables are observed, including them with a wrong functional form in a parametric regression model will lead to biased estimates of the treatment effect (e.g., Wooldridge, 2012, Chapter 9). Classical semiparametric methods, which only make functional form assumptions about the treatment parameter, but not about the covariates, may alleviate this problem (Athey, 2019). However, they tend to be slow to

converge and require more observations compared to parametric methods (Powell, 1994). These methods are especially inadequate in high-dimensional settings, i.e., when the number of parameters to estimate is large relative to the sample size. On the one hand, this can occur through having more variables/features than observations, which is problematic for many traditional approaches in statistics (e.g., James et al., 2021, Chapter 6). On the other hand, this can also occur when including various transformations and interactions of a relatively low-dimensional set of covariates to be more robust against model misspecification. Estimating parameters for each of these transformations could quickly lead to a high-dimensional setting (Belloni et al., 2014). One example for a semiparametric regression method is Robinson (1988), who models the relationship between covariates and both outcome and treatment using a kernel regression. DML builds on this method and provides a framework in which the kernel regressions can be replaced by modern ML methods (Chernozhukov et al., 2018), which enables the application in high-dimensional settings. In a parallel development within biostatistics, Laan and Rubin (2006) developed "Targeted Maximum Likelihood Estimation" (TMLE), a semiparametric estimation technique similar to DML that also allows for the use of ML methods. For a detailed review and comparison of TMLE and DML from a biostatistics perspective, see Diaz (2020).

A more direct predecessor of DML is the "double selection" procedure (Belloni et al., 2014), which consists of three steps: First, it uses lasso regression as the ML method to select which covariates are predictive of the outcome; second, it uses lasso to select which covariates are predictive of the treatment; third, it uses OLS to regress the outcome on the treatment and the union of all covariates selected in the first two steps. This method can deliver unbiased estimates in situations where there are many potential confounding variables (potentially more than observations) from which only a few are important for adjustment (the "sparsity assumption") (Belloni et al., 2014). DML generalizes this idea and introduces a sample splitting procedure, which facilitates the application of many modern ML methods beyond lasso regression.

(2) As a second perspective, we review applications of DML published across disciplines. Overall, we have identified 46 published papers containing applications of DML to real-world data, of which 36 contained sufficient information about the implementation to be considered in this overview (see Appendix A.1 for a list of these papers and a detailed description of our selection process). We summarize important characteristics of these applications (Figure 2.1). Since Chernozhukov et al. (2018) published the original paper in econometrics, a majority of subsequent applications occurred either directly in economics, or as part of further method devel-

opments within statistics or econometrics. However, the method has also received widespread attention in other quantitative disciplines such as healthcare/medicine or sociology (Figure 2.1A).

Within DML, researchers most often used lasso and random forests for the predictive parts, followed by boosting methods (Figure 2.1B). On average, an application considered 1.55 different ML methods. However, most papers (27/36, 75%) used only one ML algorithm and do not assess the robustness of their estimates to different predictive methods. A likely reason for the dominance of lasso is the early implementation of DML in the statistical software Stata (StataCorp, 2019), which uses lasso and which multiple papers explicitly mentioned.

A majority of the applications investigate effects of a binary treatment variable (Figure 2.1C), but there are also many continuous treatments and a few categorical/multilevel treatment settings.

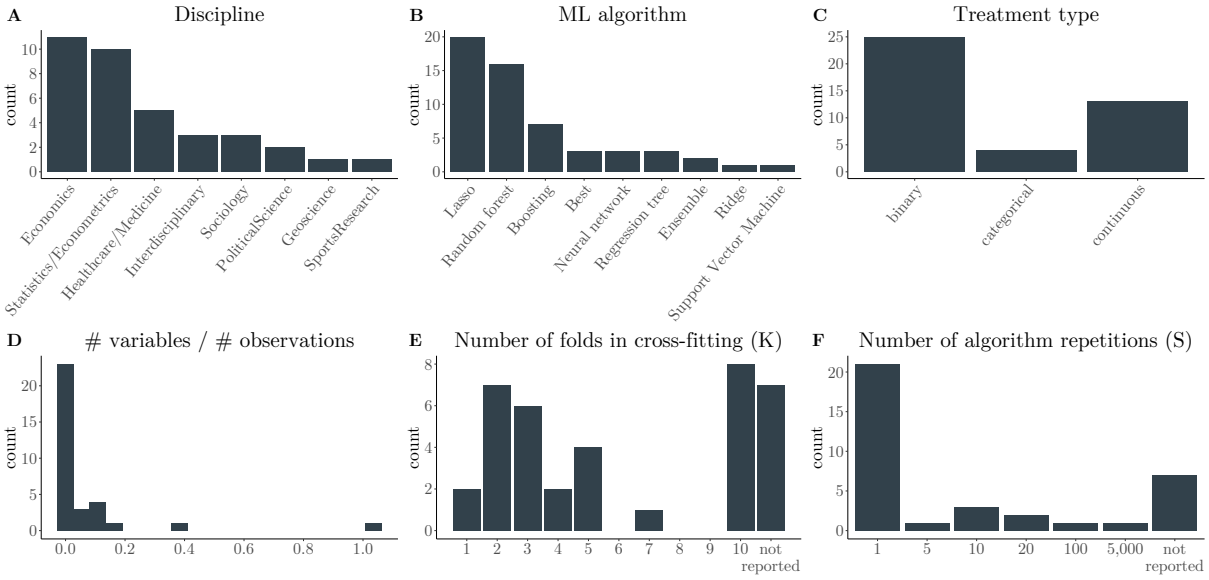


Figure 2.1: Overview of DML applications in the literature. **A** Discipline the application was published in. **B** Different ML algorithms used within DML. **C** Treatment type considered in application. **D** Dimensionality: ratio of the number of variables to the number of observations. **E** Number of folds the data is split into within DML. **F** Number of algorithm repetitions for increased robustness.

One advantage of DML is its ability for valid inference in high-dimensional settings. However, there are arguably few real-world applications where the number of raw variables exceeds the number of observations. Hence, high-dimensionality in practice might rather be a result of flexibly modeling nonlinear relationships with many parameters. In the applications of DML in the literature, the distribution of the ratio of the number of variables to the number of observations supports this hypothesis (Figure 2.1D). We define the number of variables as the raw covariates, not including transformations such as polynomials or interactions. Only one appli-

cation is high-dimensional in raw covariates: Chan and Meunier (2022) estimate the effect of technological intensity on support for the EU FDI screening mechanisms, using one observation for each of the 28 EU member states ($n = 28$) and adjusting for 29 variables. In all other applications, the number of raw covariates is significantly smaller than the number of observations. However, these settings can still benefit from high-dimensional methods if one includes many transformations of the raw covariates or if one uses flexible nonparametric ML methods like random forests or neural networks, which estimate many parameters in the training process to fit complex functional forms.

Finally, when applying DML, researchers must choose two specific parameters in the algorithm. First, K is the number of folds into which the algorithm splits the data, using $K-1$ parts to train the ML model, and estimating the effects on the remaining part. Hence, $K = 1$ means no sample splitting, which does technically not fit the definition of DML in Chernozhukov et al. (2018). In their foundational paper, Chernozhukov et al. (2018) introduce the method with two folds, but recommend four or five folds, because larger numbers of folds allow the ML methods to train on larger samples. In the applications we observed, choices of two and three folds are popular, while the most frequent number of folds is 10 (Figure 2.1E). Again, this is likely due to the popular Stata implementation, which uses $K = 10$ as default.

The second parameter is the number of repetitions of the full DML algorithm in the applications. Chernozhukov et al. (2018) state that the random sample splitting can have an effect on the estimates in finite samples and thus recommend repeating the algorithm S times (they use $S = 100$) and reporting the median estimate across the repetitions. A majority of applications does not follow this advice (Figure 2.1F), which we can partly explain by the Stata default ($S = 1$), but also by the S-fold increase in computation time. However, some publications (8/36, 22%) use multiple repetitions. The largest number is 5,000 repetitions in Chan and Meunier (2022), which is feasible (and maybe necessary) because of their small sample size of 28 observations. Our results below will show that in smaller samples, using a larger number of repetitions can increase the robustness of the estimates considerably.

To sum up, in most of the applications we considered, researchers do not explicitly establish why their choice of specific ML algorithms, number of folds and number of repetitions is appropriate for their application. Therefore, our paper aims to provide guidance about the impact of these choices and potential trade-offs.

(3) Other papers have assessed the performance of DML when comparing different ML-based causal inference methods, albeit with a focus different from ours: They typically compare one (or a few) specific implementation(s) of DML to other novel or established methods, whereas we focus mostly on how different implementations of DML can perform in a large variety of settings. Here, we review these publications assessing the accuracy of DML and other methods when estimating causal effects (Table 2.1). These evaluations come from different disciplines: three (Loiseau et al., 2022; McConnell and Lindner, 2019; Zivich and Breskin, 2021) from healthcare/medicine, two (Gordon et al., 2022; Yang et al., 2020) from economics, and one (Qiu et al., 2022) from geoscience. Most evaluations consider the average treatment effect of a binary treatment and assess the methods on simulated data. Exceptions are Qiu et al. (2022), who work with a continuous treatment, and Gordon et al. (2022) as well as Loiseau et al. (2022), who evaluate the methods by comparing their estimates to results from randomized controlled trials (RCTs). In our study, we also evaluate the methods on simulated data, but focus on settings with a continuous treatment variable (e.g., medication dosage, temperature, price, etc.).

In the first evaluation that included DML, McConnell and Lindner (2019) assess how different ML-based causal methods perform when estimating average treatment effects. They use random forests in DML and show that, while DML easily dominates traditional methods, other methods (e.g., Bayesian Causal Forests) are slightly more accurate in several cases. Yang et al. (2020) compare tree-based methods in DML for three different functional forms of confounding and find that gradient boosting in DML is superior to regression trees and random forests across their simulation settings. Zivich and Breskin (2021) demonstrate the benefits of doubly robust methods and cross-fitting when estimating effects with ML. They find that methods like DML that use these techniques outperform singly-robust methods and methods without cross-fitting, but are computationally more expensive. In contrast to the previous studies, Gordon et al. (2022) assess whether DML (using neural networks) can recover experimental estimates of advertising effectiveness at Facebook. While DML delivers more accurate estimates than a naive comparison and an alternative observational method (stratified propensity score matching), it is still unable to recover the benchmark from the RCTs. The authors explain that Facebook does not record all relevant targeting data, which violates the unconfoundedness assumption in their DML application. Loiseau et al. (2022) also assess multiple methods by comparing their estimated effects to RCT estimates, in addition to a simulation study. Their findings suggest using DML for larger sample sizes, but G-computation when only having access to smaller datasets.

However, they only consider settings with few covariates and correctly specified models in their simulations, and use only relatively inflexible (regularized) linear methods for the predictive parts of DML. They call for further research on the performance benefits of using more flexible (e.g., tree-based) ML methods. Finally, Qiu et al. (2022) simulate data from a chemical transport model to assess the ability of various statistical approaches to adjust for meteorological variability when estimating effects of emission changes. In their setup, DML with random forests by far outperforms both traditional approaches and DML with lasso, but is not unbiased in all areas they consider.

Our study adds four contributions to these evaluations: (1) We extensively introduce and review the DML framework with a focus on intuitive understanding, (2) we apply DML to a much wider range of simulated settings to assess boundary conditions for the method, (3) we compare a variety of predictive methods within the DML framework, and (4) we provide guidance on decisions researchers face when applying DML in their field.

Table 2.1: Method evaluations considering DML

		ML methods										
Paper	**Treatment type**	**Linear regression**	**Lasso/Ridge**	**GAMs**	**Regression trees**	**Random forests**	**Boosting**	**Neural networks**	**Ensemble**	**Alternative methods**	**Evaluation method**	**Parameters**	**Findings/recommendations**
McConnell and Lindner (2019)	Binary					✓				OLS, AIPW, TMLE, BART, ps-BART, BCF, GRF	Simulations	Confounding strength, sample size (N), # of confounders (J), # of noise variables	DML slightly worse than BCF and ps-BART, but better than rest. DML is best if many covariates (>150).
Yang et al. (2020)	Binary				✓	✓	✓			-	Simulations	Various functional forms, # of confounders (J), sample size (N)	Gradient boosting in DML performs better than regression trees and random forests across DGPs. Bias increases in J, decreases in N.
Zivich and Breskin (2021)	Binary								✓	G-computation, IPW, AIPW, TMLE, CF-TMLE	Simulations modeled after medical application	Provide different model specifications to methods (true, linear, ML), # repetitions	Doubly robust methods with cross-fitting and ML outperform singly-robust and not cross-fit ML methods, but need more computing resources.
Gordon et al. (2022)	Binary							✓		Stratified PS-Matching (SPSM)	Comparison to RCTs from ad campaigns at Facebook	-	DML and SPSM unable to recover experimental benchmark. Both perform better than naive comparison. DML more accurate than SPSM. Reason: Not all relevant targeting data is logged and available for adjustment.
Loiseau et al. (2022)	Binary		✓							PS-matching, IPTW, G-computation	Simulation and comparison to RCTs of diabetes medication	Sample size, homogeneous vs. heterogeneous treatment effect	DML has smallest bias, G-computation smallest MSE. DML improves with sample size. Recommend G-computation for N<100, DML for N>500.
Qiu et al. (2022)	Continuous		✓			✓				OLS, polynomial regression, cubic splines, GAMs	Simulations from a chemical model for air quality effects of emission changes	-	DML with random forests performs best, but still not ideal in all areas. Other methods mostly perform poorly.
Our evaluation	Continuous	✓	✓	✓		✓	✓	✓		OLS, Naive XGBoost	Simulations	Various functional forms; confounding strength; # of confounders (J); sample size (N); inclusion of noise variables; inclusion of variables related to outcome; inclusion of variables related to treatment; violation of unconfoundedness; inclusion of bad controls	XGBoost, neural networks and random forests in DML perform best across a variety of settings. Choose between algorithms based on predictive accuracy in first stage. Decisions about causal structure and (parts of) variable selection not made automatically by DML but should be based on theory. Choose K based on sample size, test different S and choose number with stable results.

Note: OLS: Ordinary least squares, AIPW: Augmented inverse probability weighted estimator, TMLE: Targeted maximum likelihood estimator, BCF: Bayesian Causal Forests, BART: Bayesian additive regression trees, PS: Propensity score, GRF: Generalized random forests, CF: cross-fit, GAMs: Generalized additive models

2.3 Method review

2.3.1 DML in the partially linear model

In this section, we review double/debiased machine learning (Chernozhukov et al., 2018), aiming to provide an intuition about when, how and why the method works. One important setting in which researchers can apply DML to potentially relax assumptions is causal effect estimation in the presence of observed confounding (Figure 2.2).

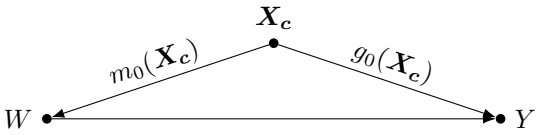


Figure 2.2: Directed acyclic graph (DAG) for the assumed causal structure. W: treatment variable, Y: outcome variable, $\boldsymbol{X_c}$: observed confounding variables. The relationships between $\boldsymbol{X_c}$ and W ($m_0()$), and $\boldsymbol{X_c}$ and Y ($g_0()$), are potentially complex and nonlinear.

In the assumed causal structure of Figure 2.2, we want to estimate the causal effect of a treatment variable W on an outcome variable Y. However, observed confounders $\boldsymbol{X_c}$ complicate the identification and estimation by influencing both the treatment and the outcome. If we do not adequately adjust for such confounders, the estimated causal effect is biased (e.g., Hernán and Robins, 2020, Chapter 7), i.e., different from the true value the estimator was supposed to provide (Wooldridge, 2012). We can express the data-generating process (DGP) of this setting with a partially linear regression (PLR) model, where V_y and V_w are noise terms:

$$Y = \beta W + g_0(\boldsymbol{X_c}) + V_y \tag{2.1}$$

$$W = m_0(\boldsymbol{X_c}) + V_w. \tag{2.2}$$

In the partially linear outcome model, we assume the treatment to have a linear, additive functional form, while the functional form of the confounders can be either linear or nonlinear. In this setting, the treatment has a constant, homogeneous effect on the outcome. We use the partially linear model for the introduction of DML, but DML is not limited to this setting. In case of a binary treatment, the treatment variable can arbitrarily interact with the confounders, which allows for the presence of heterogeneous treatment effects (see Section 2.6).

The assumptions encoded in Figure 2.2 as well as Equations 2.1 and 2.2 imply that we can identify the causal effect of W on Y after adjusting for observed confounders $\boldsymbol{X_c}$ (Hernán and Robins, 2020, Chapter 2). Different disciplines use different names for this assumption (with

slightly different technical definitions): unconfoundedness, exogeneity, ignorability, selection on observables (Imbens and Wooldridge, 2009), conditional independence assumption (Angrist and Pischke, 2009), exchangeability (Hernán and Robins, 2020), or a case for backdoor adjustment (Pearl et al., 2016). Researchers make this assumption in many studies across disciplines, hence it is highly relevant for applied research (Imbens and Wooldridge, 2009).

However, even if this assumption holds, two challenges appear. First, the functional relationships of the confounders with the treatment ($m_0(\boldsymbol{X_c})$) and the outcome ($g_0(\boldsymbol{X_c})$) may be nonlinear and complex. Secondly, there could be a large number of potential confounders in $\boldsymbol{X_c}$, such that it is not obvious which covariates one should include in the estimation process in finite samples. Such a high-dimensional setting can either arise naturally through many observed variables (e.g., covariates generated from text, images, or genes), or when including transformations of observed variables (e.g., polynomials, interactions, etc.) to allow for potential nonlinearities in the model (Belloni et al., 2014). Traditionally, researchers would make additional assumptions to address these challenges. For example, we could estimate the outcome model by adjusting linearly for confounding; in addition, it is common (Wooldridge, 2012, Chapter 9) to add a transformed (e.g., squared) variable to allow for some nonlinearities if theory suggests so (e.g., diminishing growth). This process is equivalent to the "parametric" assumption that the functional form is known and correctly stated by the researcher (Imbens, 2004). Also, researchers typically select the variables entering the estimation based on theory, domain knowledge, or intuition, which relies on assumptions as well. In some cases, all of these assumptions might be correct, but they are usually untestable, rather subjective or even arbitrary, and different assumptions might lead to very different causal estimates.

When facing these challenges – nonlinearities and variable selection problems – it is natural to think about using (supervised) ML. One of the well-known strengths of many ML methods is their ability to deal with high-dimensional data and to fit complex functional forms (e.g., Athey, 2019; Hastie et al., 2009; Mullainathan and Spiess, 2017). However, a "naive" application of ML methods for these causal questions leads to biased estimates, since the goal of traditional ML is prediction, which is fundamentally different from causal parameter estimation (Mullainathan and Spiess, 2017). The reason is that ML methods use regularization, which keeps predictive models from overfitting, but introduces a bias into parameter estimates (Chernozhukov et al., 2018).

At this point, the DML method comes in and suggests an estimation procedure which can use highly flexible, regularized ML methods while still providing an unbiased estimator for the causal effect. As a consequence, we can relax assumptions about functional forms and variable selection. Algorithm 1 depicts a version of the DML algorithm for the partially linear regression model (Chernozhukov et al., 2018). In other settings (e.g., binary treatment in the interactive model, instrumental variables, etc.), the algorithms and final estimators can be slightly different (see Section 2.6). In the first step of the algorithm, we split the full dataset randomly into K folds. Secondly, we hold out one of these folds, and use the remaining $K-1$ folds to train two ML models: The first model predicts the treatment W from the potential confounders $\boldsymbol{X_c}$; the second model uses the same variables to predict the outcome Y. In step three, we use these trained models to make predictions for treatment and outcome, respectively, on the data not used for training. Then, we subtract these predictions from the true values to obtain the residuals for treatment and outcome ($\hat{V}_W$ and $\hat{V}_Y$). Next, we linearly regress the residual of the outcome ($\hat{V}_Y$) on the residual of the treatment ($\hat{V}_W$) and obtain the coefficient for $\hat{V}_W$. Finally, we repeat steps 2-5 for each of the K folds, resulting in K different coefficients, which we finally average to obtain the final causal estimate.[1] Because the random splitting in the first step can potentially influence the estimation results, there is an additional step for more robustness in smaller samples: Chernozhukov et al. (2018) recommend repeating the full algorithm multiple (e.g., 100) times with different partitions, and finally reporting the median estimate across all splits.

Algorithm 1 DML algorithm for the partially linear regression model

1. Split the data into K folds
2. Train two machine learning models on $K-1$ folds:
   a) Outcome: W, features: $\boldsymbol{X_c}$
   b) Outcome: Y, features: $\boldsymbol{X_c}$
3. Use the models to make predictions ($\hat{W}$ and $\hat{Y}$) on the held-out fold
4. Compute residuals as $\hat{V}_W = W - \hat{W}$ and $\hat{V}_Y = Y - \hat{Y}$
5. Use a linear regression to estimate coefficient from residuals: Regress $\hat{V}_Y$ on $\hat{V}_W$, obtain the coefficient on $\hat{V}_W$
6. Repeat for all folds, average resulting coefficients to obtain the final causal estimate
7. For more robustness w.r.t. the random partitioning in finite samples:
   Repeat the algorithm S (e.g., 100) times for different splits, then report the median estimate

[1] Chernozhukov et al. (2018) name this algorithm "DML1". There is an alternative algorithm "DML2", which supposedly could perform better in small samples. Instead of computing a coefficient for each fold, DML2 collects the residuals across all folds and estimates the final effect on the full, residualized dataset. In our simulations, DML2 only outperformed DML1 in very rare cases with very large K and extremely small samples, therefore we continue with DML1, which we find more intuitive.

DML relies on two steps to obtain causal effect estimates. First, it estimates a model for both the outcome and the treatment. As a consequence, the final estimator is robust to minor mistakes in the estimation of either of these models, whereas traditionally, one would have to assume that the outcome model was correctly specified. The authors call this technique "orthogonalization", and it is closely related to the "double robustness" property of other estimators (e.g., Hernán and Robins, 2020, Chapter 13). Doubly robust estimators rely on estimating two models from the covariates: a treatment model and an outcome model. Combining the two models in the final estimator leads to the following robustness property: The estimator is consistent, as long as one of the two models is correctly specified. That is, even if one model is misspecified (wrong functional forms, missing variables), but the other one is correctly specified, the estimates will still be valid (Hernán and Robins, 2020, Chapter 13). The term orthogonalization then comes from the fact that in DML, we use the residuals for the final estimation. These residuals have been constructed to be orthogonal to the confounders (i.e., cannot be explained by the confounders). That is, by subtracting the predicted outcome and treatment values from their true values, we approximately remove the confounding influence in both variables. This estimator thereby builds on a type of double robustness in the sense that the errors of both models enter the overall estimation error as a product, such that if one error goes towards zero, the overall error does so as well. Thus, even though the regularization bias of using ML directly in the outcome model is too large to ignore, we can get closer to unbiased estimates if we also consider the treatment model and combine both models in a doubly robust way, as DML does. Please see Section 2 of Chernozhukov et al. (2018) for a formal derivation of this approach.

The second step is an efficient kind of sample splitting, which the authors call "cross-fitting". The goal of cross-fitting is to remove a bias caused by overfitting, which can occur if we use the same data to fit the ML models and to estimate the causal effect. Thus, in cross-fitting, we split the data into (at least) two samples: one for training the ML models, the other one for predicting, computing the residuals, and estimating the causal effect. This sample-splitting generally leads to a loss of efficiency, because we only use a part of the data in each step. However, cross-fitting regains efficiency by swapping the samples and repeating the procedure, such that we use each sample once for fitting the model and once for estimating the causal effect. This leads to multiple estimates (one for each sample) which we finally average.

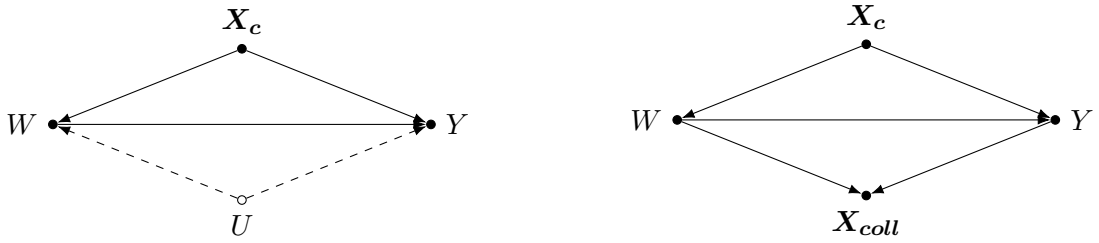
Using DML may allow researchers to relax variable selection and functional form assumptions. Traditionally, a researcher would assume a prespecified set of confounders with prespec-

ified (e.g., linear) functional forms. If these variable selection and functional form assumptions are violated, traditional methods only remove a part of the confounding variation, while the remaining confounding variation will still bias the estimated effect. However, if we employ ML algorithms to learn the important variables and the appropriate functional forms, we can rely on a weaker assumption: We do not necessarily need to know the correct variables and functional forms beforehand, we only need to assume that the ML algorithms perform reasonably well at the two prediction tasks. By virtue of the double robustness property, a "reasonably good" performance does not mean that the ML methods have to recover the true model perfectly; the method is insensitive to small mistakes in those models.[2] This assumption is not directly testable, but it is well aligned with the objective of flexible ML methods and thus – in most settings – weaker than the traditional assumptions, if we do not have very strong theory for a specific parametric model.

We emphasize that the DML algorithm alone does not by itself guarantee a causal interpretation of the estimates. There are important assumptions and decisions researchers need to make before applying the method or interpreting its results. Most importantly, the researcher still must assume a causal structure that implies unconfoundedness or another form of causal identification (e.g., instrumental variables). Typically, this assumption can be violated in two ways. First, there could be remaining unobserved or unmeasured variables that influence both treatment and outcome (Figure 2.3A) (Imbens and Wooldridge, 2009). DML cannot reduce the bias induced by such unobserved confounders. DML can flexibly adjust for *observed* confounders, but cannot handle confounders that are not observed. Second, researchers could unintentionally include a "bad control" in the algorithm (see also Hünermund et al., 2023). A bad control is any variable that we should not adjust for, because adjusting for it introduces a new bias (e.g., Angrist and Pischke, 2009; Cinelli et al., 2024). In the simplest case, a bad control can be a so-called "collider", which is influenced by treatment and outcome instead of influencing them (Figure 2.3B). In practice, realistic cases of bad controls might be more complex (see, e.g., Imbens, 2020; Pearl and Mackenzie, 2019). As a consequence of these two potential pitfalls,

[2]Technically, this means that we have to assume that the ML algorithms achieve $n^{1/4}$-consistency. Traditionally, if we assume a parametric model and it is correctly specified, we achieve $\sqrt{n}$-consistency. Loosely speaking, this means that, as the sample size n grows, the error between the true and the estimated value shrinks to zero by the factor $1/\sqrt{n}$, which is faster than the factor $n^{-1/4}$. Since the DML method makes use of the double robustness property, the errors of the treatment and outcome model multiply. Thus, in order to achieve $\sqrt{n}$-consistency without having to make the parametric assumptions, we must assume that the ML methods achieve $n^{1/4}$-consistency in both models ($n^{1/4} * n^{1/4} = n^{1/2} = \sqrt{n}$) (Chernozhukov et al., 2018; Belloni et al., 2017).

before applying DML, researchers have to consider (1) whether there might be any important unobserved confounders and (2) whether every variable included is a good control. Both of these assumptions are not directly testable from data, so the researcher has to argue from theory and domain knowledge about their plausibility.



(A) The unobserved confounder U biases the estimated effect.

(B) Adjusting for bad controls like colliders $\boldsymbol{X_{coll}}$ introduces a bias.

Figure 2.3: Possible violations of unconfoundedness

2.3.2 Possible ML algorithms

The underlying statistical theory of DML allows researchers to make use of a variety of different predictive or ML methods for the treatment and outcome model (Chernozhukov et al., 2018). The authors mention algorithms such as "random forests, lasso, ridge, deep neural nets, boosted trees, and various hybrids and ensembles of these methods". In our implementations of DML, we will focus on the following predictive methods for the ML tasks:

- linear regression, potentially with lasso regularization
- generalized additive models (GAMs)
- random forests
- boosted regression trees (XGBoost)
- neural networks.

These methods differ in three dimensions: the flexibility to model various functional forms, the ability to perform variable selection, and the degree to which they need large sample sizes to achieve flexibility.

First, tree-based methods like random forests and boosted trees, as well as neural networks with a nonlinear activation function, are inherently capable of fitting nonlinear functional forms, whereas linear regression and lasso are purely linear methods when applied to the original features. One can make the latter methods more flexible by transforming the features (e.g.,

with interactions and polynomials). However, these transformations lead to an increase in the numbers of parameters to estimate, which is problematic for linear regression (e.g., Mullainathan and Spiess, 2017). Also, researchers have to create the nonlinear transformations by hand with little theoretical guidance (Which variables should interact to which degree? Which order of polynomials is sufficient? Which further functional forms might be useful?). GAMs fit multiple smooth functions of the features simultaneously, which enables them to flexibly model a variety of functional forms (Hastie et al., 2009, Chapter 9). However, this comes at a cost because the fitting of multiple functions also increases the dimensionality, which can be problematic if the number of raw variables is large relative to sample size. Furthermore, GAMs are limited to additive relationships and smooth functional forms, and they can only model interactions if we add them manually (James et al., 2021, Chapter 7).

Second, lasso and tree-based methods like random forests and boosted trees can perform variable selection by default. That is, they will only make use of features that are predictive of the outcome, which enables them to work in high-dimensional settings with many variables or transformations. In contrast, linear regression, GAMs, and neural networks do not perform variable selection by default and thus fail in high-dimensional settings. There are approaches that enable a kind of variable selection in GAMs (e.g., Marra and Wood, 2011), but by default, the method only works in low-dimensional settings (Hastie et al., 2009, Chapter 9). Similarly, while neural networks do not select variables automatically, there are various other techniques to avoid overfitting, e.g., early stopping, weight decay, dropout, and L1-regularization (e.g., Hastie et al., 2009, Chapter 11).

Lastly, while random forests, boosted trees, and neural networks can very flexibly fit different functional forms, they might need relatively large sample sizes to do so (e.g., Hastie et al., 2009, Chapter 10). By contrast, larger sample sizes naturally cannot help linear methods to flexibly fit nonlinear functional forms.

In this paper, we focus on the selected predictive methods on grounds of their respective ease of implementation in statistical software and programming languages. For some of the methods, we can tune parameters to improve predictive accuracy. In our implementations, we tune at most one parameter (e.g., in lasso, random forests, XGBoost, and neural networks). Our main goal is not necessarily to find the model that minimizes test error for each method, but to compare how different methods perform in the DML framework without substantial researcher input in the ML step. We specify parameter and tuning choices in the simulation section. In

general, one can also consider further tuning of the methods, other algorithms, or ensembles of the mentioned methods, which might lead to further improvements in applications.

2.4 Simulations

In this section, we present and discuss the results from applying DML to a variety of simulated datasets. The goal is to assess the ability of the method to recover the true (simulated) causal effect in settings that differ with respect to the true data-generating process. First, we describe our implementation of DML, the different algorithms for the ML parts and the choice of parameters. Then, we begin the simulations with an arguably realistic baseline scenario. From there, we look for boundary conditions of the method by varying functional forms, numbers and kinds of variables, sample size, confounding strength, and causal structure. We generate 100 simulated datasets for each simulation setting and report how a variety of different methods perform across these 100 simulations.[3]

2.4.1 Method implementations

After having reviewed the theory behind DML in the previous section, we now present how we have implemented the different methods (Table 2.2) for our simulation study. We implement all methods in R (R Core Team, 2023).[4] As a benchmark for DML, we include three alternative methods. First, we estimate a "Simple" OLS regression of the outcome on the treatment, completely ignoring any other variables. If confounding is present in the data, this method will lead to biased results, since it does not adjust for confounding at all. Secondly, we estimate another OLS regression, but this time adjust linearly for all observed covariates. If the true confounding relationships are linear and low-dimensional, this will adequately adjust for confounding (see, e.g., Imbens and Rubin, 2015). However, this approach cannot model any nonlinear relationships by default. Adding interactions and squared terms for each covariate manually could help account for some nonlinearities, but also increases the number of variables in the model. Lastly, we implement one additional benchmark method that "naively" uses ML (as described in the introduction of Chernozhukov et al. (2018)), by only modeling the outcome model $g_0(\boldsymbol{X_c})$, nei-

[3] We experimented with 50 to 1,000 simulation replications and found 100 to be fairly representative for the distribution of resulting coefficients, while still being relatively economical with computing resources. We show results for different numbers of replications in our baseline scenario in Appendix A.3 (Figure A.1).

[4] All our code is available on OSF: `https://osf.io/eswfk/?view_only=0fb868c0d78e4550bf0a2ab5a71ce7b1`.

ther relying on orthogonalization nor cross-fitting. To do so, we first obtain an estimate of β by linearly regressing Y on W. Then, we get an estimator $\hat{g}_0$ by regressing $Y - \hat{\beta}W$ on $\boldsymbol{X_c}$ using an ML method. Lastly, we use this estimator to predict the original outcome Y, compute the outcome residual and regress the residual on the treatment W to obtain an effect estimate. We implement this naive ML method with XGBoost as the ML algorithm, selecting the parameters as described below. The results of other ML methods in this naive framework are very similar, so we focus on the version with XGBoost as representative.

Table 2.2: Description of implemented methods

Label	**Description**
Simple OLS	Ordinary least squares estimation of a linear regression model without any covariates
OLS	OLS estimation of a linear regression model with all covariates included linearly
XGBoost (naive)	XGBoost without orthogonalization and cross-fitting
OLS (DML)	DML with OLS regression as predictive algorithm
Lasso (DML)	DML with lasso as predictive algorithm
GAMs (DML)	DML with GAMs as predictive algorithm
Random forests (DML)	DML with random forests regression as predictive algorithm
XGBoost (DML)	DML with XGBoost as predictive algorithm
Neural nets (DML)	DML with neural networks as predictive algorithm

We implement all versions of DML as described in Algorithm 1 (DML1 of Chernozhukov et al. (2018)). As defaults, we use $K = 5$ folds for the cross-fitting procedure and repeat the full algorithm nine times for different sample splits ($S = 9$), from which we take the median estimate. The authors recommend $K = 5$; and $S = 9$ is a number of repetitions that leads to stable estimates in our simulations. We vary these parameters in later settings. For the ML parts of the algorithm, we use different predictive methods: OLS regression, lasso, generalized additive models (GAMs), random forests, boosted trees (as implemented by XGBoost), and neural networks:

1. We implement lasso with the *glmnet* package by Simon et al. (2011). We determine the regularization parameter λ via 10-fold cross-validation and choose it to minimize the cross-validation mean squared error on the $K - 1$ folds used for training.

2. For GAMs, we use the *mgcv* package to model each continuous covariate with a smooth nonlinear function, estimated with restricted maximum likelihood ("REML") (Wood, 2017).

3. We implement random forests with the *randomForest* package (Liaw and Wiener, 2002), train using 200 trees and tune *mtry* (the number of variables randomly sampled from at each split) with respect to out-of-bag error.

4. For boosted trees, we utilize the implementation by the *xgboost* package (Chen et al., 2023). For the learning rate (eta) and the maximum tree depth, we use default values (0.3 and 6, respectively). Our implementation uses early stopping if the validation set performance does not improve for 10 rounds. We determine the optimal maximum number of boosting iterations by 5-fold cross-validation from up to 200 rounds and use it subsequently in the final model.

5. Finally, we implement neural networks using the computationally very efficient *nnet* package (Venables and Ripley, 2002). We use four units in one hidden layer with a sigmoid activation function and tune the parameter for weight decay between 0 and 5. Tuning the number of units and fixing the decay did not lead to better results. Tuning both the number of units and the parameter for the weight decay further improved the results, but led to an unreasonable increase in computational cost on CPUs (more than 7 times longer in our baseline than tuning one parameter or using DML with tuned random forests). In practice, if the computational resources are available, one can allow for more hidden layers and units, a different activation function, regularization techniques, and tune multiple parameters to achieve a good predictive fit. For our purpose, the above approach seems to achieve a good trade-off between predictive accuracy and computational cost, which makes the results more comparable to the alternative methods.

2.4.2 Baseline simulation

We assume the causal structure from Figure 2.2 for most simulations settings. That is, we can estimate the causal effect by adequately adjusting for all observed confounders ("unconfoundedness"). In later settings, we examine the consequences of a misspecified causal structure, i.e., we mistakenly leave out important confounders or adjust for bad controls.

In all following simulations, we draw exogenous variables from a multivariate normal distribution with a mean of zero and a randomly generated covariance matrix (i.e.,., $\boldsymbol{X_c} \sim N(0, \Sigma)$, $\Sigma = A'A$, with $A \sim N(0, 0.5)$). The true causal coefficient of interest is 1 ($\beta = 1$). We draw intercepts and noise terms from a standard normal distribution ($\alpha, \epsilon \sim N(0, 1)$), which leads to rather high-signal settings with an average R^2 around 80% for the correct parametric model in

the main simulations. In our baseline scenario, we construct treatment W and outcome Y based on Equations 2.3 and 2.4. We use functional forms with some complexity that could plausibly occur in nature: a linear form, higher-order polynomials, an interaction, and a step-function:

$$W = \alpha_0 + \delta_1 X_1 + \delta_2 {X_2}^2 + \delta_3 X_1 X_2 + \delta_4 step(X_3) + \delta_5 {X_4}^3 + \epsilon_0 \tag{2.3}$$

$$Y = \alpha_1 + \beta W + \gamma_1 X_1 + \gamma_2 {X_2}^2 + \gamma_3 X_1 X_2 + \gamma_4 step(X_3) + \gamma_5 {X_4}^3 + \epsilon_1. \tag{2.4}$$

Thus, the variables X_1 to X_4 are confounders that one should adjust for. We set the confounding strength $\delta_j = \gamma_j = 0.1$. $step()$ is the implementation of a step function, where we draw the initial variable from a standard normal distribution, before assigning each value to one of four steps between -1 and 1, depending on the quartile the value belongs to.

The results of our baseline simulation demonstrate the ability of DML with flexible ML methods to appropriately adjust for the various confounding influences (Figure 2.4). As expected, the

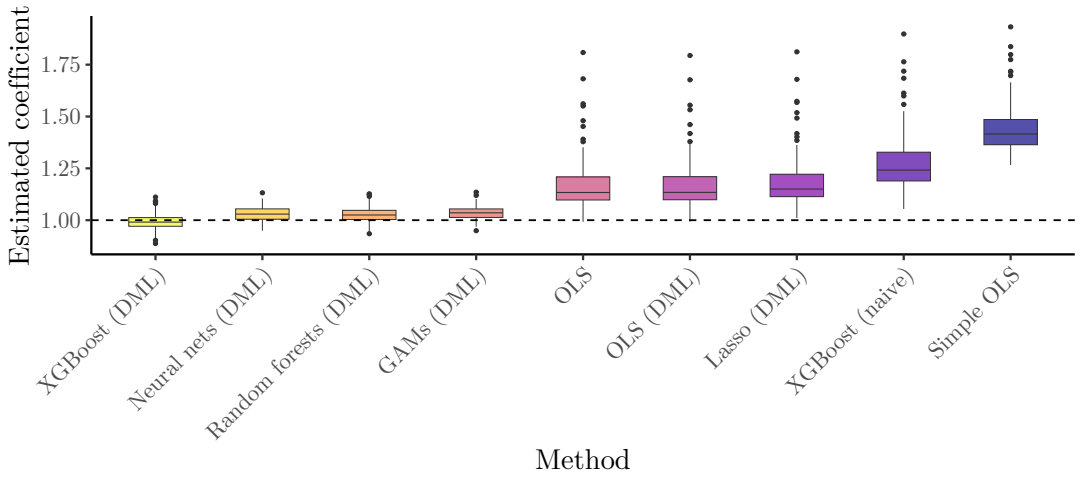


Figure 2.4: Results for our baseline simulation with sample size $N = 1,000$. The horizontal axis displays the different methods from Table 2.2. The vertical axis depicts the estimated coefficient. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method.

naive simple OLS regression performs worst due to its inability to adjust for confounding in any way. The naive ML method is able to adjust for some confounding, but still incurs significant bias due to regularization bias and overfitting. This is evidence for the notion that we should not use simple ML tools for causal inference. The traditional OLS regression, adjusting for all covariates linearly, outperforms the naive ML method, but is still biased.

Within the DML framework, the linear methods perform very similarly compared to the standard OLS regression. By contrast, the inherently flexible ML methods are most successful

at adjusting for the observed confounding. The median estimates of DML with GAMs, random forests, neural networks, and XGBoost are only off by about 3%, 2%, 2%, and 1%, respectively. These methods seem to be able to learn the functional form of the confounding from the data, without the need to include transformed covariates.

2.4.3 Extended simulations

In the following, we extend our simulations to eleven further scenarios to assess DML's capabilities and limitations (Table 2.3).

Table 2.3: Description of simulation scenarios

Case	Category	Data-generating process	Goal
1	Functional form	Confounding influence can take on different functional forms: linear, u-shaped, interactions, cubic, step function, or random	Examine performance of methods for various functional forms that differ in the degree of nonlinearity, smoothness, and additivity
2	Confounding strength	Multiplying the confounding coefficients by larger integers	Assess impact of stronger confounding on estimates of methods
3	Confounding dimension	Varying numbers of confounders	Assess robustness of methods to larger numbers of important confounders
4	Sample size	Varying the sample size	Examine data size requirements of methods
5	Inclusion of further variables	Including varying numbers of noise variables	Assess robustness of methods to the inclusion of irrelevant variables
6		Including variables only related to outcome	Learn whether inclusion of these variables is beneficial or harmful
7		Including variables only related to treatment	Learn whether inclusion of these variables is beneficial or harmful
8	Violations of identification	One unobserved confounder in addition to observed confounders	Assess robustness of methods to unobserved confounding
9		Covariates are colliders instead of confounders	Examine consequences of misclassifying variables as good controls
10	Parameters in DML	Varying the number of folds K into which we split the sample	Examine impact of choosing K and distributing observations between prediction and estimation
11		Varying the number of algorithm repetitions S	Examine additional robustness achieved by repeating DML

For the first five cases, we slightly deviate from the basic setup of the baseline's DGP. Instead of keeping all confounding coefficients fixed to 0.1, we randomly draw them from a standard normal distribution ($\delta_j, \gamma_j \sim N(0,1)$). We constrain δ_j and γ_j to have the same sign to rule out that confounding effects of multiple confounders cancel out. These new coefficients make the adjustment more challenging, both because of their higher average magnitude and because of the randomness of their magnitude. Similarly to Belloni et al. (2014), Chiang et al. (2022), and McConnell and Lindner (2019), the confounding coefficients decrease with each additional confounder. We achieve this by multiplying each confounding coefficient with $1/j$, for $j = 1, ..., J$ confounding variables. Finally, for all of the following simulation settings, we do not present

results from using DML with OLS regression as predictive method ("OLS (DML)"), since the results of this method are virtually equivalent to directly estimating an OLS regression ("OLS").

Case 1

In Case 1, we assess how well the different methods can adjust for different functional forms of the confounding influences $g_0()$ and $m_0()$. We assume that $g_0()$ and $m_0()$ have the same functional form (although with potentially different coefficients).[5] The sample size is still $N = 1,000$ and there are five confounders with the specified functional form. Table 2.4 shows the equations that generate the various functional forms. The first functional form of the confounding is linear. Secondly, the confounders enter the model by a squared term and thus have an u-shaped influence. This nonlinear functional form commonly occurs in nature in the form of diminishing growth, e.g., the relationship between the amount of fertilizer and crop yield. Third, if there is more than one confounder, they can interact with each other, e.g., in pairs of two as we implement here. For each confounder, we randomly draw the identity of the other confounder it will interact with. This is the first non-additive and non-smooth functional form. Fourth, the influence of the confounders follows a non-smooth step function. This is challenging for methods only modeling smooth functions, but favors tree-based methods. To implement the step function, we first compute the quartiles for each confounder. Then, we transform the confounder by assigning to each original value a value from [-3, -1, 1, 3], depending on the quartile to which the original value belongs. The result is a monotonically increasing step function consisting of three steps. This function can occur when decision makers base their decisions on heuristics: For example, when setting prices, a product manager might consider thresholds rather than a fully linear model of competitor prices. Only if a competitor price exceeds a certain threshold, the manager adjusts the own-price accordingly. Fifth, the confounding influence follows a cubic functional form. Cubic functions can occur in economics, when marginal costs fall with increasing output but rise again at some point (e.g., Beach, 1949). Since the cubic term leads to a very strong confounding influence, we scale it down by a factor of 0.25 to make it more comparable to the other functional forms. Sixth, we draw the functional form randomly for each confounder.

[5]This assumption is not necessary in DML, but it helps us to evaluate which methods can adequately model the respective functional form. If the same confounder has a different functional form in the treatment model than in the outcome model, the resulting coefficient of most DML methods in our simulations is between the ones for the respective functional forms (see Figure 2.5). For OLS, the estimates are unbiased if either the outcome *or* the treatment model is correctly specified (e.g., linear).

We select from each of the first five forms with equal probability. This case is most similar to the baseline scenario, though more challenging because of the randomness in the confounding coefficients.

Table 2.4: The different options for the functional form of the confounding

Label	**Equation**
Linear	$g(X_c) = m(X_c) = X_c$
U-shaped	$g(X_c) = m(X_c) = {X_c}^2$
Interactions	$g(X_c) = m(X_c) = \sum_{j=1}^{J} X_{c_j} X_{c_k} \quad \text{with} \quad k \in_R \{\{1, ..., J\} \setminus j\}$
Step	$g(X_c) = m(X_c) = \begin{cases} -3 & \text{if} \quad X_c < Q_1 \\ -1 & \text{if} \quad Q_1 \leq X_c < Q_2 \\ 1 & \text{if} \quad Q_2 \leq X_c < Q_3 \\ 3 & \text{if} \quad Q_3 \leq X_c \leq X_{c_{max}} \end{cases}$
Cubic	$g(X_c) = m(X_c) = 0.25 * {X_c}^3$
Random per confounder	For each confounder, select from above with $p = .2$ each $(g_0(X_{c_j}) = m_0(X_{c_j}))$.

The performance of the different methods varies significantly across the different functional forms (Figure 2.5). Since every simulation contains observed confounding, the method not adjusting for confounders at all ("Simple OLS") performs worst across scenarios, followed by the naive ML method. These methods' estimates vary widely across the 100 simulations, and their median displays a severe bias. When the confounding influence is linear (Figure 2.5A), all DML methods and the OLS regression perform well. DML with XGBoost is marginally biased in the direction opposite to the confounding, which potentially indicates minor overfitting.

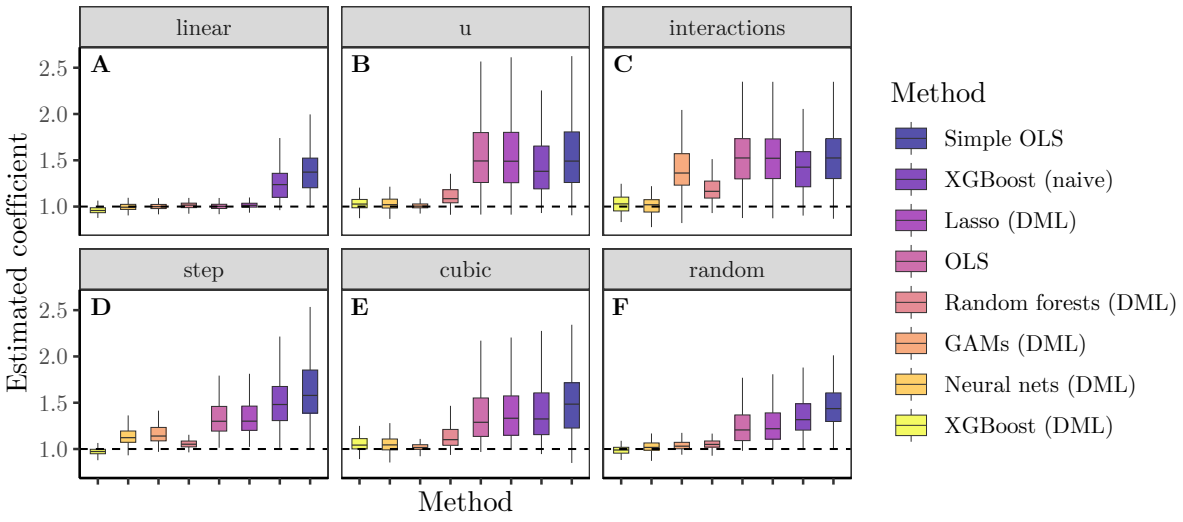


Figure 2.5: Results for Case 1 - distribution of estimated coefficients for each method across 100 simulations by functional form (outliers not displayed). The dashed line marks the true causal effect ($\beta = 1$). **A** Linear confounding. **B** U-shaped/squared confounding. **C** Pairwise interactions between confounders. **D** Confounding via step function. **E** Cubic confounding. **F** Confounding functional form drawn randomly for each confounder.

In case of a u-shaped confounding influence (Figure 2.5B), the linear methods are unable to adjust for any confounding, thus providing results with bias similar to the simple OLS regression. The results demonstrate that although we can classify lasso as a ML method, it does not have any more flexibility than a classical OLS regression when it comes to fitting nonlinear functional forms from raw, untransformed variables. In this setting, DML with GAMs performs best, easily fitting the smooth quadratic function. DML with neural networks and DML with XGBoost have a slightly wider distribution of estimates, but are also close to unbiased at the median. DML with random forests is more biased, but still easily dominates the linear methods.

Confounding through interactions (Figure 2.5C) is a non-smooth, non-additive functional form, hence purely linear methods are again unable to adjust for such confounding. GAMs in DML achieve more accurate results, but are still heavily biased. Both tree-based methods adjust better for interactions, although DML with XGBoost once more incurs the smaller bias. DML with neural networks most accurately estimates the effect and delivers unbiased estimates, slightly better than DML with XGBoost.

Next, for the step function, linear methods are superior to not adjusting for confounding at all (Figure 2.5D). This is because we constructed the step function to increase monotonically. DML with GAMs gets even closer to the true coefficient, but cannot fit the steps well with smooth functions. Here, DML with neural networks is more accurate than DML with GAMs, but still incurs some bias and cannot compete with the tree-based methods. DML with random forests adjusts for confounding more successfully, and DML with XGBoost performs best, delivering estimates that are close to unbiased.

While the linear methods only adjust for small parts of the challenging, but smooth cubic confounding influence, DML with GAMs adjusts almost perfectly (Figure 2.5E). The tree-based methods and DML with and neural networks eliminate much confounding, but not as reliably as GAMs within DML.

If we draw the functional form randomly with equal probability for each confounder, the naive methods again perform worst, followed by the purely linear models (Figure 2.5F). DML with random forests, GAMs, neural networks, and XGBoost all deliver mostly accurate estimates, with DML with neural networks or XGBoost being closest to the true value most often. This result is very similar to our baseline simulation. Because the "random" functional form is most similar to the baseline simulation, we use it in the simulations for Cases 2-5 and 10-11, where we vary other characteristics of the data-generating process.

Cases 2-4

We visualize the impact of different characteristics in Cases 2-4 (and in Case 5) in terms of the mean absolute error (MAE) of the estimated coefficient. We chose the MAE over the MSE since it allows for a more direct interpretation as the average absolute deviation from the true effect. In Case 2, we vary the strength of the confounding influence by multiplying the confounding coefficient of the outcome (γ_j) with different positive integers. This increases the confounding bias and thus amplifies the importance of adequate adjustment. Increasing the strength of the confounding leads to a strongly increasing bias for methods that do not adjust well (Figure 2.6). In fact, the MAE for all methods increases almost linearly, which is especially problematic for methods that are strongly biased in the first place. From this observation, we conclude that a larger confounding strength does not necessarily make the adjustment itself more difficult, but scales up the impact of the remaining confounding variation for all methods. The larger the confounding strength, the more important it is to adequately adjust for the confounding.

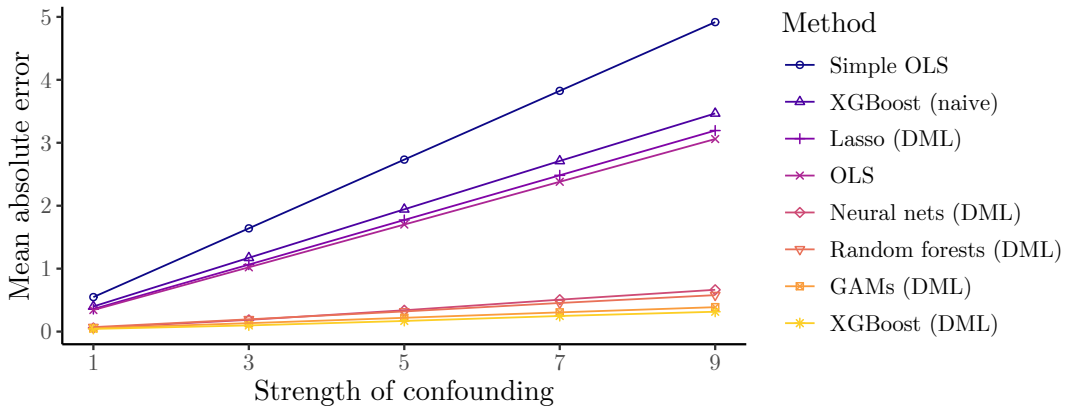


Figure 2.6: Results for Case 2 - mean absolute error in estimated coefficient across 100 simulations by varying confounding strength.

In Case 3, we investigate how the number of confounding variables complicates the adjustment. All methods become less accurate as the number of confounders increases (Figure 2.7). The simple OLS regression remains worst, while the bias of the naive ML method and the linear methods becomes more similar as the number of confounders increase. The flexible DML methods are all reasonably accurate up to a number of five confounders. After that, their performance deteriorates, although at different paces. DML with random forests appears most sensitive to the inclusion of more confounders; at 20 confounders, it is slightly less biased than the linear methods; with 50 confounders, its estimates are essentially equally or more biased.

DML with GAMs, XGBoost, or neural networks also incurs severe bias for very large numbers of confounders, but remain significantly more accurate than the alternative methods. Interestingly, DML with neural networks seems less affected by the inclusion of more than 20 confounders: For 50 confounders, the bias even somewhat decreases. These results demonstrate that the capability of DML to adjust for confounding in high-dimensional settings does not refer to the situation with many important raw confounders relative to sample size. Rather, DML can successfully adjust for complex functional forms, which it can model with a large number of parameters within flexible ML methods.

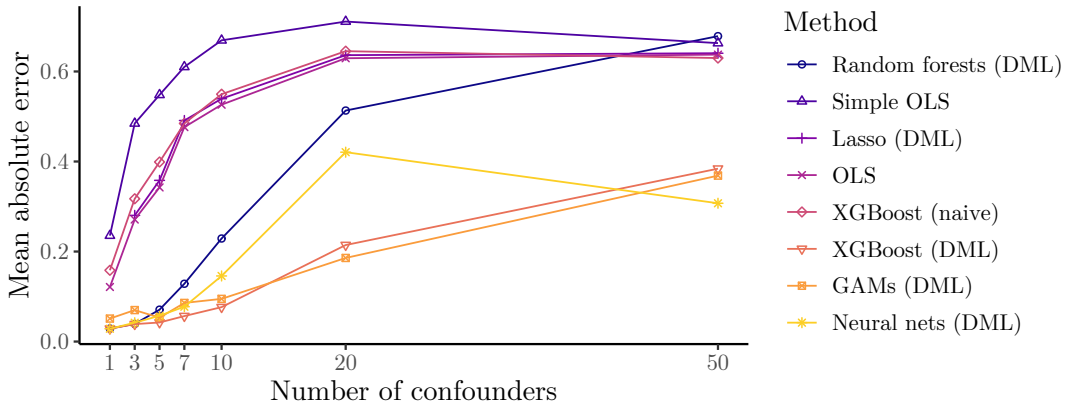


Figure 2.7: Results for Case 3 - mean absolute error in estimated coefficient across 100 simulations by varying numbers of confounders.

While many ML algorithms are capable of flexibly learning functional relationships from data, most have certain functional forms they can model very easily, while they might need many more observations to learn others. For example, tree-based methods are good at modeling step-functions and interactions, but need more data points to model smooth or linear functions (see, e.g., James et al., 2021, Chapter 8). Therefore, in Case 4, we vary the size of the simulated dataset. When the confounding influence has a random functional form (Figure 2.8A), the simple OLS regression and the linear methods do not benefit from additional observations, since they are unable to learn nonlinear functional forms. By contrast, the naive ML method improves with larger sample size, but the improvement is far too slow to be substantial. For the flexible DML methods, the MAE decreases significantly with sample size, with the biggest improvements occurring between 20 and 500 observations. After this point, DML with GAMs does not improve further, whereas DML with random forests, XGBoost, or neural networks continues to learn and remove more of the confounding influence. In the case of linear confounding (Figure 2.8B), we make similar observations. While the simple OLS regression does not benefit

from additional observations, the naive ML method does, but again, the improvement is far too slow to realistically approach the true effect. Since the confounding is linear, the linear methods and DML with GAMs perform well at any sample size. DML with random forests and neural networks have slightly higher bias in very small samples, but they quickly learn the linear relationships as the sample size grows, ending up with estimates that are virtually indistinguishable from linear methods in large samples. Even in very small samples ($N = 20$), OLS does not substantially outperform the flexible DML methods. These results present a strong argument for using flexible DML methods over OLS, as they perform considerably better if the OLS model is misspecified, and not substantially worse in cases where the OLS model is correctly specified.

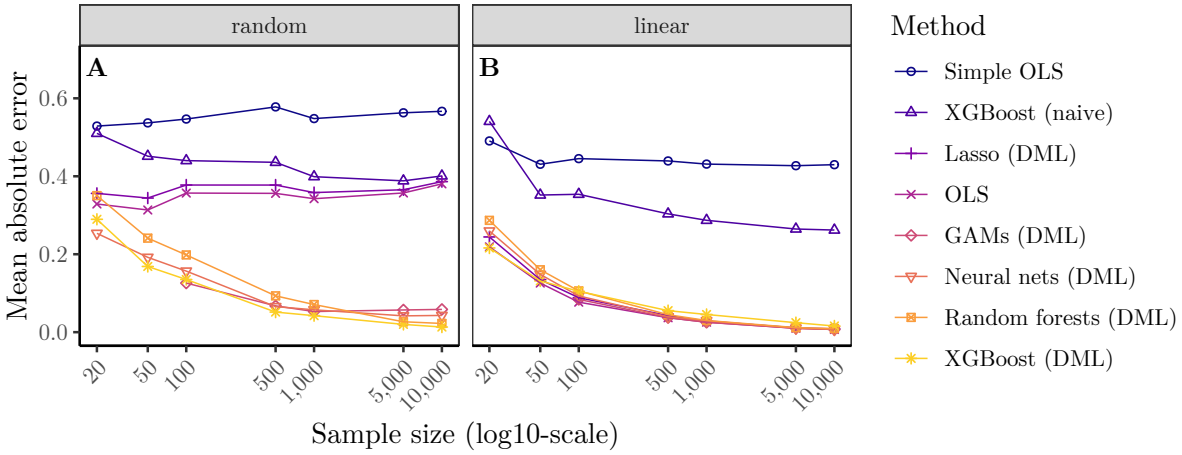


Figure 2.8: Results for Case 4 - mean absolute error in estimated coefficient across 100 simulations by varying sample size. Sample size displayed in log10-scale. **A** Confounding with random functional forms. **B** Linear confounding.

Cases 5-7

So far, we have only considered scenarios where all covariates are confounders. However, there could also be noise variables $\boldsymbol{E}$, variables only influencing the outcome $\boldsymbol{X_p}$, and variables only influencing the treatment $\boldsymbol{X_z}$ (Figure 2.9). In Cases 5-7, we explore the relevance of these variables for the performance of DML. First, in Case 5, we expect that methods capable of variable selection are less influenced by the inclusion of noise variables than others. Also, at least asymptotically, $\boldsymbol{X_p}$ and $\boldsymbol{X_z}$ should not impact the overall bias. However, in finite samples, adequately adjusting for $\boldsymbol{X_p}$ can increase precision (and decrease standard errors), while adjusting for $\boldsymbol{X_z}$ can hurt precision (and increase standard errors) (Cinelli et al., 2024).

The inclusion of noise variables (Figure 2.10) affects none of the methods severely, except for DML with neural networks. The high flexibility of neural networks, combined with their inability

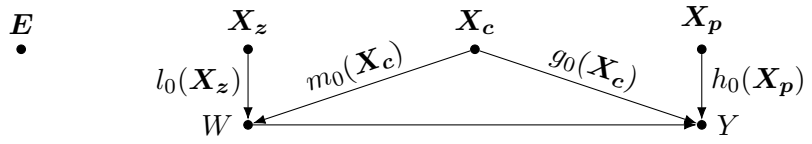


Figure 2.9: DGP including noise variables $\boldsymbol{E}$, variables only influencing the treatment ($\boldsymbol{X_z}$), and variables only influencing the outcome ($\boldsymbol{X_p}$). The influence of $\boldsymbol{X_z}$ and $\boldsymbol{X_p}$ is potentially also complex and nonlinear ($l_0()$, $h_0()$).

for variable selection in our implementation, likely causes them to overfit the noise variables (which could potentially be avoided by further tuning and regularization). Even though some other methods also do not perform variable selection, they can still handle moderate numbers of noise variables. An OLS regression, for example, simply estimates near-zero coefficients for all noise variables. The main problem with high dimensionality arises when we try to make linear methods similarly flexible as, for example, ML methods like random forests, XGBoost, or neural networks. Including various transformations of all variables (e.g., interactions and polynomials) could become problematic for OLS if the number of parameters approaches or exceeds the number of observations.

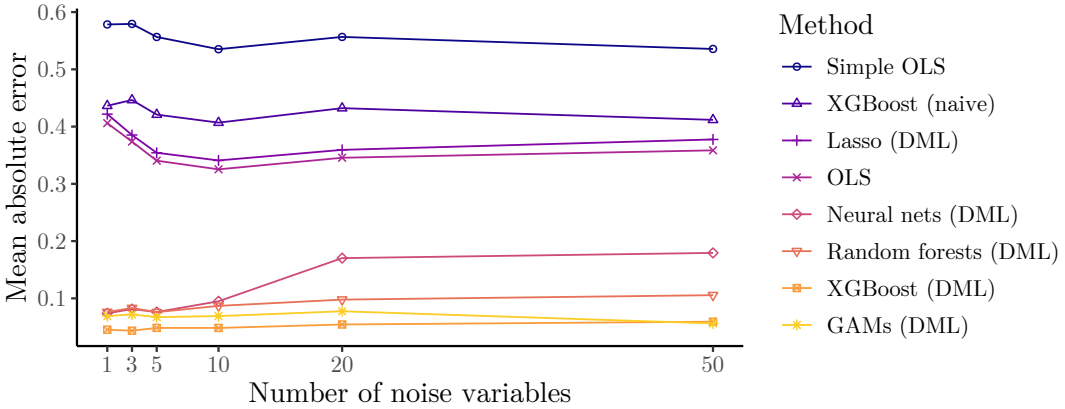


Figure 2.10: Results for Case 5 - mean absolute error in estimated coefficient across 100 simulations by varying number of noise variables.

For the remaining cases, we return to the general setup of the baseline scenario, but change the nature of the variables we use. In Case 6, we introduce four additional variables ($\boldsymbol{X_p}$) which only influence the outcome. They take on the same functional forms as the confounders, but do not influence the treatment. We then estimate two models for each method: one adjusting for all variables ($\boldsymbol{X_c}$ and $\boldsymbol{X_p}$), the other only adjusting for the confounders ($\boldsymbol{X_c}$).

For most methods, adjusting for $\boldsymbol{X_p}$ in addition to $\boldsymbol{X_c}$ has little impact on the median coefficient estimate (Figure 2.11A). However, the boxplots tend to have a narrower distribution

for the models that try to adjust for all variables. This in line with theory: By adjusting for $\boldsymbol{X_p}$, we reduce the variance in the outcome, which leads to more precise estimates (Cinelli et al., 2024). The distribution of the standard errors reinforces this finding: When adjusting for $\boldsymbol{X_p}$, the standard errors of all methods become significantly smaller (Figure 2.11B). From these results, we can derive the recommendation that researchers should adjust for all pre-treatment variables influencing the outcome, even if theory suggests that they are not related to the treatment.

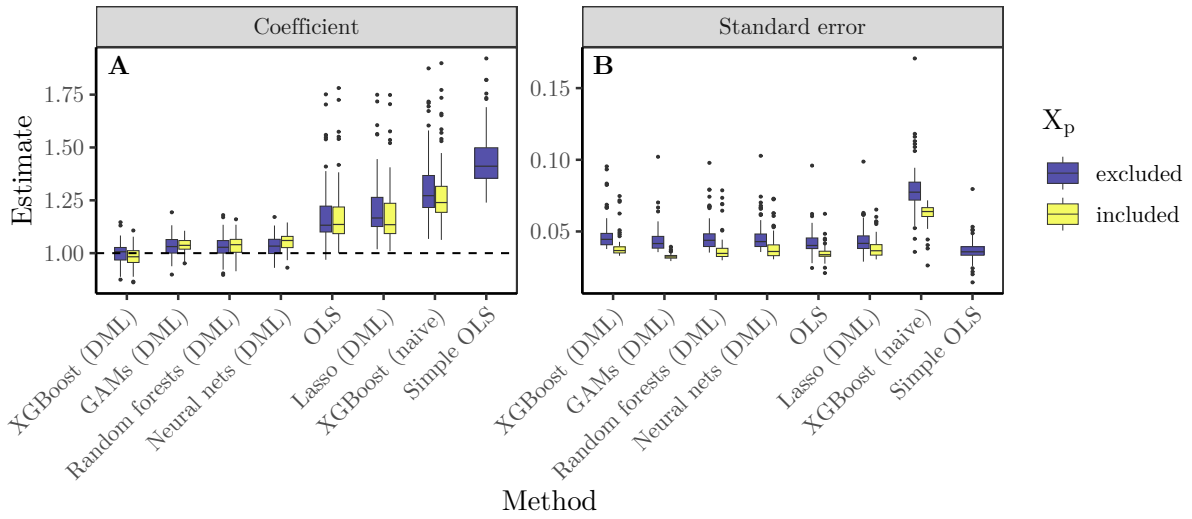


Figure 2.11: Results for Case 6 - distribution of **A** coefficient estimates and **B** standard error estimates across 100 simulations including (yellow) and excluding (blue) variables only related to the outcome. The dashed line marks the true causal effect ($\beta = 1$). Smaller standard errors are preferable.

Different from the previous case, in Case 7, the additional covariates ($\boldsymbol{X_z}$) influence only the treatment instead of only the outcome. These variables again take on the same functional forms, and we once more estimate models adjusting and not adjusting for them.

The results stand in contrast to the previous case: Now, the models that adjust for all variables, including $\boldsymbol{X_z}$, become less precise compared to the models that only adjust for confounders $\boldsymbol{X_c}$ (Figure 2.12A). In our simulations, this leads to both a more biased median and a wider distribution of estimates. Also, for most methods, including the variables related to the treatment leads to considerably larger standard errors (Figure 2.12B). Including these variables reduces the variance in the treatment, leaving less exogenous variation to estimate the effect (Cinelli et al., 2024). Our results demonstrate that in finite samples, this can even lead to bias in the effect estimates. The bias should vanish asymptotically, but the loss of precision is sufficient reason to avoid including these variables in any models.

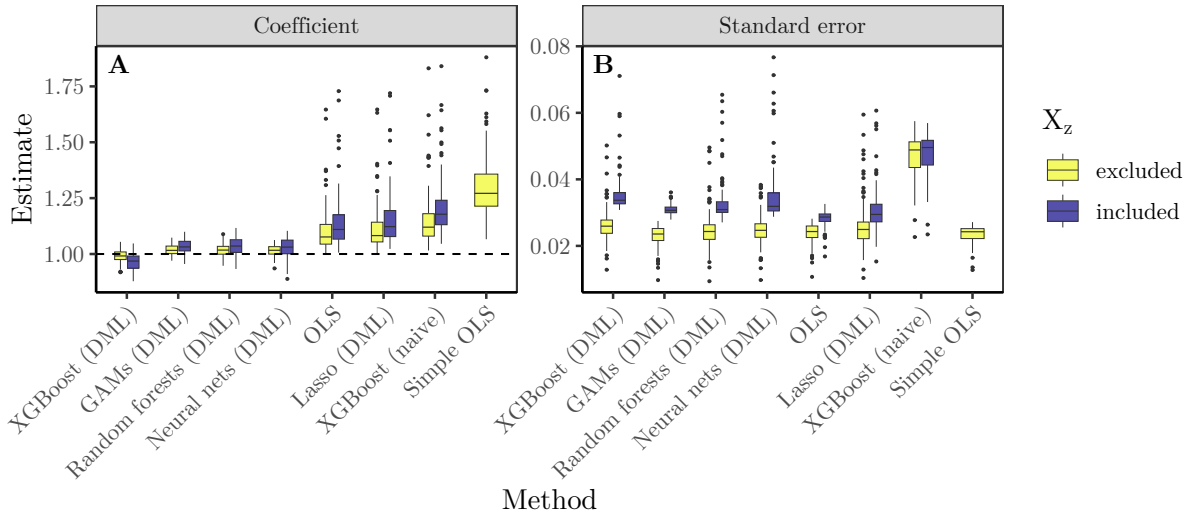


Figure 2.12: Results for Case 7 - distribution of **A** coefficient estimates and **B** standard error estimates across 100 simulations including (blue) and excluding (yellow) variables only related to the treatment. The dashed line marks the true causal effect ($\beta = 1$). Smaller standard errors are preferable.

Cases 8-9

As the final modifications to our DGP, we demonstrate in two different scenarios how violations to the assumed causal structure impact the estimates. So far in our simulations, assuming unconfoundedness, homogeneous treatment effects, and adjusting for all observed covariates was a valid identification strategy. Now, in Case 8, we include an additional confounder U in the DGP from Equations 2.3 and 2.4, as visualized earlier in Figure 2.3A. Its coefficient is relatively large ($\delta_6 = 0.5$), but this confounder is not observed, hence we cannot adjust for it. The unobserved confounding affects all methods and makes them unable to recover the true causal effect (Figure 2.13). Since U is not observed, no method can adjust for it, irrespective of

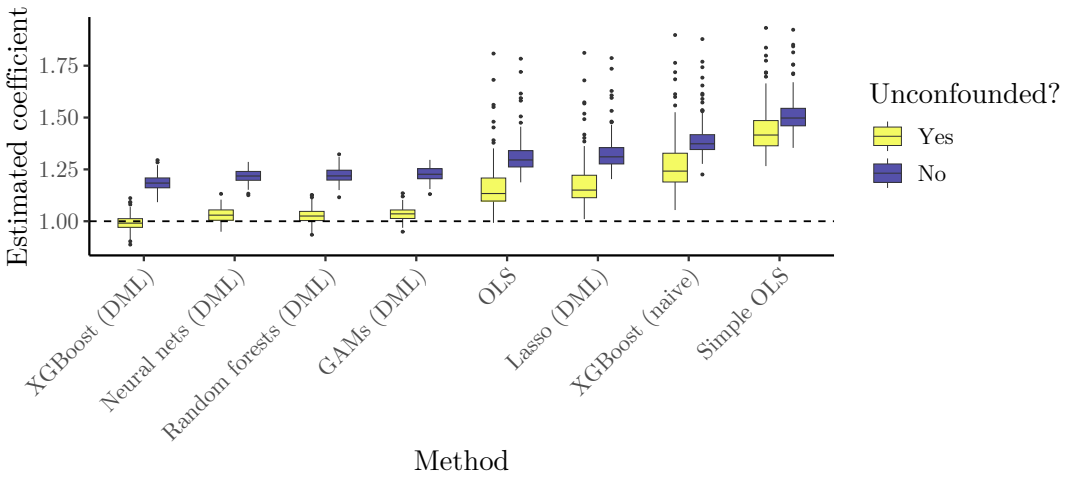


Figure 2.13: Results for Case 8 - distribution of estimated coefficients across 100 simulations without (yellow) or with (blue) unobserved confounding. The dashed line marks the true causal effect ($\beta = 1$).

the method's flexibility. However, even though the estimates of all methods are biased, the bias is less pronounced for the flexible DML methods. This is not because they adjust for some part of the unobserved confounder, but because they still perform well at adjusting for the observed confounders. So, even though unobserved confounding affects all methods, flexible DML might still *relatively* outperform less flexible methods and lead to closer-to-truth estimates. This holds under the assumption that the unobserved confounding biases the estimate in the same direction as the observed confounding, otherwise the biases could cancel each other out. The plausible direction of the confounding can often be established from theory and domain knowledge for a specific application.

A second violation of our identification strategy occurs when we adjust for a bad control, that is, a covariate that we should not adjust for. In the simplest case, a bad control is a collider variable as visualized in Figure 2.3B. To demonstrate the importance of the distinction between good and bad controls, we implement Case 9 such that all observed covariates are colliders instead of confounders. This means that there are five variables in $\boldsymbol{X_{coll}}$, caused by treatment and outcome with the same functional forms we used in the baseline. The simulation results confirm the theoretical statement that adjusting for bad controls leads to biased estimates (Figure 2.14). The only unbiased method is the simple OLS regression which does not adjust for

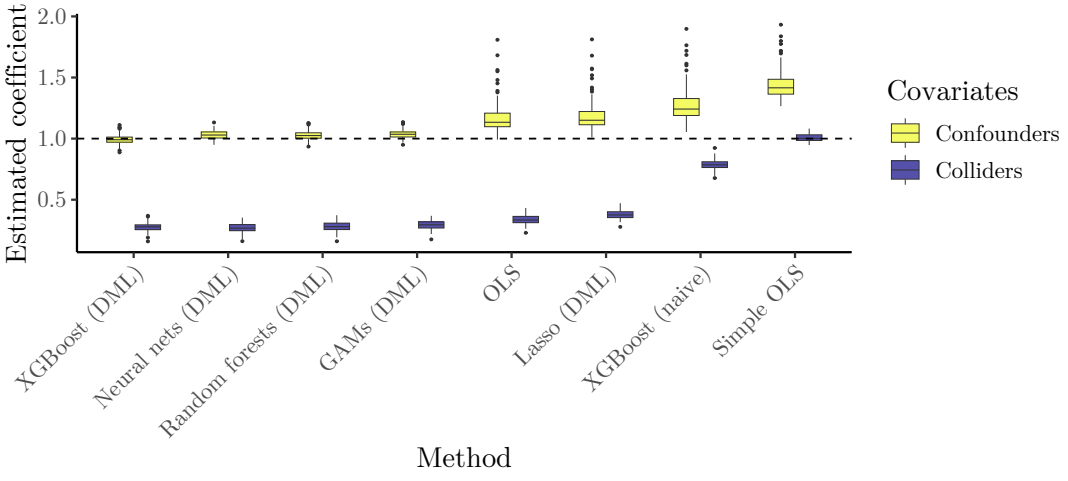


Figure 2.14: Results for Case 9 - distribution of estimated coefficients across 100 simulations when adjusting for confounders (yellow) or colliders (blue). The dashed line marks the true causal effect ($\beta = 1$).

any covariates. Interestingly, results of the baseline simulation are basically inverted if we adjust for colliders instead of confounders: The "better" we adjust with flexible DML methods, the *more* (downward) biased our results are. This emphasizes the importance of correctly classifying

the covariates before entering them into any estimation algorithm. Since DML has no way of knowing whether a variable is a good or bad control, researchers have to make this decision based on theory and domain knowledge. These results caution against blindly throwing all available variables into DML or any other algorithm for causal effect estimation.

Cases 10-11

In the final Cases 10 and 11, we return to the baseline with the random functional form, but now we vary parameters of the DML algorithm. First, in Case 10, we vary the number of folds K into which we split the dataset. Chernozhukov et al. (2018) note that larger values of K lead to larger samples for the ML step, which can help to increase predictive accuracy. At the same time, the sample for estimating the effect becomes smaller, but this step might depend less on the sample size. The authors mention values of 4 or 5 for K as superior to $K = 2$ in their tests, but we showed that 2 and 10 are the most common values for K in applications (Figure 2.1E; although $K = 10$ is almost exclusively used in combination with lasso). For extremely small samples ($N = 20$), we observe that the median estimate of the flexible DML methods becomes more accurate as K gets larger, but also that the variation in the estimates increases (Figure 2.15A). This is because

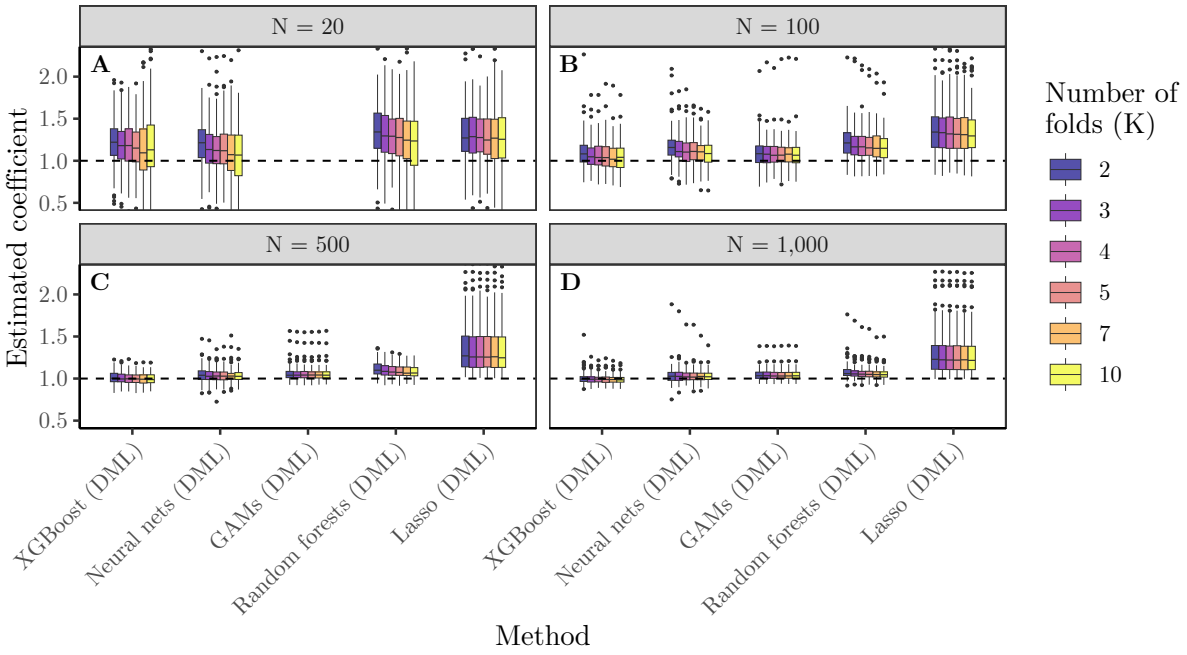


Figure 2.15: Results for Case 10 - varying the number of folds K in DML for different sample sizes; distribution of estimated coefficients across 100 simulations. The dashed line marks the true causal effect ($\beta = 1$). **A** 20 observations. The number of parameters in flexible GAMs becomes too large for this sample size. **B** 100 observations. **C** 500 observations. **D** 1,000 observations.

the ML methods benefit from getting more observations for training; but for the extreme case of $K = 10$, there are only two observations left for estimating the effect.[6] For moderate sample sizes (Figure 2.15B and 2.15C; $N = 100$ and $N = 500$), the downsides of larger numbers of folds disappear and larger Ks lead to both more accurate median estimates and smaller or at least similar variation in estimates. The same holds for larger samples (Figure 2.15D), but there the choice of K is less influential, since even small Ks provide sufficient observations to the ML algorithms. Similarly to before, we again observe here that only the ML methods that benefit from larger sample sizes are significantly affected by the choice of K.

Secondly, we vary how often we repeat DML to make the estimates more robust to the randomness in the sample splitting. We repeat the algorithm S times and report the median estimate from these S estimates, which is more robust than the mean (Chernozhukov et al., 2018). Our results show no obvious pattern in the accuracy of the median estimate: it remains relatively similar for larger numbers of repetitions (Figure 2.16). However, the variation of the estimates decreases moderately as the number of repetitions increases. This is most pro-

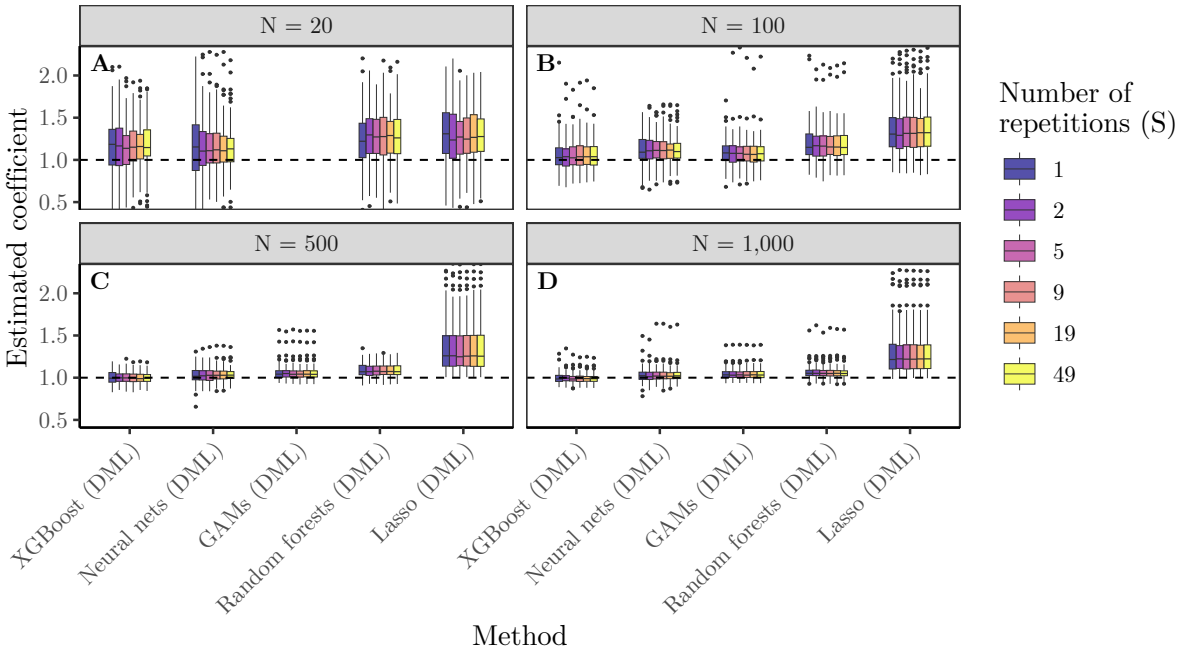


Figure 2.16: Results for Case 11 - varying the number of algorithm repetitions S in DML for different sample sizes; distribution of estimated coefficients across 100 simulations. The dashed line marks the true causal effect ($\beta = 1$). **A** 20 observations. The number of parameters in flexible GAMs becomes too large for this sample size. **B** 100 observations. **C** 500 observations. **D** 1,000 observations.

[6] In such extreme cases, the alternative DML2 algorithm delivered more stable estimates, but the advantage was already negligible at $K = 5$.

nounced for very small samples, especially for DML with neural networks (Figure 2.16A). In larger samples, the benefit of many repetitions is negligible (Figure 2.16C-D). We will return to evaluating the benefits of multiple repetitions in the context of our application, where we observe a substantially larger impact of the choice of S.

2.4.4 Choosing between different ML algorithms

While simulation results from the previous sections have given us an understanding of the capabilities of (versions of) DML in different scenarios, the choice between the different ML algorithms within DML is still not obvious, i.e., it is not obvious whether we should implement DML with, e.g., random forests, XGBoost, or neural networks. In the following analysis we investigate whether the accuracy of the different ML methods in predicting treatment and outcome, respectively, is indicative of the accuracy with which they can recover the causal effect. The underlying assumption is that a method which models the relationships between the confounders and treatment/outcome well will also be able to eliminate more of the confounding influence and thus deliver a less biased effect estimate compared to an estimator that is less accurate in predicting treatment and outcome. The cross-fitting procedure allows us to directly extract the predictive accuracy by averaging the errors across the respective folds that we held out from training. We use our baseline simulation to calculate the MSE for the prediction of treatment and outcome and the respective bias in the causal estimate.

The results indicate a clear pattern: Low values for the MSE(W) and the MSE(Y) are associated with yellow colors, i.e., small bias (Figure 2.17). In other words, when the prediction of Y and W is good (small MSE), the bias in the causal estimate is small. Consequently, the scatterplot shows that the points in the bottom left corner have mostly small bias since both predictions are accurate, points in the upper right corner have mostly high bias because of inaccurate predictions. Hence, under the assumption that we observe and adjust for all relevant confounders, we can use the predictive accuracy of the first stage as a guide for which estimate to trust more. If DML with two different ML methods gives very different results in terms of the estimated treatment effect, it seems reasonable to rely more on the one with smaller MSEs in the treatment and outcome predictions. Importantly, this comparison is only reasonable when the unconfoundedness assumption holds and when the models that we compare do not include any bad controls.

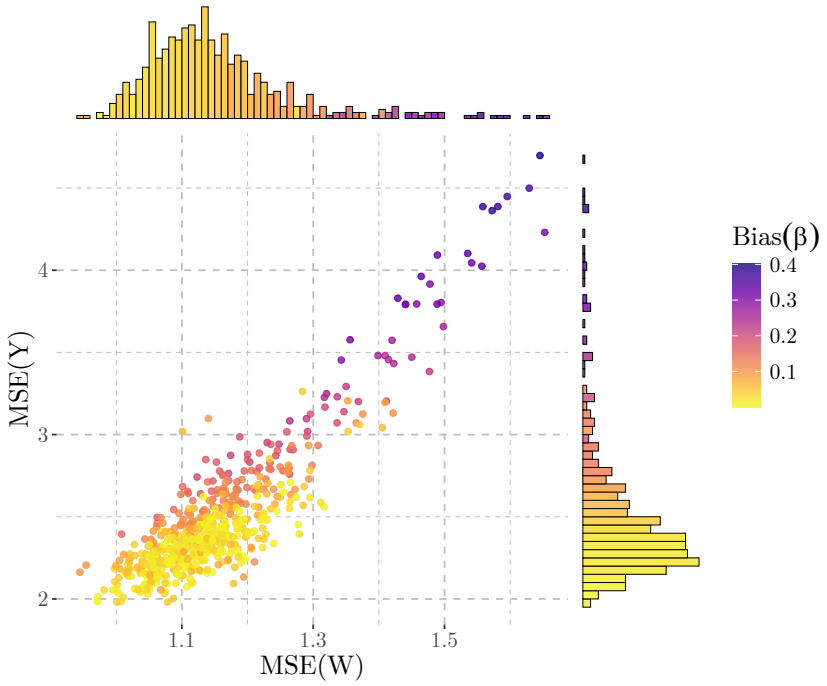


Figure 2.17: The relationship between the predictive accuracy of the first stage and the bias in the effect estimation. The horizontal axis shows the test MSE of the treatment prediction, the vertical axis the test MSE of the outcome prediction. The color scheme indicates the degree of bias, i.e., yellow represents a very small bias in the estimated causal effect, blue is relatively large bias. The histograms in the margins display the frequency distribution across the respective MSEs. We excluded the 2% largest outliers in terms of MSE.

2.5 Application to real-world data

In this section, we use DML for an application that estimates the effect of one determinant of house prices. Building on Rosen's (1974) theory of hedonic prices, a variety of studies have applied hedonic price regressions to estimate the effects of particular attributes of goods or services (e.g., Agrawal and Kamakura, 1999; Nelson, 1978). We replicate and extend the central regression model of Harrison and Rubinfeld (1978), who use housing market data to measure the effect of air pollution on housing prices. As part of their analysis, they employ a rich cross-sectional dataset from the Boston area (506 observations, 14 variables). They state that potential buyers know about the damages of air pollution and consequently are willing to pay more for a house in an area with low air pollution levels compared to an identical house in an area with high air pollution levels, and this higher willingness-to-pay would be reflected in a negative estimate for the effect of air pollution. Harrison and Rubinfeld (1978) argue that identification of this effect is possible with variation in air pollution over space after adjusting for a large number of observed neighborhood variables.

The authors assume that some covariates influence both air pollution and house prices, while others only affect house prices (Figure 2.18). Additionally, they argue for nonlinear relationships between housing attributes and house prices, thus transforming some of the variables accordingly. In their preferred specification (Equation 2.5), mv is the median value of homes in a specific tract, nox (concentration of nitrogen oxides[7]) is the measure of air pollution, and β is the coefficient of interest:

$$\begin{aligned} log(mv) = \alpha + \beta nox^2 + \gamma_1 rm^2 + \gamma_2 age + \gamma_3 log(dis) + \gamma_4 log(rad) + \gamma_5 tax \\ + \gamma_6 ptratio + \gamma_7 (B - 0.63)^2 + \gamma_8 log(lstat) + \gamma_9 crim \\ + \gamma_{10} zn + \gamma_{11} indus + \gamma_{12} chas + \epsilon. \end{aligned} \tag{2.5}$$

The other variables refer to the covariates in Figure 2.18 and are included with varying functional forms. The authors give theoretical justification for some of these functional forms, but choose others simply because they provide good fit. Given the large number of covariates and the authors' emphasis on the importance of specifying the correct functional forms, this application is an ideal example to demonstrate the strengths and flexibility of DML, which offers a data-driven way to flexibly learn functional forms and appropriately adjust for the covariates. Variable descriptions are available in Table IV of Harrison and Rubinfeld (1978), which we replicate in our Appendix (Table A.2), where we also provide descriptive statistics (Table A.3).

Figure 2.18: The causal structure Harrison and Rubinfeld (1978) argue for when estimating the effect of air pollution on house prices. Some variables influence both treatment and outcome, others only affect outcome. The authors include all variables in a regression model of the outcome. We include the same variables in DML to predict both treatment and outcome and finally estimate the effect.

When we first applied DML with 5-fold cross-fitting to this dataset, we observed that repeated runs of the standard algorithm led to substantially different coefficient estimates, especially for the most flexible DML methods. Since this could be a consequence of the random

[7] measured in parts per hundred million (pphm)

sample splitting in the first step, we systematically experimented with different numbers of repetitions S to be more robust to this kind of randomness. For each number of repetitions, we run the full algorithm (including S repetitions) 100 times and report the distribution of median estimates for each algorithm. For example, for the number of repetitions $S = 5$, we estimate the model 5 times, compute the median estimate, and repeat this procedure 100 times. The results indicate a clear pattern: the larger the number of repetitions S, the more stable the median coefficient estimates (Figure 2.19). This pattern is considerably more pronounced in this real-

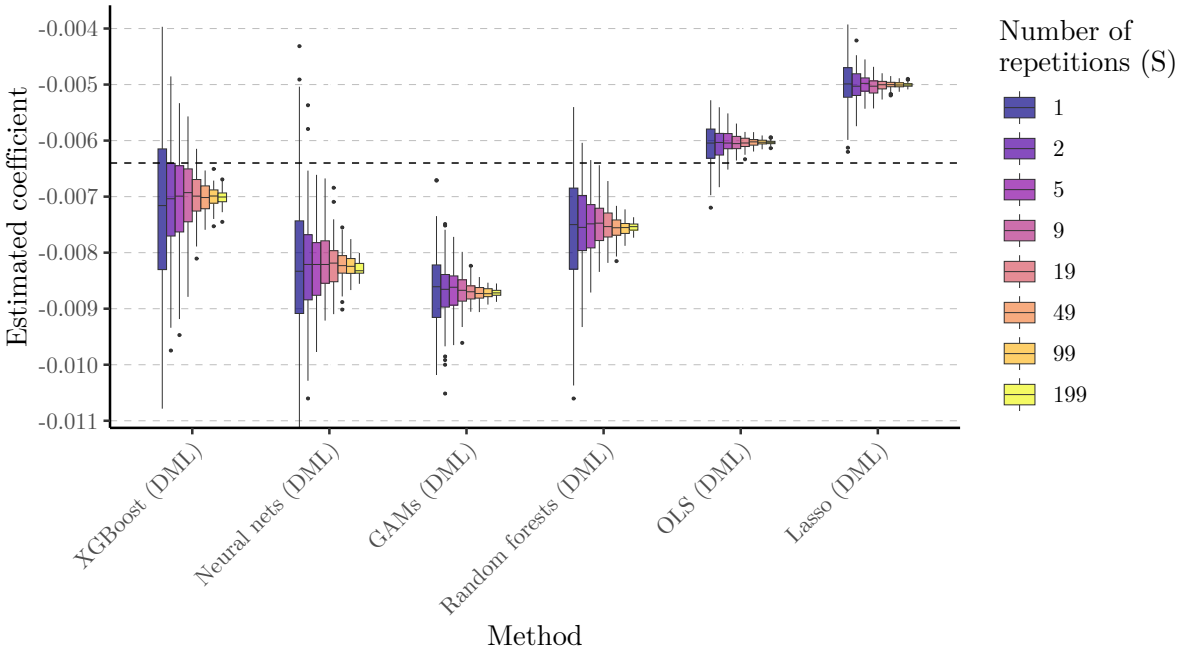


Figure 2.19: Varying the number of algorithm repetitions S in DML when estimating the effect of air pollution on house prices. Each boxplot shows the distribution of estimated coefficients across 100 repeated runs when using DML with S algorithm repetitions. The black dashed line indicates the original estimate of Harrison and Rubinfeld (1978), -0.0064.

world dataset compared to our previously simulated data (Figure 2.16). One explanation is that not all variables in this real-world data neatly follow a symmetric distribution like the normal distribution in our simulations. As a consequence, the random sample splitting can lead to very heterogeneous samples for different runs. However, this is in essence an empirical question and therefore researchers should try different numbers of sample splits for their applications and choose a number at which the results become stable. For example, since we observed stable results for relatively large numbers of repetitions in this application, we use $S = 199$ for our final estimation and report the median for the effect estimates and standard errors.

Table 2.5 presents the estimation results from applying DML to this real-world dataset. In the first three columns, we report the method, the effect estimate and the standard error,

respectively. Since the air pollution variable is specified as nonlinear by the authors (nox^2), the estimated effect depends on the air pollution level at which it is evaluated. Thus, in column four, we display the percentage effect of a one-unit change in air pollution (nox) evaluated at the mean level of nox (5.547 pphm). Columns five and six contain the mean squared error (MSE) of the respective ML method when predicting outcome and treatment in the first stage of DML. Under the assumption that the causal structure is correctly specified, we can use this as an indicator for which effect estimate to trust most, as we previously discussed in Section 2.4.4.

Table 2.5: Results for the effect of air pollution on housing prices

Method	Effect estimate	Std. error	Effect at mean (%)	MSE(Y)	MSE(W)
OLS (H&R)	-0.0064	0.0011	-7.08	-	-
Simple OLS	-0.0146	0.0011	-16.17	-	-
XGBoost (naive)	-0.0137	0.0013	-15.14	-	-
OLS (raw)	-0.0058	0.0011	-6.47	-	-
OLS (flex)	-0.0071	0.0013	-7.88	-	-
OLS (DML, flex)	-0.0093	0.0006	-10.30	0.5571	1437.38
OLS (DML, raw)	-0.0059	0.0012	-6.58	0.0402	58.91
OLS (DML, H&R)	-0.0064	0.0012	-7.14	0.0375	54.88
GAMs (DML)	-0.0087	0.0015	-9.63	0.0346	42.71
Neural nets (DML)	-0.0081	0.0016	-9.00	0.0349	34.46
Lasso (DML, flex)	-0.0071	0.0015	-7.86	0.0316	33.89
XGBoost (DML)	-0.0070	0.0019	-7.73	0.0295	20.85
Random forests (DML)	-0.0075	0.0018	-8.27	0.0266	19.03

Note: MSE: mean squared error. H&R: covariate specification by Harrison and Rubinfeld (1978). raw: only using untransformed variables. flex: including squares and first-order interactions of all variables.

The first row replicates the original regression model specification of Harrison and Rubinfeld (1978) with an OLS regression, resulting in an effect estimate of -0.0064. Row two shows the results when not adjusting for any covariates, which leads to an absolutely larger effect estimate (-0.0146). The naive XGBoost implementation leads to a similarly large estimate, suggesting a similar inability to adjust for confounding. OLS without any transformed variables leads to an effect estimate that is smaller in absolute terms than the authors' nonlinear specification. The more flexible specification "OLS (flex)" – including squares and first-order interactions of all variables – leads to a larger estimate of -0.0071. For all of these methods we do not observe a predictive accuracy on unseen data in the last two columns. However, when we implement DML with different predictive algorithms, we observe the predictive accuracy in the first stage. The DML results are sorted in descending order by the MSE of the first-stage prediction. As the first DML implementation, we use an OLS regression in the first stage, while including squares and interactions of all variables. The results illustrate the problems with this approach: Since the number of parameters becomes large relative to the number of observations in each sample, the regression overfits significantly, as evident in the extremely poor predictive accuracy of the

first stage. Also, the estimated effect is relatively large, in addition to varying heavily across repetitions of the algorithm. The next two methods again use OLS regression in DML, but now either use all variables linearly ("OLS (DML, raw)") or use the specification by Harrison and Rubinfeld (1978) ("OLS (DML, H&R)"). This should give results similar to using the same specifications directly within an OLS regression. Within DML, however, we can also observe the predictive accuracy. The effect estimates are indeed very similar, and in addition, the MSEs for outcome and treatment indicate the merit of the authors' nonlinear specification: Their specification leads to a higher predictive accuracy compared to the purely linear model. The final five rows show results from using DML with different flexible predictive algorithms. All flexible ML methods perform better than the OLS regression in the prediction tasks. Under the assumption that the identification strategy is valid, we can thus expect the effect estimates of the flexible methods to be more precise than those of the OLS regression. There is also some variance in the flexible DML estimates, but they all agree in being more negative than the coefficient estimated in the authors' specification, suggesting that the authors in the original publication somewhat underestimate the true effect of air pollution. It is also worth discussing the relationship between the estimated effect and the directions of the confounding influence. The largest estimate in absolute terms comes from not adjusting at all ("Simple OLS"), while the smallest estimate comes from adjusting only linearly ("OLS (raw)"). The more flexible and more credible specifications result in estimates substantially smaller than the naive approaches, but larger than the ones from linear adjustment. We can explain this observation by different confounders biasing the estimated effect in distinct directions. The linear specification might adjust well for confounders biasing the effect downwards, but poorly for confounders biasing the effect upwards. This heterogeneity in confounding direction is also supported by Harrison and Rubinfeld (1978), who argue for a upward bias of some variables (e.g., distance to highways, distance to employment centers) and a downward bias of others (e.g., percentage lower status, crime rate, industry proportion) (see Figure A.2 in the Appendix).

2.6 DML beyond the partially linear model

So far, we have introduced and assessed DML for the partially linear model with one continuous treatment in a cross-sectional setting under the assumption of unconfoundedness. However, the general statistical theory translates to various other settings as well. In this section, we

briefly mention further situations that differ in terms of the treatment, the data structure or the identification strategy.

2.6.1 DML in the interactive model

The partially linear regression (PLR) model assumes the treatment to have a homogeneous, additively separable effect on the outcome. We can apply DML with the PLR for both continuous and binary treatments. However, for binary treatments, another version of DML allows for arbitrary effect heterogeneity by letting the treatment fully interact with the confounders (Chernozhukov et al., 2018). Contrary to Equation 2.1 for the PLR, the function $g_0()$ in Equation 2.6 contains both the treatment and the confounders:

$$Y = g_0(W, \boldsymbol{X_c}) + V_y. \tag{2.6}$$

In this setting, we can use a slightly different estimator for the average treatment effect (ATE) without having to assume homogeneous treatment effects. In what follows, we briefly describe the intuition behind this estimator by first starting with two "naive" ML estimators, which we then combine to form the estimator we can use for DML in the interactive model. The first "naive" estimator for the ATE uses regression adjustment in the outcome model, that is:

$$\hat{\tau}_{naive} = \frac{1}{n} \sum_{i=1}^{n} \Big(\hat{\mu}_{(1)}(X_i) - \hat{\mu}_{(0)}(X_i) \Big). \tag{2.7}$$

Here, $\hat{\mu}_{(1)}(X_i)$ is the predicted outcome when we only use the treated observations for prediction; $\hat{\mu}_{(0)}(X_i)$ is the predicted outcome only using control observations. These predictions can come from any predictive method, including ML methods. However, as we discussed for the PLR, this estimator is biased if we do not specify the parametric model correctly, because we only consider the outcome model (Chernozhukov et al., 2018). The second "naive" estimator in this setting relies on the estimated propensity score. When predicting a binary treatment from confounding variables, we can interpret the prediction as the (estimated) probability of receiving treatment (or the estimated "propensity score", see Imbens and Rubin, 2015, Chapter 12). We can then use this estimated propensity score $\hat{e}(X_i)$ to weight observations by their probability of being treated, by which we make treatment and control group more comparable. This procedure is called "inverse probability weighting" (IPW, or inverse propensity weighting, or Horvitz-Thompson estimator) and results in the estimator in Equation 2.8 (see, e.g., Hernán and Robins,

2020; Rosenbaum and Rubin, 1983). Here, we weight the treated units by the inverse of the estimated propensity score (first term), while we weight the control units (second term) by the inverse of one minus the estimated propensity score:

$$\hat{\tau}_{IPW} = \frac{1}{n}\sum_{i=1}^{n}\Big(\frac{W_i Y_i}{\hat{e}(X_i)} - \frac{(1-W_i)Y_i}{1-\hat{e}(X_i)}\Big). \tag{2.8}$$

We can estimate the propensity score for this method using any predictive method, including ML methods. However, the estimator can again be biased if we did not correctly specify the treatment (or propensity) model, because it does not consider the outcome model (Imbens and Rubin, 2015, Chapter 12).

Nevertheless, similar to the PLR, we can construct an unbiased estimator considering both the treatment and the outcome model simultaneously in a "doubly robust" way. One name for this estimator is "Augmented Inverse Propensity Weighting" (AIPW), because it augments the IPW estimator with the predictions from the outcome model (Robins et al., 1994; Robins and Rotnitzky, 1995). Thus, we can understand the AIPW estimator as a combination of the naive estimator $\hat{\tau}_{naive}$ and the IPW estimator $\hat{\tau}_{IPW}$. The innovation of Chernozhukov et al. (2018) is using ML methods for both the propensity and the outcome model in combination with cross-fitting, which allows for flexible weighting and adjustment. The intuition for the estimator in Equation 2.9 is very similar to the DML estimator in the PLR:

$$\begin{aligned}\hat{\tau}_{AIPW} = \frac{1}{n}\sum_{i=1}^{n}\Big(& \overbrace{\hat{\mu}_{(1)}(X_i) - \hat{\mu}_{(0)}(X_i)}^{\text{Regression adjustment using ML estimators}} \\ & + \frac{W_i}{\hat{e}(X_i)}\underbrace{(Y_i - \hat{\mu}_{(1)}(X_i))}_{\text{residual treated}} - \frac{1-W_i}{1-\hat{e}(X_i)}\underbrace{(Y_i - \hat{\mu}_{(0)}(X_i))}_{\text{residual control}}\Big).\end{aligned} \tag{2.9}$$

The first term is exactly the naive estimator from before. The second term is similar to the IPW estimator, but instead of weighting the outcome variable Y_i, we weight the residual of the outcome variable. The outcome residual is the observed outcome Y_i minus the prediction in the outcome model for treated ($\hat{\mu}_{(1)}(X_i)$) or control ($\hat{\mu}_{(0)}(X_i)$) units, respectively. The general structure of the DML algorithm remains. The cross-fitting procedure is still necessary, but here, we need to train two outcome models and use them for prediction, one for the treated observations ($\hat{\mu}_{(1)}(X_i)$), one for the control observations ($\hat{\mu}_{(0)}(X_i)$). The treatment model is now the propensity model, so we should make sure that the predictive/ML method returns a

probability measure. With these three models, we can estimate $\hat{\tau}_{AIPW}$ for each fold and average the estimates across the folds to obtain the final estimate (Chernozhukov et al., 2018).

Like other estimators relying on the estimated propensity score, this estimator is sensitive to propensity score estimates that are too close to 0 or 1. In these cases, weighting by an extremely small number can inflate the residual and thus lead to a disproportionate impact of some units. There are multiple approaches for dealing with this issue in traditional methods (see, e.g., Imbens and Rubin, 2015). Chernozhukov et al. (2018) use *trimming* in one of their applications, where they trim the propensity score at extreme values. For this, one defines a trimming threshold (e.g., .01), and excludes all observations with propensity scores smaller than that threshold or larger than one minus that threshold (e.g., $< .01$ or $> .99$). In our experiments with DML in this setting, we observed that these extreme values for the propensity score occur more often for ML methods compared to traditional methods. While this could be a consequence of overfitting in poorly trained models, it could also indicate violations of the overlap assumption in these applications (Imbens and Rubin, 2015, Chapter 14).

2.6.2 DML for instrumental variables models

In some applications, the unconfoundedness assumption might not be plausible. However, there could be an instrumental variable which we consider exogenous after adjusting for observed covariates. In these cases, DML can make the conditional exogeneity assumption of the instrument more plausible by flexibly adjusting for the observed covariates. The respective DML procedure is very similar to the unconfoundedness setting, though it now involves an additional prediction model for the instrumental variable (Chernozhukov et al., 2018). For example, in the partially linear instrumental variables model, we make predictions and compute residuals for the treatment, outcome, *and* instrument. Then, instead of using the residuals in a linear regression, we plug them into the standard IV estimator in place of the respective variables. For a binary treatment and a binary instrument, Chernozhukov et al. (2018) demonstrate the estimation of local average treatment effects (LATE, Imbens and Angrist (1994)), which allows the instrument to interact arbitrarily with the covariates.

2.6.3 DML in further settings

Here, we briefly mention other settings in which researchers might want to apply DML. First, like Chernozhukov et al. (2018) in their first application, we can also use DML in experimental

settings. There, the goal is not necessarily to adjust for observed confounding, which we can largely rule out if the random assignment was properly executed. Nevertheless, adjusting flexibly for covariates might help to increase the statistical precision of the estimates. Due to random assignment, we typically know the treatment or propensity model and do not need to estimate it, but using ML in the outcome model could be advantageous. Second, in some cases, we are interested in the effects of more than one treatment variable. If the effects of multiple variables are causally identified, we can simply compute the residual for each treatment and regress the outcome residual on all of the multiple treatment residuals (Chernozhukov et al., 2018). Similarly, DML translates to settings with multi-valued treatments or multiple binary treatment versions (see, e.g., Farrell, 2015; Knaus, 2022). Fourth, there is currently a lack of clear guidance about whether or how DML extends to settings with panel or longitudinal data. Chernozhukov et al. (2018) focus on a cross-sectional setting, and adaptations to panel settings do not seem trivial for two reasons: First, the time dimension complicates the cross-fitting procedure; and second, the elimination of unobserved heterogeneity is not straightforward in complex nonlinear models. In Chapter 3, we further explore these issues and consider various potential solutions.

2.7 Discussion

Our simulations and the application have demonstrated the capability of DML to adjust for various forms of confounding influences, as long as flexible ML methods are used (e.g., boosted trees, neural nets, or random forests), and the sample size is reasonably large relative to the number of confounders. Also, a sound choice of parameters within the DML algorithm can improve the accuracy (e.g., number of folds) and robustness (e.g., number of repetitions) of the estimates. Furthermore, we have illustrated that accuracy in predicting treatment and outcome is indicative of accuracy of the effect estimation. Additionally, we have highlighted that assumptions about causal structure are still required, as DML cannot account for unobserved confounding or recognize a bad control. Based on these findings, in what follows we derive recommendations for applied researchers about when and how to apply DML (see Table 2.6).

Table 2.6: Problems and recommendations for the application of DML

Decision problem	Recommended decision
Is the effect causally identified?	Determine identification independently of DML. If the effect is identified based on observables, use DML to flexibly adjust for covariates.
Which variables should we include?	Confounders, outcome influencers, (noise variables), no instruments, no bad controls.
Which ML algorithm should we use?	Use flexible ML methods like random forests, XGBoost, or neural networks, ideally with some parameter tuning. Choose between ML algorithms based on the predictive accuracy in the first stage.
How many folds K should we use for cross-fitting?	Small samples: $K = 5 - 10$; larger samples: $K = 2 - 5$.
How many repetitions S should we use?	Test different sizes of S, choose the lowest S that still leads to stable estimates.
What if we have a very small sample size (e.g., $n < 100$)?	Even though the sample size limits DML's flexibility, it can still outperform OLS. Still, run a plausible parametric model in addition to DML and compare the estimates.
What if we are confident in our parametric model?	You can still use DML as a robustness check to rule out functional form misspecifications.

2.7.1 *When* should we (not) use DML?

First, while we have shown DML to be a useful estimation method, it is *not* a novel strategy for causal *identification*. That is, if unobserved confounding is a problem in a specific application, DML is not a cure for it. Instead, DML can serve as a way of estimating the effect from the data *after* one has decided for a plausible identification strategy. Then, DML can estimate the effect more flexibly than traditional methods by modeling potentially complex functional forms in a data-driven way. In our study, we have focused on using DML for settings with the identification strategy "adjusting for observed confounders". However, DML is also applicable for strategies like randomized experiments (where it can potentially increase precision) and instrumental variables, with researchers continuously exploring extensions to further strategies (e.g., difference-in-differences (Chang, 2020)). For strategies that depend on adjusting for covariates well, DML may provide a solution that does not rely on correctly prespecified functional forms. In summary, DML does not help to relax identification assumptions, but it can relax estimation assumptions. For any application, we should first use theory and domain knowledge to outline a plausible causal structure (which we can represent, e.g., with a DAG). If one of the mentioned identification strategies is feasible in this causal structure, researchers can use DML to estimate the effect without making strong parametric assumptions. Second, if researchers find themselves in a situation with cross-sectional data, where identification is possible, and there is no strong theoretical basis for choosing specific functional forms, DML offers a promising alternative to arbitrarily specifying a parametric model. Even in cases where theory suggests functional relationships for some confounders, we recommend using DML as it might help to assess the robustness of the specification to different functional forms learned from the data.

Third, DML's ability to flexibly adjust for covariates is especially pronounced for sample sizes significantly larger than the number of important raw covariates. The method's benefit over parametric methods will likely be less pronounced for very small samples, so we particularly recommend using DML when the number of observations significantly exceeds the number of important raw covariates. Finally, at this point, we can give no recommendation for the use of DML with panel data. The original paper of Chernozhukov et al. (2018) focuses on establishing DML for settings with cross-sectional data. While there is gradual process for understanding DML in panel settings and the value it can add there, at the time of writing, there are no theoretical guarantees available that are comparable to the cross sectional setting. However, we will explore the challenges of panel data and potentially feasible DML approaches in Chapter 3.

2.7.2 *How* should we use DML?

Once we have decided that using DML is appropriate in our application, we still have a plethora of decisions to make in the process of applying DML. The first decision is which variables to include in the estimation. This is related to the question of identification, but goes further. For identification, we must adjust for all confounders (X_c), meaning variables influencing both the treatment (or instrument in instrumental variables settings) and the outcome. Secondly, we must not include any bad controls such as colliders (X_{coll}) (see, e.g., Angrist and Pischke, 2009; Cinelli et al., 2024). These two criteria ensure identification, but other types of variables also can influence the precision of the estimates. We should include variables only influencing the outcome (X_p), should not include variables only influencing the treatment (or instrument) (X_z), and can, but do not have to, include noise variables (E). We recommend using DAGs to systematically consider to which of these categories each covariate belongs.

The second decision is which ML algorithm to use within DML. The literature overview in Section 2.2 has shown that most published applications of DML rely on lasso regressions. In contrast, based on our simulation analyses, we caution against the use of lasso because DML using lasso with raw covariates produces biased estimates as soon as the confounding is not only linear. We rather recommend flexible methods with the ability to fit any functional forms, like random forests, boosted trees, or neural networks. Ideally, researchers should tune these methods to achieve high quality predictions and avoid overfitting. If the computational resources allow it, running DML with different ML algorithms can help to assess the robustness of the estimates. Also, estimating a traditional parametric model is a helpful reference point. If all

methods agree, we can have confidence that the estimates are robust to different functional forms. If the DML methods agree, but are very different from the parametric model, we might want to question the plausibility of the parametric model. If different DML methods lead to very different estimates, we can proceed in three different ways: (1) We could return to the causal structure and question whether our identification strategy is valid. If we are confident that it is, we could (2) use the results as as form of sensitivity analysis and report the results as a range of plausible estimates. (3) If we want to obtain a most plausible point estimate, we could use the predictive accuracy of each ML algorithm in the first stage to determine which parameter estimate we trust most. If we have entered the correct variables, methods performing better at predicting treatment and outcome should also adjust more completely for confounding and thus deliver more accurate causal estimates. If we observe different algorithms performing best for outcome and treatment, respectively, it is also possible to use the best-performing method for the respective prediction task. If we had to pick the one single ML method that performs best, we would recommend the use of boosted trees in the XGBoost implementation. The reason is that XGBoost performs very well across a very broad range of settings, and we could rarely identify any setting in which XGBoost is clearly dominated by a different method.

Third, we must decide about the number of folds K into which we split the dataset in DML. One important factor for this decision is the sample size. For small samples, we should choose moderate to large numbers of folds (5-10 folds), because they provide more observations to the prediction task, which increases the flexibility of the ML methods. However, K should not be so large that very few observations remain in the fold we use for estimating the effect. In larger samples, the choice of K is less influential. Since large values for K quickly increase computation times, a choice of $K = 2$ is reasonable for very large samples. However, we still recommend a moderate number of folds (e.g., $K = 5$) if computational resources allow it.

Finally, we have to choose the number of DML repetitions S for achieving robust estimates. We recommend treating this as an empirical question: For any specific application, researchers can run DML with different S and observe whether the estimates of repeated runs are stable. As we illustrated in Section 2.5, we should finally choose the smallest S with stable estimates because larger numbers of S significantly increase computation times.

2.7.3 Limitations of our study

Our study evaluates the DML method in a variety of settings, but there are interesting settings and implementations we did not consider. First, in our simulations, we focus on the partially linear model under unconfoundedness with a continuous and homogeneous treatment. It would be interesting to see how our conclusions translate to settings with binary treatments, treatment heterogeneity and unobserved confounding with instrumental variables. Second, Chernozhukov et al. (2018) describe two different score functions for employing DML in the partially linear model. We only use the "partialling-out" score function and do not contrast it with the "IV-type" score function, since we did not see significant differences in a few initial simulations. Third, one could compare additional ML algorithms within DML, implement DML with distinct ML methods for the treatment and the outcome model, respectively, and compare DML to further traditional or ML-based statistical methods. Fourth, in most simulations, we are changing one or two factors of the DGP while holding all others constant. This is a practical strategy, but might not uncover all nuances of how different characteristics of the data are interrelated. Lastly, we assess DML in a cross-sectional setting without a time dimension. However, many disciplines regularly deal with repeated observations over time in panel or longitudinal data. In these settings, methods like fixed effects estimation can eliminate the influence of time-constant, unobserved variables (Wooldridge, 2010, Chapter 10). The feasibility of DML in these settings is not yet well understood, since the time-dimension complicates the cross-fitting procedure and nonlinear relationships might hinder the elimination of the unobserved effects. Exploring if and how we can adapt DML to allow estimation under weaker assumptions in these settings is an important avenue for further research. We address this challenge in the following Chapter 3.

Chapter 3

Double Machine Learning meets Panel Data – Promises, Pitfalls, and Potential Solutions

Jonathan Fuhr and Dominik Papies

Statement of contribution

Jonathan Fuhr conducted all literature review and data simulation, suggested and implemented the methods and analyses, and wrote the first draft of the manuscript. Dominik Papies devised the initial idea, provided continuous supervision and suggestions, and gave extensive feedback to an earlier version of the working paper.

Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC number 2064/1 – Project number 390727645. The authors acknowledge support by the state of Baden-Württemberg through bwHPC.

Chapter 3 is based on the working paper "Fuhr, J., and Papies, D. (2024). Double Machine Learning meets Panel Data – Promises, Pitfalls, and Potential Solutions. arXiv:2409.01266 [cs, econ, stat]".

Abstract

Estimating causal effect using machine learning algorithms can help to relax functional form assumptions if used within appropriate estimation frameworks. However, most of these frameworks assume settings with cross-sectional data, whereas researchers often have access to panel data, which in traditional methods helps to deal with unobserved heterogeneity between units. In this paper, we explore how we can adapt double/debiased machine learning (DML) (Chernozhukov et al., 2018) for panel data in the presence of unobserved heterogeneity. This adaptation is challenging because DML's cross-fitting procedure assumes independent data and the unobserved heterogeneity is not necessarily additively separable in settings with nonlinear observed confounding. We assess the performance of several intuitively appealing estimators in a variety of simulations. While we find violations of the cross-fitting assumptions to be largely inconsequential for the accuracy of the effect estimates, many of the considered methods fail to adequately account for the presence of unobserved heterogeneity. However, we find that using predictive models based on the correlated random effects approach (Mundlak, 1978) within DML leads to accurate coefficient estimates across settings, given a sample size that is large relative to the number of observed confounders. We also show that the influence of the unobserved heterogeneity on the observed confounders plays a significant role for the performance of most alternative methods.

3.1 Introduction

Across multiple quantitative disciplines, recent years have seen an explosion of research on methodologies that try to use machine learning (ML) to help estimate causal effects (e.g., Athey et al., 2019; Chernozhukov et al., 2018). These methods aim to relax assumptions in the causal estimation process by using modern ML methods to learn certain properties of the data. Arguably one of the most popular of these methods is the double/debiased machine learning (DML) framework by Chernozhukov et al. (2018), which we already introduced in Chapter 2. DML can help relax assumptions about how to adjust for observed confounders by modeling the confounding relationships with flexible ML methods. That is, in the case of a large number of potentially important confounders, or in cases where the functional forms of the confounding influences are unknown, DML uses flexible ML methods to pick the most important confounders and to adjust for them flexibly. This framework has seen applications in a variety of disciplines (e.g., Felderer et al., 2023; Gordon et al., 2022; Parpouchi et al., 2021), as well as further developments and extensions to settings beyond the original ones (e.g., Bodory et al., 2022; Chiang et al., 2022; Liu et al., 2021).

At the same time, many more traditional methods from statistics and econometrics are still the default approaches for credible causal effect estimation. This is mostly because they aim to relax assumptions that are stronger than the estimation assumptions DML can relax. These methods address assumptions about causal identification, e.g., how to estimate causal effects in the presence of unobserved confounding. Examples are panel data methods, difference-in-differences, synthetic control, instrumental variables, and regression discontinuity designs (e.g., Cunningham, 2021; Huntington-Klein, 2022). While there has been progress in using some of these methods in combination with the DML framework (see, e.g., Chernozhukov et al., 2024), very little research has explored how to adapt DML to settings with panel data, which will be the focus of our paper.

In many applications, we observe the same units (e.g., individuals, firms, cities, etc.) repeatedly over time. This kind of data structure – panel data – can be very beneficial for the identification and estimation of causal effects, since it can help to eliminate any time-constant source of unobserved confounding or heterogeneity between units (Wooldridge, 2010). However, even when using panel data in this way, we could still benefit from methods that flexibly adjust for the *observed* and *time-varying* confounders. Hence, combining a method like DML with

panel data methods could enable us to relax both assumptions about unobserved confounding and about functional forms in the observed confounding. For example, we might be interested in estimating the effect of price on demand for a product repeatedly sold across various stores. Our data might contain information about advertising and promotions, which act as covariates we would want to flexibly adjust for with a method such as DML. However, time-constant unobserved store characteristics might also influence both the price and the demand (e.g., the management quality), which we could potentially eliminate by exploiting the panel structure.

Unfortunately, it does not yet seem obvious how we can use DML for panel data in the presence of unobserved heterogeneity. Chernozhukov et al. (2018) developed the original method and its statistical guarantees in a cross-sectional setting under the assumption of unconfoundedness (or with instrumental variables). Panel data poses two major problems for DML: (1) DML uses a form of sample splitting called "cross-fitting", which relies on i.i.d. data and becomes complicated if another dimension (e.g., time) is present. (2) In linear panel data models, we can use fixed effects to eliminate time-constant confounding (e.g., Wooldridge, 2010), but only if the parametric model is correctly specified. By contrast, it is not straightforward where we can similarly handle unobserved heterogeneity within the DML algorithm, especially in the settings with nonlinear confounding influences, in which the original DML excels.

In this paper, our goal is to explore the potentials and problems that using DML for panel data can pose. We state the challenges, consider different potential solutions, evaluate them in a variety of simulations, clarify and discuss the necessary assumptions, and finally give recommendations for applied researchers when using DML in panel data settings. Our focus is on whether different point estimators can reliably recover a known true causal effect from panel data, potentially in the presence of unobserved heterogeneity (as opposed to deriving asymptotic properties and constructing variance estimators).

To preview our results, we find that violations of the independence assumption within the cross-fitting procedure are less consequential for the estimated coefficients than expected. However, many of the considered estimation methods struggle to remove the unobserved heterogeneity in settings with nonlinear observed confounding. For example, a seemingly natural approach of conducting DML on the time-demeaned variables (similar to fixed effects estimation) is strongly biased in settings with nonlinear confounding. We also show that the influence of the unobserved heterogeneity on the observed confounders has an impact on the accuracy of several methods. Nevertheless, an alternative approach that explicitly models the unobserved

heterogeneity in the ML models within DML, based on the correlated random effects approach (Mundlak, 1978), performs well across a variety of settings. Since explicitly modeling the unobserved heterogeneity involves the introduction of additional predictors in the ML models, this approach requires the sample size being large relative to the number of observed confounders.

The remainder of our paper proceeds as follows: Section 2 briefly reviews DML and traditional panel data methods, before discussing literature at the intersection of these research streams. Section 3 states the challenges when using DML for panel data and considers several solutions for each challenge. In Section 4, we assess these potential solutions on simulated data, generated from a variety of data-generating processes, and point out advantages and drawbacks of each method. Section 5 concludes with a discussion of the results and derives recommendations for using DML with panel data in practice.

3.2 Literature review

Our paper contributes to the literature by drawing together research around DML and the classic econometric literature around panel data, seeking to explore how these research streams can be mutually beneficial. We therefore first review the original DML method and textbook panel data methods. Then, we survey work that aims to use general ML methods with panel data, extends DML to similar settings, or directly tries to adapt DML to work with panel data.

DML (Chernozhukov et al., 2018) is a general estimation framework that allows using modern ML methods to flexibly adjust for observed confounding influences. Using flexible ML methods instead of a predetermined (e.g., linear) model helps to relax assumptions about variable selection and functional forms in settings where we observe all important confounders. DML does not directly address the problem of *unobserved* confounding (though it can help in instrumental variables settings to make the (conditional) exogeneity assumption of instruments more credible). We illustrate the intuition behind the DML algorithm for a data-generating process (DGP) based on the causal graph in Figure 3.1. Here, we want to estimate the causal effect of a treatment W_i on an outcome Y_i. However, confounders $\boldsymbol{X_i}$ influence both treatment and outcome and therefore bias the estimation if we do not adequately adjust for them. In practice, researchers often attempt to adjust for such confounders by including the corresponding variables linearly (or with another prespecified functional form) in a regression model. Yet the true way the confounders influence treatment and outcome (the functions $m_0()$ and $g_0()$) might be unknown

and potentially complex. This is where DML suggests modeling these influences with flexible ML methods, which are potentially capable of learning these relationships from the data. For this, the DML algorithm proceeds in five steps (here, illustrated for the partially linear model): (1) Randomly split the dataset into K folds, (2) hold out one fold, train two ML models on the remaining $K-1$ folds: first, predict treatment W_i from confounders $\boldsymbol{X_i}$; second, predict outcome Y_i from confounders $\boldsymbol{X_i}$, (3) make predictions for treatment and outcome from the estimated models, using the held out fold, and subtract them from the true values to obtain residuals, (4) use OLS to regress the outcome residual on the treatment residual and obtain the coefficient, (5) repeat steps (2)-(4) for each of the K folds, and average the resulting coefficients to get the final estimate.

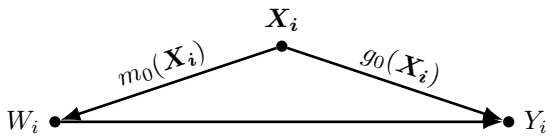


Figure 3.1: Causal graph for the assumed causal structure "unconfoundedness". W_i: treatment variable, Y_i: outcome variable, $\boldsymbol{X_i}$: observed confounding variables. The relationships between $\boldsymbol{X_i}$ and W ($m_0()$), and $\boldsymbol{X_i}$ and Y_i ($g_0()$), are potentially complex and nonlinear.

Chernozhukov et al. (2018) coin the term "cross-fitting" for the technique of splitting the data, doing training and effect estimation on separate samples, and repeating the procedure for all samples. The random splitting in the cross-fitting procedure relies on the observations being independent and identically distributed (i.i.d.) (Chernozhukov et al., 2018), which is one of the challenges arising for applications with clustered or panel data. We will further elaborate on this issue when we describe potential ways of combining DML with panel data.

On the other hand, a major emphasis in econometric research is finding methods that can estimate causal effects even in the presence of unobserved confounding. We can overcome one particular type of such omitted variable bias by exploiting the special structure of panel data – observing the same units repeatedly over time (e.g., Wooldridge, 2010). This type of data is very prevalent in many research fields, where the units could represent individuals, firms, products, cities, etc. If some unobserved confounders vary only between units, but not over time for the same unit, we can use panel data estimators (e.g., the fixed effects estimator) to eliminate any such time-constant unobserved heterogeneity (Wooldridge, 2010). By eliminating any time-constant confounding, we can relax the assumption of "no unobserved confounding" to "no time-variant unobserved confounding", which is considerably more plausible in many applications.

The fixed effects (FE) estimator works by either time-demeaning the data or by using a dummy variable regression (Wooldridge, 2012). In the time-demeaning approach, we subtract the time mean of each variable from the variable itself. Through this "within-transformation", we eliminate any (even unobserved) time-constant variables, since they are identical to their time means (Wooldridge, 2012). In the equivalent dummy variable approach, we simply include a binary/dummy variable for each unit in the regression. While this leads to a large number of explanatory variables (which can be problematic with small samples and many units), it gives the identical effect estimates as the time-demeaning approach in two-dimensional panels (Wooldridge, 2012). Transferring either approach to the DML framework is not trivial. One challenge is the decision at which step of the DML algorithm the time-demeaning or the inclusion of dummy variables should occur. The first option is "early", meaning we demean all variables before training the two ML models or include the dummy variables already in the training of the ML models. The alternative is "late", meaning that after computing the residuals in step (3), we time-demean the residuals before running the residual-on-residual regression, or we include the dummy variables directly in the final regression without demeaning first. We will discuss potential benefits and drawbacks of each of these approaches in Sections 3.3 and 3.4, but also present another option that mostly overcomes these challenges.

One lesser known alternative to the fixed effects estimator is the correlated random effects (CRE) approach (e.g., Wooldridge, 2010). Whereas on the one hand, a classical random effects approach assumes no correlation between the unobserved heterogeneity and the covariates, and on the other hand, the fixed effects approach does not restrict this relationship at all, the CRE framework unifies these approaches by explicitly modeling this correlation (Wooldridge, 2012). The initial suggestion by Mundlak (1978) was to let the unobserved heterogeneity be correlated with the *average* level of the covariates over time, which amounts to including the time means of each covariate in a pooled OLS or random effects regression (in addition to the original variables which vary by both time and unit). Later approaches by Chamberlain (1982, 1984) instead model the unobserved heterogeneity with a linear model of the covariates' time history. The attractive feature of these CRE approaches in our setting is that at least in the linear case, they give coefficient estimates identical to the FE approach (Wooldridge, 2012), while more explicitly modeling the unobserved heterogeneity instead of removing it, which could prove useful for the ML prediction steps.

In Table 3.1, we summarize some features of the standard DML, the fixed-effects estimator, and further developments related to DML that have explored panel data or similar settings. The first two rows show the previously described original DML framework and the traditional fixed effects estimator. The DML framework uses cross-fitting and allows for usage of any well-performing ML algorithm, but assumes a cross-sectional setting and does not consider any unobserved heterogeneity. On the other hand, the traditional fixed effects estimator and the correlated random effects approach can handle time-constant unobserved heterogeneity in panel data, whereas they do not use any ML algorithms (and no sample splitting) to make adjustment for time-variant variables more flexible. We use these two methods as a starting point and explore how the literature up to the time of writing this article (April 2024) has attempted to harmonize ML-based effect estimation with panel data settings.

Table 3.1: Methods at the intersection of DML and panel data

Method	**Reference**	**ML algorithms**	**Splitting procedure**	**Made for panel data**	**Considers unobs. het.**
DML	Chernozhukov et al. (2018)	Any	Standard cross-fitting	No	No
Fixed effects / CRE	Wooldridge (2010)	n.a.	None	Yes	Yes
Cluster-Lasso	Belloni et al. (2016)	Only lasso	None	Yes	Yes (early demeaning)
DML for Diff-in-diff	Chang (2020)	Any	Standard cross-fitting	Not really (only 2 periods)	No
Multiway DML	Chiang et al. (2022)	Any	Multiway cross-fitting	No (but clustered data)	No
(Debiased) orthogonal lasso	Semenova et al. (2023)	Only lasso	Neighbors-left-out cross-fitting	Yes	Yes (sparse FEs)
Extensions of within-group, first-difference, and CRE	Clarke and Polselli (2023)	Lasso, regression trees, random forests (any)	Split by unit in cross-fitting	Yes	Yes

First, Belloni et al. (2016) published a method for effect estimation in panel data settings with potentially high-dimensional time-varying confounders, even before the publication of DML in Chernozhukov et al. (2018). They treat the unobserved heterogeneity as fixed effects and remove them through demeaning all variables in a first step. Then, they use a variant of lasso that accounts for the clustering of units (Cluster-Lasso) to select important predictors in the treatment and in the outcome model, and use the selected variables as controls in a final OLS regression of the outcome on the treatment (Belloni et al., 2016). While lasso can learn the correct model under certain sparsity assumptions even without sample splitting, researchers

must manually decide which variable transformations to include in the algorithm, since the authors' method does not facilitate the use of more flexible ML methods such as random forests or neural networks (Chernozhukov et al., 2018).

Second, Chang (2020) extends the DML framework to the semiparametric difference-in-differences estimator, for situations where the parallel trends assumption only holds after adjusting flexibly for controls. While this setting does have a time dimension, the author only uses it to derive a time indicator for post-treatment, which indicates whether an observation comes from before or after reception of the (binary) treatment. The derived DML estimator for the average treatment effect on the treated (ATT) is very similar to the ATT estimator in Chernozhukov et al. (2018), but uses the difference in observed outcomes instead of the outcome itself (see also Chernozhukov et al., 2024). Moreover, Chang (2020) only considers time-constant confounders and therefore does not deal with unobserved unit-level heterogeneity. This setting is thus closer to the classical cross-sectional setting and does not encounter the problems we face when dealing with panel data in DML.

The third method also does not directly consider panel data but instead extends DML to multiway clustered data (Chiang et al., 2022). This setting is related to panel data in that the data is also double indexed, but instead of a time dimension there are two distinct unit/cluster dimensions (e.g., markets and products). To account for the clustered structure, Chiang et al. (2022) develop a novel multiway (K^2-fold) cross-fitting procedure that enables DML with any ML algorithm, even if the data is not i.i.d. but clustered along multiple dimensions. However, the authors' main goal is *valid inference* in the multiway clustered setting, they do not address identification by accounting for cluster-specific unobserved heterogeneity. Also, their method does not need to consider the unique challenges of the time dimension in cross-fitting.

Fourth, Semenova et al. (2023) develop methods for conditional average treatment effects in high-dimensional dynamic panel data settings. They build on the correlated random effects approach (Mundlak, 1978) and assume approximate sparse fixed effects (i.e., only few fixed effects are important/nonzero). Their "neighbors-left-out" splitting procedure adapts cross-fitting to support weakly dependent (panel) data. However, the authors rely on a version of lasso as the only possible ML algorithm for panel data and only hint at the potential of other ML methods in such a setting.

Finally, in an independent development parallel to ours, Clarke and Polselli (2023) develop DML estimators for panel data that account for unobserved individual heterogeneity in the par-

tially linear model. They implement their approaches with lasso, regression trees, and random forests, but can in principle incorporate any ML algorithm. For the cross-fitting, they split the data by unit, such that the full time series of each unit ends up in the same fold. Their first approach is an "approximation approach", where they first transform the raw data to remove the unobserved heterogeneity, and then use DML on the transformed data. The alternative approaches integrate the correlated random effects model by Mundlak (1978) in a DML framework. In simulations with nonlinear and discontinuous settings, Clarke and Polselli (2023) find superior performance of the latter methods built on CRE compared to standard linear methods or the approximation approach. However, they also demonstrate that DML with tree-based ML algorithms does not lead to valid inference in their settings.

Our study contributes to this literature in general and Clarke and Polselli (2023) specifically in four ways: (1) In addition to estimators similar to those in Clarke and Polselli (2023), we consider further intuitively appealing approaches for dealing with unobserved heterogeneity within DML; (2) we assess these approaches in a substantially wider variety of simulation settings that differ with respect to the true DGP, revealing, e.g., the particular sensitivity of the most promising approach to an increasing number of observed confounders; (3) we explicitly demonstrate and discuss the consequences of DGPs where the unobserved heterogeneity also influences the observed confounding, which is likely to occur in many applications, and (4) we investigate the impact of various distinct splitting strategies in the cross-fitting procedure on the estimated effects.

3.3 Possible methods for DML with panel data

In this section, we provide a detailed description of the problems that emerge when using DML for settings with panel data, and consider adaptations of DML to address these issues. We identify two problems: (1) the sample-splitting/cross-fitting procedure of DML with dependent data, and (2) accounting for potential unobserved heterogeneity in DML. For both problems, we explore a variety of possible solution approaches in the following.

3.3.1 Different splitting strategies for DML with panel data

One essential component of the original DML framework (Chernozhukov et al., 2018) is the cross-fitting procedure. This kind of sample splitting is important to remove overfitting bias

that can arise if we use the same observations for training the ML models and estimating the effects (Chernozhukov et al., 2018). To avoid this overfitting in DML, we train the ML methods on a part of the data, but make predictions and estimate the effects on another part that we did not use for training. In general, sample splitting leads to less efficient estimation, since we use only part of the data for training and estimation, respectively. Nevertheless, DML regains full efficiency by switching the roles of the training and estimation sample and averaging the resulting estimates ("cross-fitting") (Chernozhukov et al., 2018).

However, cross-fitting relies on the assumption that the observations are independent and identically distributed (i.i.d.) (Chernozhukov et al., 2018). As soon as we enter settings with panel data, this assumption is violated, because data points are dependent over time (i.e., serial correlation/autocorrelation) and/or within a cluster, and can end up in the same or in different samples when randomly splitting the data (Chiang et al., 2022). As a consequence, there is a danger of overfitting the ML models to the hold-out data. While this certainly makes consistency statements, asymptotic analysis, and valid confidence intervals difficult (e.g., Wooldridge, 2010), it is unclear how severe the practical consequences for the estimated effect coefficients really are. In our study, we will assess how different splitting strategies affect the finite-sample performance of different DML estimators. We leave the analysis of asymptotic properties and the construction of valid confidence intervals to future research.

When implementing cross-fitting for panel data, there are a variety of options for how to split the sample (Table 3.2). First, we could do the standard random split as in the original DML algorithm for the cross-sectional setting, ignoring the special panel structure. This can potentially lead to the overfitting bias previously described.

Table 3.2: Different approaches to sample splitting when using DML with panel or clustered data

Splitting strategy	**Description**	**Problems**
Random	Random splitting as in classic DML, ignoring panel structure	Ignores panel structure and dependency within units/across time
By unit	All observations of the same *unit* end up in the same fold, otherwise random	ML methods cannot predict unit-specific effects in the hold-out fold; time dependence potentially still present
By time (period)	All observations of the same *period* end up in the same fold, otherwise random	Time dependence of period-adjacent observations; cluster dependence potentially still present
By time (folds)	T/K *adjacent* periods end up in the same fold	Time dependence at the splitting point of adjacent folds; cluster dependence potentially still present
By time (neighbors-left-out)	T/K *adjacent* periods end up in the same fold, folds adjacent to the prediction fold are excluded from training	Cluster dependence potentially still present

Secondly, we could split the data by the unit dimension and ensure that observations of each unit (index i) end up in only the training or the estimation sample at any point (as done in, e.g., Clarke and Polselli (2023)). While this ensures independence between the training and estimation folds in the unit dimension, it becomes problematic if we expect the ML methods to also model unit-specific effects (Semenova et al., 2023). That is, we cannot model unit-specific unobserved heterogeneity by, e.g., including unit dummies as predictors in the ML algorithm, since the units in the hold-out sample (used for prediction and estimation of effects) are not present in the training sample. Thus, when splitting by unit, we cannot remove the unobserved heterogeneity with such an approach.

Alternatively to splitting by unit, we could split along the other dimension, i.e., by time. Here we consider three different options. These options have in common that they do not address the potential cluster dependence along the unit dimension. First, we can ensure that all observations of the same period (index t) end up in the same sample. However, this does not help in the case of serial correlation, since the adjacent observations in the time dimension can still end up in the other sample and thus induce a dependence.

The fourth splitting strategy improves upon the previous one by splitting the time-ordered data into K folds. That is, observations with indices from $t = 1$ to $t = T/K$ end up in the first fold, from $t = T/K + 1$ to $t = 2T/K$ in the second fold, etc. This procedure reduces the time dependence to only be substantial at the splitting point of adjacent folds: The last observation of the first fold and the first observation of the second fold might be correlated, but the further one moves from the splitting point, the smaller the correlation gets.

To also eliminate the correlations around the splitting points, Semenova et al. (2023) propose a strategy they call "neighbors-left-out cross-fitting". They suggest dividing the data into a relatively large number of time-adjacent folds ($K \geq 10$), of which we hold out not only the fold used for prediction and estimation, but also the folds in its immediate neighborhood. By this, the training and the estimation samples should be approximately independent, even in weakly dependent time series or panel data. See Appendix B.1.1 for a graphical illustration of this and the previous cross-fitting approach.

3.3.2 Accounting for unobserved heterogeneity in DML

The second challenge when using DML with panel data is accounting for unobserved heterogeneity. One of the main motivations for using panel data in the first place is to account for

unobserved influences that only vary along one dimension and are constant in the other (e.g., Wooldridge, 2010, Chapter 10). This can, for example, be a unit-specific but time-constant unobserved variable such as U_i in Figure 3.2 or in the partially linear model of Equations 3.1 and 3.2. If this variable influences both treatment and outcome, it acts as an unobserved confounder and leads to an omitted variables bias if we employ standard cross-sectional methods (Wooldridge, 2010). We will later consider three different DGPs that differ with respect to which variables the unobserved heterogeneity influences (see Figure 3.2). However, since we typically do not know the true DGP in practice, the ideal method should perform well in each of these settings.

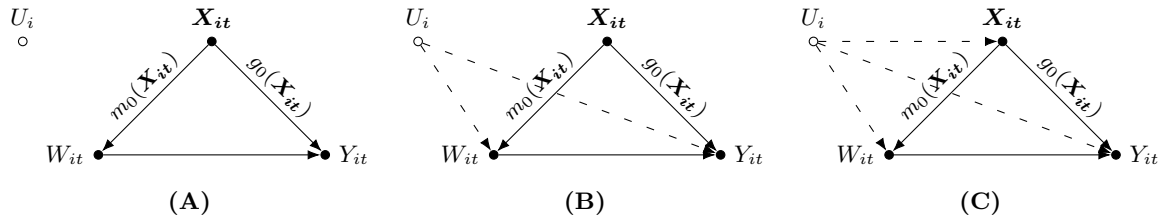


Figure 3.2: Possible DGPs for panel data settings. W_{it} (treatment), Y_{it} (outcome), and X_{it} (observed confounders) vary across both units and time. U_i is unobserved unit-specific and time-constant heterogeneity. We consider three causal structures: **(A)** U_i does not influence any other variables (or does not exist), **(B)** U_i only influences W_{it} and Y_{it}, **(C)** U_i additionally influences X_{it}.

$$Y_{it} = \alpha_1 + \beta W_{it} + \gamma g_0(\boldsymbol{X_{it}}) + \delta U_i + \mu_{it} \tag{3.1}$$

$$W_{it} = \alpha_2 + \gamma m_0(\boldsymbol{X_{it}}) + \delta U_i + \eta_{it} \tag{3.2}$$

In traditional econometrics, the most common way of dealing with unobserved heterogeneity in panel data is fixed effects estimation (e.g., Wooldridge, 2010, Chapter 10). If there is unobserved confounding which varies only along one cluster dimension, using fixed effects can eliminate its biasing influence (Wooldridge, 2012, Chapter 14). Traditionally, we implement fixed effects estimation by time-demeaning all variables and running OLS on the transformed variables. Since the unobserved heterogeneity is fixed over time, it disappears from the demeaned equation (Wooldridge, 2012). Alternatively, we can run an OLS regression where we include a dummy/binary variable for each unit, which leads to the same estimates as fixed effects estimation in the standard setting (Wooldridge, 2012).

However, if we want to use DML to flexibly adjust for observed confounding in the presence of unobserved heterogeneity, it is not obvious where and how in the algorithm we can consider fixed effects (or alternative ways of accounting for unobserved heterogeneity). Should we demean

the variables before the ML predictions or demean the residuals afterwards? Can we consider the fixed effects directly within the ML step? How? In the following, we discuss several conceivable approaches for this setting (Table 3.3).

First, we could ignore the potential for unobserved cluster-level confounding and simply use the original DML algorithm on the pooled data ("Pooled DML"). This is only appropriate if there is no unobserved heterogeneity; if this assumption is violated, the estimates will be biased.

Table 3.3: Different approaches to fixed effects within DML

Approach	**Description**
Pooled DML	Ignores the panel dimension and unobserved heterogeneity, does standard DML on full data
Early demeaning	First demeans all variables, then runs standard DML on the demeaned data
Late demeaning	First runs steps (1)-(3) of DML on original data, then demeans the residuals before running OLS
DML with dummies	Runs standard DML, but adds a dummy for each unit in the prediction steps
DML with CRE	Runs standard DML, but adds time means for treatment and confounders in the prediction steps as in correlated random effects

The second and third approach are inspired by the traditional fixed effects estimation and deal with the unobserved heterogeneity by introducing an additional step in the DML algorithm. In what we call "early demeaning", we try to eliminate the unobserved heterogeneity *before* DML. That is, we demean all variables first, then perform DML on the demeaned variables. If we fully eliminate the unobserved heterogeneity by this fixed effects transformation, the prediction tasks in DML should be easier, since the ML methods only need to model observed variation. The "within-transformed" outcome and treatment equations for the ML training are

$$\begin{aligned} y_{it} - \bar{y_i} &= \gamma g_0(x_{it} - \bar{x_i}) + \mu_{it} - \bar{\mu_i} \\ w_{it} - \bar{w_i} &= \delta m_0(x_{it} - \bar{x_i}) + \eta_{it} - \bar{\eta_i}, \end{aligned} \tag{3.3}$$

where $\bar{w_i}$ is the mean treatment over time ($\frac{1}{T}\sum_{t=1}^{T} w_{it}$). This early demeaning is only appropriate if the time mean is additively separable in the true DGP, which holds in the linear case, but not necessarily if $g_0()$ or $m_0()$ are nonlinear, as we will show below. This approach is similar to the "approximate approach" in Clarke and Polselli (2023).

In the third approach, we deal with the fixed effects *later* in the DML algorithm ("late demeaning"). Here, we run steps (1)-(3) of DML first (ML and residualization), before we

demean the residuals in the final DML step and regress the demeaned outcome residual on the demeaned treatment residual:

$$v_{it}^{y} - \bar{v_i^{y}} = v_{it}^{w} - \bar{v_i^{w}} + \epsilon_{it} - \bar{\epsilon_i}. \tag{3.4}$$

This late demeaning increases the difficulty for the ML prediction tasks early in the algorithm, since the unobserved heterogeneity is still present and acts as additional noise. However, if the ML method still manages to successfully model the observed confounding, the remaining unobserved heterogeneity may be additively separable afterwards.

In contrast to the previous two approaches, the final two strategies try to directly model the unobserved heterogeneity within the two ML models in DML. Instead of an additional step in the algorithm, these approaches change the predictors/features supplied to the ML methods. Approach four ("DML with dummies") includes unit dummies in the prediction steps of DML, inspired by the equivalence of the dummy variable regression to fixed effects estimation in the standard linear setting (Wooldridge, 2012). That is, the outcome and treatment equations for the ML training are

$$\begin{aligned} y_{it} &= \gamma g_0(x_{it}, z1_i, ..., zN_i) + \mu_{it} \\ w_{it} &= \delta m_0(x_{it}, z1_i, ..., zN_i) + \eta_{it}, \end{aligned} \tag{3.5}$$

where $z1_i$ to zN_i are dummy variables for each unit, which are 1 if the observation belongs to that unit and 0 if it does not. This should work well in settings where there are relatively few units, or in settings with high-dimensional fixed effects (i.e., many units), if we can assume that only few of these are important ("sparse" fixed effects). This sparsity assumption is common in the literature, but often not realistic in settings with unit-specific unobserved heterogeneity (e.g., Belloni et al., 2016). If the assumption is violated, the prediction tasks will likely become too complex if the number of observations is not significantly larger than the number of relevant fixed effects.

The final approach ("DML with CRE") avoids this sparsity assumption about the fixed effects by explicitly modeling the relationship between the unobserved heterogeneity and the covariates as proposed in the correlated random effects approach (e.g., Mundlak, 1978). Chernozhukov et al. (2022b) mention a similar DML approach for panel data in the application of their novel automatic debiased ML framework, and Clarke and Polselli (2023) also introduce a related CRE estimator. In the context of DML, using Mundlak-type correlated random effects amounts to

including the treatment and covariate time means in addition to the time-varying covariates into the ML predictions for both the outcome and the treatment:

$$\begin{aligned} y_{it} &= \gamma g_0(x_{it}, \bar{x}_i, \bar{w}_i) + \mu_{it} \\ w_{it} &= \delta m_0(x_{it}, \bar{x}_i, \bar{w}_i) + \eta_{it}. \end{aligned} \tag{3.6}$$

This way, we can model the time-constant unobserved heterogeneity without introducing a large number of additional variables. However, if J is the number of covariates, the number of predictors in the ML models using the CRE approach increases by a factor of $2 * J$, since we introduce the time mean for each additional variable as well. By comparison, all other approaches only scale by the factor J.

3.4 Simulations

We now explore how the suggested methods perform on simulated data. First, we give an overview of how we implemented the considered DML methods and introduce alternative traditional statistical methods as benchmarks. Then, we describe our baseline DGPs, which we subsequently use to compare different cross-fitting techniques, as well as different estimation methods. Finally, we extend our baseline simulations and change different characteristics of the DGPs to investigate the methods' sensitivity to these modified situations.[1]

3.4.1 Method implementations

In addition to the various DML approaches described in the previous section, we also implemented several traditional statistical methods as comparison baselines (Table 3.4). The first method is a naive simple OLS regression, where we regress the outcome on the treatment but adjust neither for observed covariates nor for unobserved heterogeneity. As the second method, we also use an OLS regression of the outcome on the treatment, but now include all observed covariates linearly, i.e., pooled OLS (POLS) (Wooldridge, 2010). This method will be biased and inconsistent if there is any unobserved heterogeneity and/or if the confounding influence is not linear. Thirdly, we use the standard fixed effects estimator, where we adjust for all covariates linearly and try to adjust for the unobserved heterogeneity by including fixed effects.

[1] All code for method implementation, data generation, and estimation is available on OSF: `https://osf.io/8skxu/?view_only=1d56c1e412084ee399cd9a3fdcb39c02`.

Table 3.4: Description of implemented methods

Label	**Description**
Simple OLS	Linear regression ignoring all covariates and unobserved heterogeneity
POLS	Pooled OLS: linear regression with all covariates but not dealing with unobserved heterogeneity
Fixed effects	Linear regression with all covariates and accounting for unobserved heterogeneity with fixed effects
PDML	Pooled DML: ignoring panel dimension and unobserved heterogeneity, using XGBoost as predictive algorithm
DML (early FE)	Early demeaning: using standard DML with XGBoost as predictive algorithm on demeaned data
DML (late FE)	Late demeaning: running steps (1)-(3) of standard DML with XGBoost as predictive algorithm first, then demeaning residuals in step (4) before OLS
DML (dummies)	Standard DML with XGBoost as predictive algorithm, but adding a dummy for each unit in the prediction models
DML (CRE)	DML with correlated random effects, adding time means for treatment and confounders in the prediction models
Oracle FE	Standard fixed effects estimation, always knowing the true form of the confounding (infeasible)

Approaches 4-8 are the various DML implementations for panel data discussed in Table 3.3. As the predictive ML algorithm(s) in step (3) of DML, we use boosted trees as implemented in XGBoost. In our implementation, we use default values for the learning rate eta (0.3) and the maximum tree depth (6), and employ early stopping if the validation set performance does not improve for 10 rounds. We tune the optimal maximum number of boosting iterations by choosing from up to 200 rounds with 5-fold cross-validation. We experimented with using other flexible ML methods within DML, which lead to very similar results. We choose XGBoost as representative of flexible ML algorithms due to its strong performance within DML in cross-sectional settings (see Chapter 2), as well as its computational efficiency compared to other flexible methods (Chen et al., 2023). One noticeable feature of the "DML (dummies)" method is that generating unit dummies potentially leads to a very large number of variables. In settings with many units, this not only complicates the prediction task, but also substantially increases the computation time for this approach compared to alternatives. In Appendix B.2, we show timing results for various combinations of numbers of units and periods. In our baseline simulation with 500 units and 10 periods, computing DML with dummies for one dataset takes about 330 seconds, whereas the second slowest method (DML with CRE) is computed within less than 8 seconds.

Finally, we include an infeasible "oracle FE" method as an additional benchmark. Here we use the standard fixed effects framework, but always in combination with a parametric model

that uses the correct (in practice unknown) functional form of the covariates. This method indicates whether researchers could in theory estimate the true effect from the data if they knew the true functional form of the confounding, i.e., specified the correct parametric model.

3.4.2 Baseline data generation

For all simulations, we follow one of the causal graphs in Figure 3.2. We are interested in the effect of a treatment variable W_{it} (e.g., price) on an outcome variable Y_{it} (e.g., demand), so we need to adjust for the observed confounder(s) X_{it} (e.g., advertising), which influence both treatment and outcome. We assume and simulate a constant and homogeneous treatment effect throughout. Observed confounders, treatment and outcome can vary across multiple dimensions (e.g., both unit and time) and are therefore double-indexed. Furthermore, there can exist some form of unobserved heterogeneity (e.g., store characteristics such as management quality) between different units. We model this as an unobserved variable U_i, which varies across only one dimension (here, the unit dimension). In our baseline simulation settings, we differentiate between three causal structures that differ in how U_i influences the other variables: (A) U_i does not exist, or at least exerts no influence on the other variables. In this setting, we have no unobserved heterogeneity, hence flexibly adjusting for X_{it} should suffice. (B) U_i does exist, but influences only the treatment and the outcome directly, not the observed confounders X_{it}. (C) In addition to treatment and outcome, U_i also influences the observed confounders X_{it} and thus impacts treatment and outcome via multiple pathways. In our example, we consider structure (C) to be the most plausible, since the unobserved management quality certainly also influences decisions on advertising and promotions.

In the baseline simulation, we only consider a single variable for each of U_i, X_{it}, W_{it}, and Y_{it}. In all simulations, we draw single exogenous variables from a standard normal distribution ($N(0,1)$). If the variable is clustered and time-constant like U_i, we draw the values on the unit-level and replicate the same value across all time periods for a given unit. We generate all other variables according to Equations 3.7-3.9, where the inclusion of U_i in each equation depends on whether we simulate the causal structure (A), (B), or (C). Intercepts and noise terms follow a standard normal distribution ($\alpha, \epsilon_{it}, \eta_{it}, \mu_{it} \sim N(0,1)$). $g_0()$ and $m_0()$ indicate the functional form of the *observed* confounding. In our baseline simulations, we use either a linear functional form ($g_0(X_{it}) = m_0(X_{it}) = X_{it}$), for which linear methods are appropriate, or a nonlinear u-shaped functional form ($g_0(X_{it}) = m_0(X_{it}) = {X_{it}}^2$), for which linear methods

are misspecified, but for which flexible ML methods might be capable of learning the correct functional form.[2] We draw the coefficients for the influence of the observed confounders (γ) and for the unobserved confounders (δ) for each simulation from a standard normal distribution ($\gamma, \delta \sim N(0,1)$). We set the true causal effect of interest to 1 ($\beta = 1$). The main goal of the simulations is to investigate how well different methods can recover this coefficient across various settings. We specify the number of units N and the number of periods T in each of the following result sections.

$$X_{it} = \alpha_0 + \delta U_i + \epsilon_{it} \tag{3.7}$$

$$W_{it} = \alpha_1 + \gamma g_0(X_{it}) + \delta U_i + \eta_{it} \tag{3.8}$$

$$Y_{it} = \alpha_2 + \beta W_{it} + \gamma m_0(X_{it}) + \delta U_i + \mu_{it} \tag{3.9}$$

3.4.3 Comparison of cross-fitting techniques

Before we compare the performance of the considered estimators, we investigate how the different cross-fitting techniques introduced in Section 3.3.1 affect the coefficient estimates. For this, we simulate data according to the baseline DGP with the most complex causal structure (C) and u-shaped confounding influences. The dataset is a balanced panel with $N = 100$ units across $T = 50$ periods. We choose this relatively large number of periods to still have multiple periods in the hold-out fold when splitting the data into time-adjacent folds. We violate the original cross-fitting assumption of i.i.d. data both with the presence of unobserved unit heterogeneity and with a relatively large degree of autocorrelation. We introduce autocorrelation by changing the DGP of the outcome model (Equation 3.9) to include serially correlated errors: Now, we simulate μ_{it} as a weakly dependent time series according to an AR(1) model, i.e., $\mu_{it} = \rho\mu_{it-1} + e_{it}$, with the ar-coefficient $\rho = 0.9$. If the cross-fitting procedure affects the coefficient estimates, it should be especially visible when violating the independence assumption. For each considered DML estimator, we implement all of the cross-fitting techniques from Table 3.2.

In describing the results, we only compare the performance of different cross-fitting techniques *within* the same DML estimator. In the next section, we will further describe and interpret the differences in performance between the different estimators. Our simulation results display a surprisingly small influence of the choice of cross-fitting technique on the estimated

[2] We experimented with other nonlinear functional forms (e.g., a discontinuous step function), which led to very similar results.

coefficients (Figure 3.3), except for some special cases. Within most estimation methods, the different splitting strategies lead to very similar estimated effects. One notable exception is cross-fitting when splitting by unit: As anticipated by Semenova et al. (2023), this results in substantial bias if we expect the ML methods to model the unobserved heterogeneity in the hold-out fold, while only observing the units present in the training folds. For this splitting strategy, using unit dummies within DML (Figure 3.3B) leads to heavily biased effect estimates, because the dummy variables in the hold-out data belong to different units than the dummy variables used for training. To a (much) weaker degree, we make similar observations for DML with CRE (Figure 3.3A) and pooled DML (Figure 3.3D), though the issues become more pronounced in settings with very few units and many periods (e.g., $N = 10$ and $T = 500$, see Figure B.3 in Appendix B.1.2).

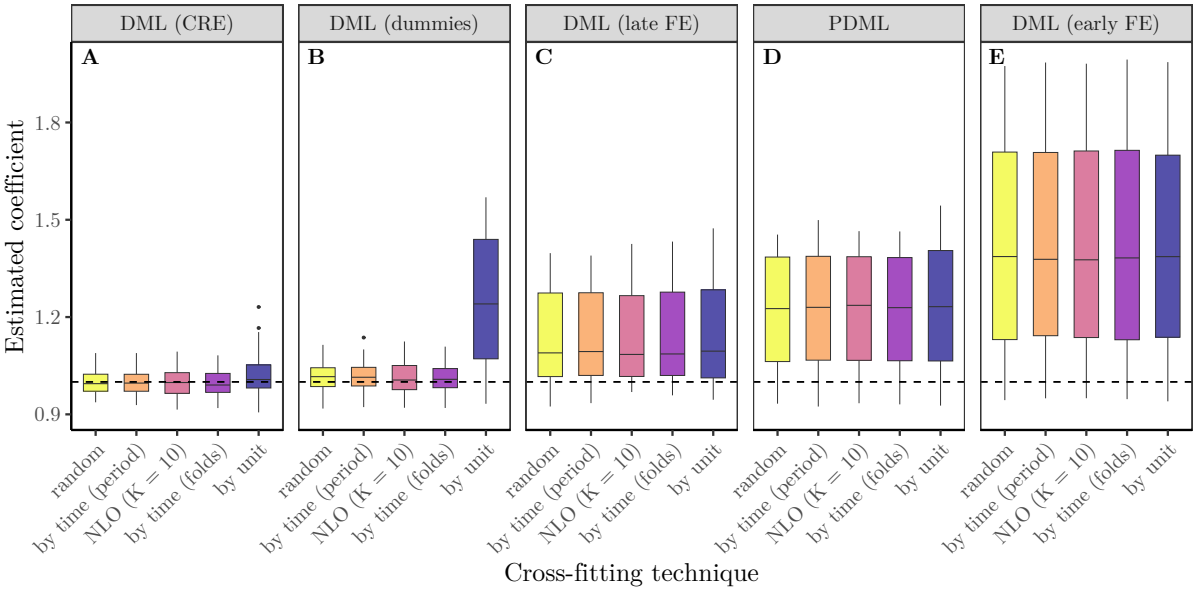


Figure 3.3: Results for utilizing different cross-fitting techniques (Table 3.2) within various DML estimators. The vertical axis depicts the estimated coefficient. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method. Data is generated according to DGP (C), with one observed confounder, u-shaped functional forms and a large degree of autocorrelation ($\rho = 0.9$). NLO: neighbors-left-out cross-fitting.

From these observations, we conclude that we should not split by unit when cross-fitting, at least if unobserved heterogeneity could be plausibly present in our specific application. While there is theoretical basis for ensuring independence of clustered data, especially for constructing valid confidence intervals (e.g., Chiang et al., 2022), failing to adequately model the unobserved heterogeneity will likely lead to a more substantial bias in the estimated effects. Besides splitting by unit, the results of the other cross-fitting techniques are hardly distinguishable. Therefore,

in the absence of a strong argument for a specific cross-fitting technique, we will proceed with the rest of our simulations using the "random" sample splitting method.

3.4.4 Comparison of estimation methods

We now compare the different estimation methods in six different baseline settings that differ in the causal structure and the functional form of the observed confounding. In all baseline simulations, we use $N = 500$ different units across $T = 10$ different periods. For now, we assume no autocorrelation of the error terms.

The plots in the first column (Figure 3.4A, C, and E) show results from simulations with a linear influence of the observed confounders X_{it}, whereas the results in the second column (Figure 3.4B, D, and F) originate from a nonlinear, u-shaped functional form. The DGPs of the first row follow the causal structure (A), where we simulate no unobserved heterogeneity. By contrast, in the second row simulations, the unobserved heterogeneity U_i influences only the treatment W_{it} and the outcome Y_{it}, but not the observed confounders X_{it} (causal structure (B)). The third row contains results from causal structure (C), where the unobserved heterogeneity additionally influences X_{it}. In terms of complexity, the simulation settings become more challenging from the top left to the bottom right panel.

Across simulation settings, the naive simple linear regression performs worst, since it neither adjusts for potential unobserved heterogeneity nor for observed confounding. By contrast, the (infeasible) fixed effects regression with the oracle for the functional form always delivers unbiased and precise estimates.

In the simplest setting with no unobserved heterogeneity and linear confounding influence (Figure 3.4A), all methods except for the naive regression perform well and give unbiased estimates. If we instead simulate u-shaped confounding relationships of the observed covariates X_{it} (Figure 3.4B), several methods become strongly biased. While this is not surprising for the linear methods (with and without fixed effects), the DML variant with early demeaning also incurs substantial bias. This is because W_{it} and Y_{it} are nonlinear in X_{it}, not in $\ddot{X}_{it} = (X_{it} - \bar{X}_i)$, and the time means are not additively separable (i.e., $g_0(X_{it} - \bar{X}_i) \neq g_0(X_{it}) - \bar{X}_i$). Hence, after demeaning, $\ddot{X}_{it}$ might not be sufficient to predict the treatment and outcome, respectively. This is especially problematic in a large N, small T setting like our baseline: After removing the between-variation along the unit dimension (N), there is only little variation left along the time dimension (T) to model the nonlinear functional form of the observed confounding.

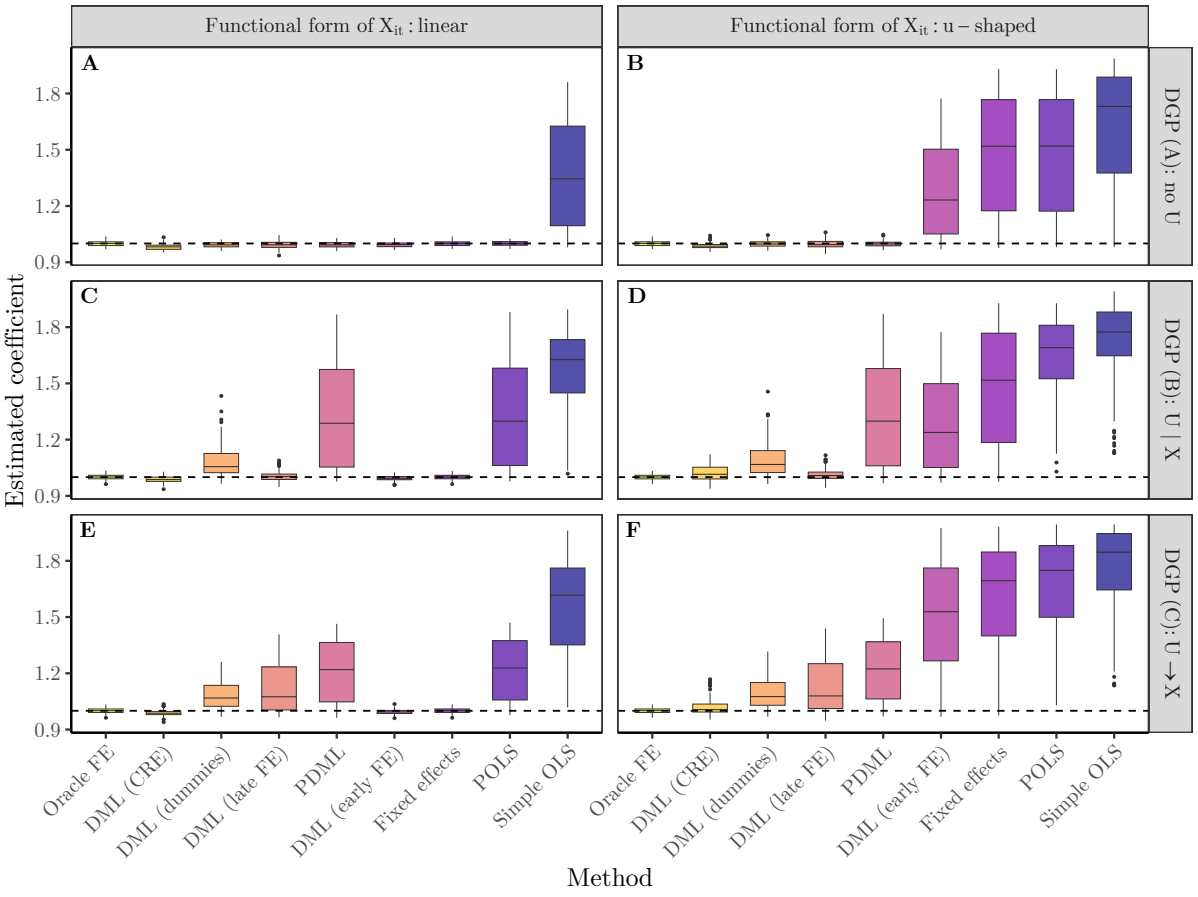


Figure 3.4: Results for our baseline simulation with $N = 500$ units and $T = 10$ periods. The horizontal axis displays the different methods from Table 3.4. The vertical axis depicts the estimated coefficient. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method. The three rows contain three different DGPs: "$no\,U$" indicates no unobserved heterogeneity, "$U \mid X$" means the unobserved heterogeneity influences treatment and outcome, but not confounders, and "$U \rightarrow X$" means the unobserved heterogeneity also influences the confounders.

In the case of unobserved heterogeneity influencing treatment and outcome in addition to *linear* observed confounding (Figure 3.4C), we see the importance of using some sort of fixed effects estimator. Now methods such as the linear regression and pooled DML are biased, since they only adjust for the observed confounding and not for the unobserved heterogeneity. While DML with dummies performs much better, it is also not quite unbiased, because the large number of non-sparse unit dummies ($z1_i, ..., z500_i$) makes the predictions challenging, and likely leads to variable selection mistakes. All other methods deliver precise and unbiased estimates. Moving to nonlinear confounding within the same causal structure (Figure 3.4D), the linear fixed effects estimator and DML with early demeaning become biased as well. DML with dummies is similarly biased as in the previous setting. Of the feasible methods, only DML with correlated random effects and DML with late demeaning are close to unbiased, with the latter

being slightly more precise. Since U_i does not influence X_{it}, the unobserved heterogeneity is additively separable after modeling the observed nonlinear confounding influence, which is how the late demeaning method operates.

In the most complex causal structure considered, the unobserved heterogeneity also influences the observed confounders X_{it}, and thus the treatment and outcome via this second indirect path as well. In this setting with linear confounding influences, the extent of the bias for pooled OLS and pooled DML is somewhat attenuated (Figure 3.4E). While these methods had no information at all about the unobserved confounding in the causal structure (B), they now can learn something about U_i, since it is part of X_{it} via the path $U_i \rightarrow X_{it}$. For these methods, the influence of U_i on X_{it} is beneficial, since they can to some extent exploit that observed confounders now contain information about the unobserved heterogeneity. On the other hand, compared to the previous causal structure (no influence $U_i \rightarrow X_{it}$), the late demeaning method now incurs bias. In the $U_i \rightarrow X_{it}$ setting, this method "overfits" treatment and outcome by modeling their relationship with U_i twice. First, by predicting these variables from X_{it}, which now also contains information about U_i. Second, by demeaning the residuals after the predictions. Both residualization and demeaning try to remove the same confounding variation induced by U_i. Although they only remove this variation imperfectly (see, e.g., pooled OLS and pooled DML), trying to remove it twice leads to unintentionally removing exogenous variation that we need for estimating the effect of W on Y from the residuals, hence causing a biased estimate.

The final and most challenging setting follows the same causal structure, but uses a nonlinear influence of the observed confounders (Figure 3.4F). As in Panel E, the late demeaning method again cannot deliver unbiased estimates. The only feasible method without substantial bias is DML with correlated random effects, followed by DML with dummies, which incurs a similar bias as in the previous three settings. Compared to Panel D, pooled DML is less biased, since it (partially) learns about U_i through X_{it}.

Across settings, only DML with correlated random effects consistently delivers estimates that are close to the true effect. If the influence of the observed confounders is linear, standard fixed effects estimation is appropriate and sufficient, but it fails if that influence is different from the one specified in the fixed effects model (e.g., nonlinear instead of linear). If we can rule out an influence of the unobserved heterogeneity on the observed confounders, DML with late demeaning gives very accurate estimates. However, since we rarely can rule out that influence

with certainty in practice (or instead have reason to believe that it exists, like in our example), our simulations suggest that DML with correlated random effects is most robust to any of the considered baseline settings.

3.4.5 Simulation extensions

From the initial baseline, we vary different characteristics of the DGP to explore how the considered methods perform in a variety of settings.

Changing the panel dimensions N/ T

First, we vary the ratio of the number of units (N) to the number of time periods (T), while keeping the overall number of observations constant. In addition to the baseline setting [$N = 500$, $T = 10$], we also consider the combinations [$N = 100$, $T = 50$], [$N = 50$, $T = 100$], and [$N = 10$, $T = 500$]. Since we currently only simulate unobserved heterogeneity on the unit dimension and not on the time dimension, we expect the smaller numbers of units (and thus decreased dimensionality) to benefit the DML with dummies method most of all.

We report results for the setting with $N = 10$ units and $T = 500$ periods (Figure 3.5). In comparison to our baseline (Figure 3.4), the estimates of two particular methods change substantially. First, using DML with dummy variables now results in virtually unbiased estimates. The ML algorithm only has to handle 10 dummy variables (one for each unit) instead of the 500 dummies in the baseline scenario. This easier task does not result in variable selection mistakes, and thus the method estimates the effect almost as precisely as DML with correlated random effects. Second, DML with early demeaning now also delivers precise estimates in the less complex nonlinear settings (Figure 3.5B and D), but still not for the complex setting in Panel F. As mentioned above, the early demeaning removes unit-specific between-variation, which the ML method subsequently cannot use for the prediction task. However, in the absence of $U_i \rightarrow X_{it}$, this is only consequential if the between-variation is crucial for modeling the confounding relationships. In the current small N, large T setting, the within-variation still consists of 500 observations per unit, which is sufficient for modeling the nonlinear functional forms. In the more complex setting (Panel F), the early demeaning still cannot fully remove the unobserved heterogeneity, since the time means are not additively separable due to the nonlinear function: $g_0(X_{it} - \bar{X}_i) \neq g_0(X_{it}) - \bar{X}_i$. The performance of the remaining methods is very similar to the baseline setting.

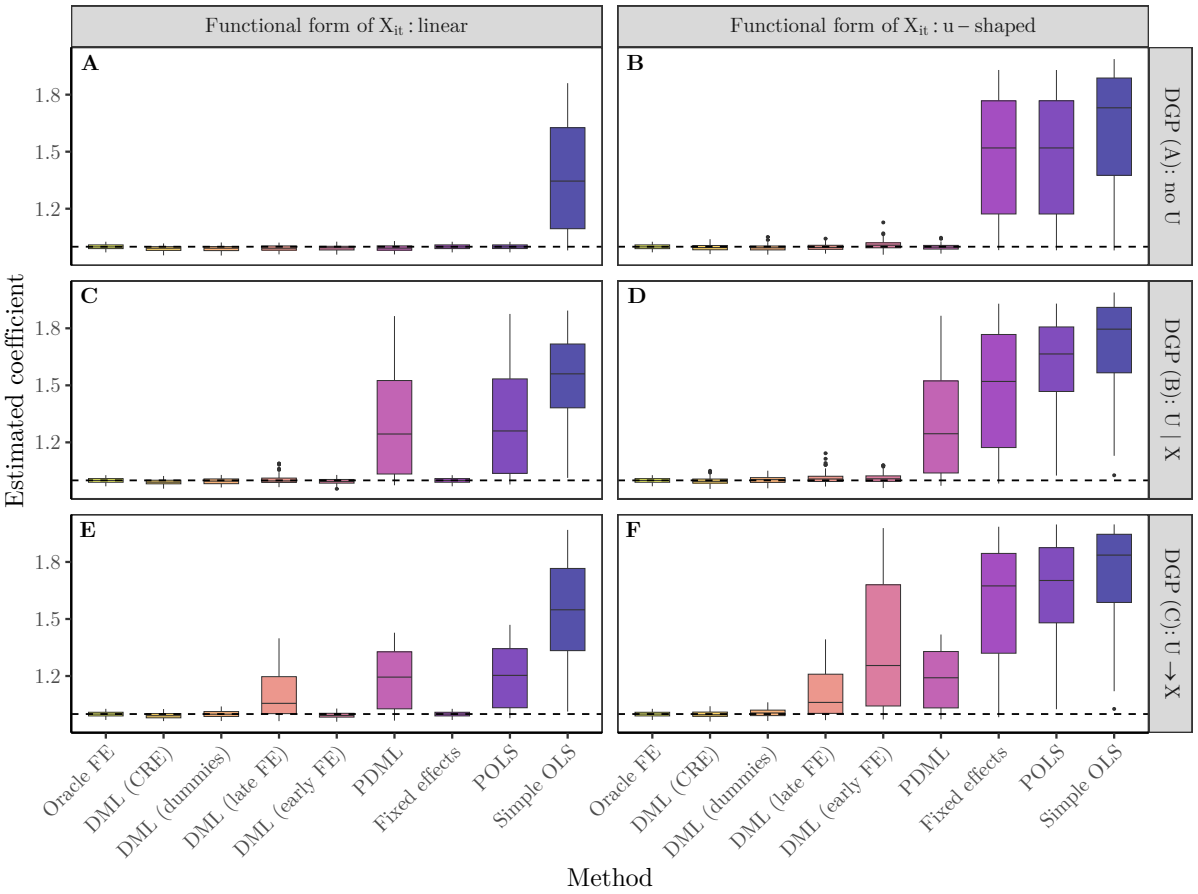


Figure 3.5: Results for the setting with $N = 10$ units and $T = 500$ periods. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method. The three rows contain three different DGPs: "$no\,U$" indicates no unobserved heterogeneity, "$U \mid X$" means the unobserved heterogeneity influences treatment and outcome, but not confounders, and "$U \rightarrow X$" means the unobserved heterogeneity also influences the confounders.

The settings with $[N = 100,\ T = 50]$ and $[N = 50,\ T = 100]$ lead to results between the baseline and the $[N = 10,\ T = 500]$ setting, such that DML with dummies and DML with early demeaning decline in accuracy as N increases and T decreases (see Appendix B.3.1).

Increasing the number of observed confounders

In the second extension, we test how the different methods scale for larger numbers of observed confounders $\boldsymbol{X_{it}}$. We focus on causal structure (C) with nonlinear confounding and generate X_{jit} for $j = 1, ..., J$ confounders according to Equation 3.10, where we draw ϵ_{jit} from a multivariate normal distribution with a mean of zero and a randomly generated covariance matrix (i.e., $\epsilon_{jit} \sim N(0, \Sigma)$, $\Sigma = A'A$, with $A \sim N(0, 1)$):

$$X_{jit} = \alpha_{0j} + \delta U_i + \epsilon_{jit}. \tag{3.10}$$

Also, we now draw separate confounding coefficients γ_j for each confounder and divide each by the overall number of confounders J (Equations 3.11 and 3.12). In doing so, we ensure that on average, the overall strength of the confounding influence is similar to the baseline scenario, such that we are only varying the number of confounders.

$$W_{it} = \alpha_1 + \sum_{j=1}^{J} \frac{\gamma_j}{J} g_0(X_{jit}) + \delta U_i + \eta_{it} \tag{3.11}$$

$$Y_{it} = \alpha_2 + \beta W_{it} + \sum_{j=1}^{J} \frac{\gamma_j}{J} m_0(X_{jit}) + \delta U_i + \mu_{it} \tag{3.12}$$

For this setting and the next, we introduce two new methods as baselines to facilitate the interpretation of the results. First, the method "FE only" does not adjust for the observed confounding ($\boldsymbol{X_{it}}$) at all, but only aims to account for unobserved heterogeneity by using fixed effects. This method allows us to determine how important directly adjusting for the unobserved heterogeneity is. Second, the (infeasible) method "Oracle w/o FE" adjusts for the observed confounding by knowing the "true" functional form, but does not account for the unobserved heterogeneity, thereby demonstrating the significance of adjusting for the $\boldsymbol{X_{it}}$ only. Finally, the additional benefit of also adjusting for the unobserved heterogeneity becomes visible in the previously described "Oracle FE" method.

We display the resulting mean absolute error (MAE) in the causal coefficient for each method for between 1 and 10 observed confounders (Figure 3.6). We first observe that the (infeasible) oracle methods are barely affected by changing the number of confounders: The full oracle

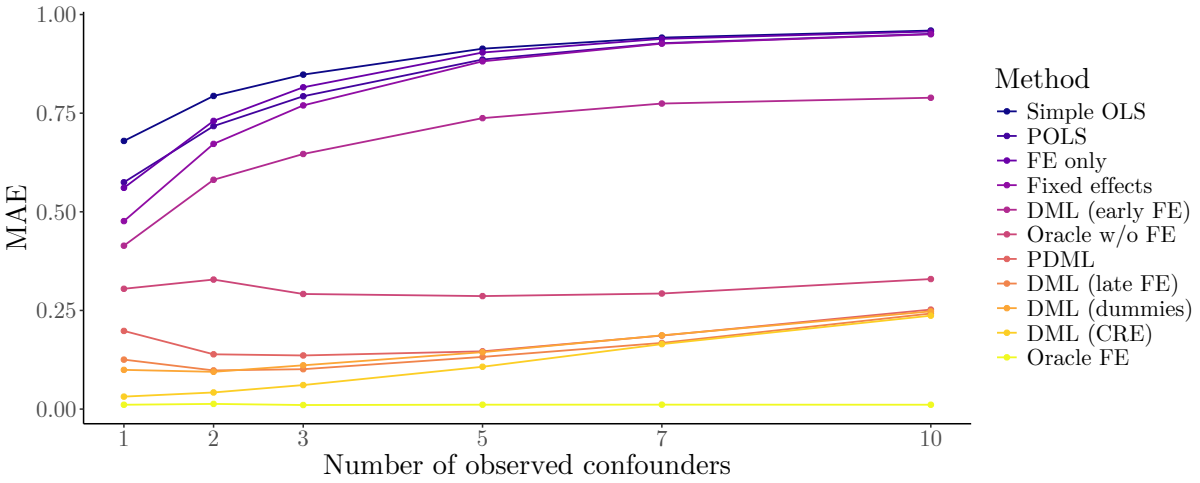


Figure 3.6: Mean absolute error in the estimated coefficient across 100 simulations by number of observed confounders. The simulated influence of the observed confounders is u-shaped, the causal structure is (C), i.e., $U_i \to X_{it}$. $N = 500$, $T = 10$.

("Oracle FE") perfectly adjusts for both observed and unobserved confounding and remains unbiased for all numbers of confounders, whereas only adjusting perfectly for observed confounding ("Oracle w/o FE") leads to an almost constant bias around 30%.

Contrary to the oracle methods, virtually all other approaches increase in MAE as the number of confounders increases. Also, the gap between methods explicitly accounting for unobserved heterogeneity and their counterpart that does not becomes smaller or vanishes for larger numbers of confounders. For example, "FE only" incurs a similar degree of bias as "Simple OLS" after 5 confounders. Comparing the DML methods (excluding early demeaning) to the functional form oracle, we observe that they all are substantially more accurate. This indicates that these methods do not only adjust for the observed confounding well (as "Oracle w/o FE" does), but also capture parts of the unobserved heterogeneity. Interestingly, the pooled DML approach outperforms the functional form oracle as well, even though it has no explicit way of adjusting for the unobserved heterogeneity. This is because PDML can *indirectly* adjust for part of the unobserved heterogeneity, as $\boldsymbol{X_{it}}$ contains variation caused by U_i ($U_i \rightarrow \boldsymbol{X_{it}}$), which facilitates the prediction of treatment and outcome. However, as the number of confounders and thus the dimensionality increases at a constant sample size, all DML methods after some point struggle to still adjust effectively. The pooled DML and "late" fixed effects approaches improve from one to two confounders, before increasing in bias similar to the others methods for more confounders. These approaches initially benefit from multiple $\boldsymbol{X_{it}}$: As the dimensionality of $\boldsymbol{X_{it}}$ increases, the direct influence of the unobserved U_i becomes less important, while the indirect influence is blocked by adjusting for the $\boldsymbol{X_{it}}$. This benefit of having multiple observed confounders in the $U_i \rightarrow \boldsymbol{X_{it}}$ case becomes even more evident when the prediction tasks are simpler, e.g., for linear confounding or larger sample sizes (see Figures B.7 and B.8, respectively, Appendix B.3.2). At the same time, adjustment for further $\boldsymbol{X_{it}}$ becomes more challenging in our baseline setting, hence the negative effects of additional confounders quickly dominate and the bias increases.

Using fixed effects late in DML and using dummies behave relatively similarly, with the dummy approach scaling slightly worse with the number of confounders. While DML with the CRE approach is almost unbiased for one confounder, its bias increases with the number of confounders more quickly than that of the other methods. From seven confounders on, it incurs bias similar to the late fixed effects method; for ten confounders, it is similar to the pooled DML. This is likely because the dimensionality in the CRE approach increases by a factor of $2 * J$, as for each observed confounder we have to adjust both for the time-varying variable and its time

mean. By contrast, the other methods only have to handle J variables, making the adjustment process less complex. While this appears to be a downside of the DML with CRE method, in the next set of simulations we investigate whether the method can handle larger numbers of confounders, provided a sufficiently large sample size.

Varying the sample size

We now assess how the sample size influences the estimates of the different methods. Is the bias in the estimates only a consequence of finite samples or is there some systematic issue with the estimators that prevents them from getting closer to the true effect? The results show that while most feasible methods are not guaranteed to substantially improve with sample size, DML with CRE can recover the true effect as long as the number of observations is large relative to the number and strength of the observed confounders.

First, we vary the number of units N in the baseline setting with causal structure (C) and one observed confounder with a u-shaped functional form (Figure 3.7). DML with CRE is the only feasible method that is close to the oracle method, and it gets more precise as the number of observations increases. None of the other DML-based methods substantially improve in larger samples, since they systematically lack the ability to model the unobserved heterogeneity.

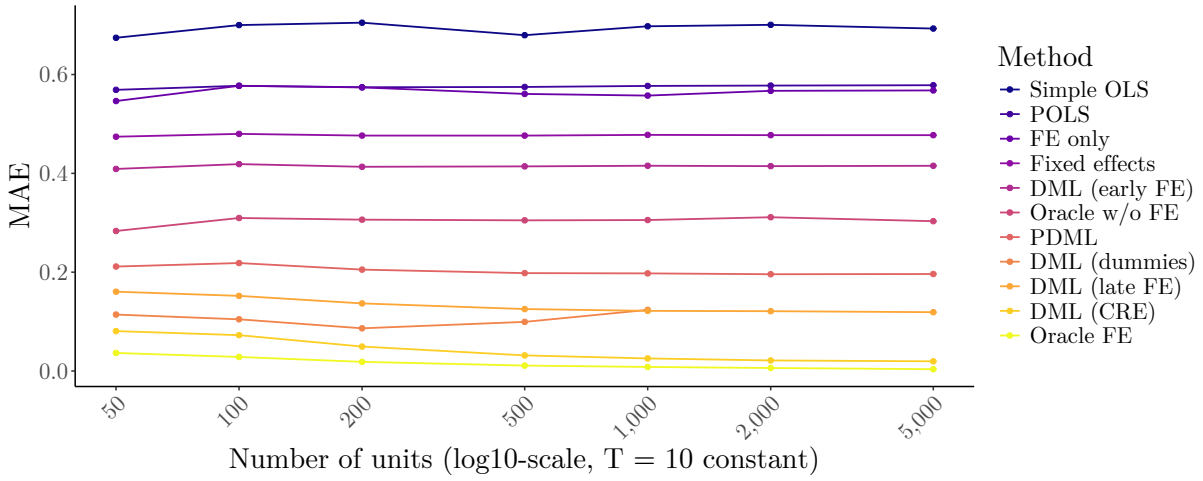


Figure 3.7: Mean absolute error in the estimated coefficient across 100 simulations for **1** observed confounder by the number of units. The number of periods is fixed at $T = 10$. The simulated influence of the observed confounder is u-shaped, the causal structure is (C), i.e., $U_i \to X_{it}$. We computed DML (dummies) only for up to $N = 1,000$, as it becomes computationally too costly for larger values.

This behavior changes when we look at a setting with more observed confounders (Figure 3.8; 5 confounders). Now the other DML-based methods (except for the early demeaning) substantially improve as the sample size increases, since the additional observed confounders contain

more information about the unobserved heterogeneity. However, DML with CRE remains dominant at every sample size, even extends its advantage for very large samples, and converges to the oracle method for $N = 5,000$.

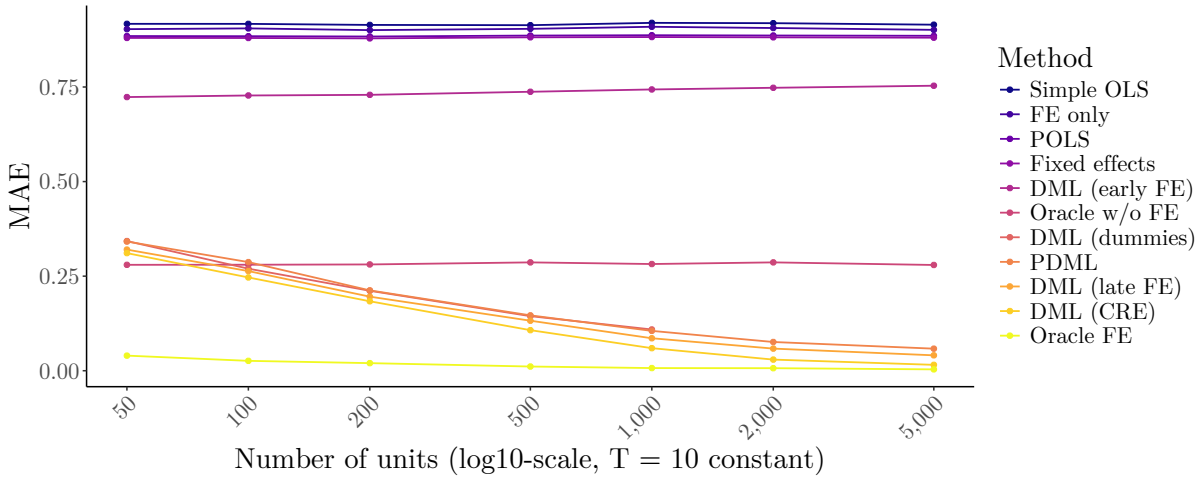


Figure 3.8: Mean absolute error in the estimated coefficient across 100 simulations for **5** observed confounders by the number of units. The number of periods is fixed at $T = 10$. The simulated influence of the observed confounders is u-shaped, the causal structure is (C), i.e., $U_i \to X_{it}$. We computed DML (dummies) only for up to $N = 1,000$, as it becomes computationally too costly for larger values.

We observe similar results for settings where we vary the number of periods T while keeping the number of units constant at $N = 500$ (see Figures B.9 and B.10, Appendix B.3.3). As the only noticeable difference, DML with early demeaning there also improves as the number of periods (i.e., the amount of within-variation) increases, although it is still heavily biased even at $T = 400$.

Two-way fixed effects

Here, we augment the unit-specific unobserved heterogeneity by also including time-specific unobserved heterogeneity. Hence, the DGP in Equations 3.13-3.15 now also contains the unobserved variable U_t, which varies only over time:

$$X_{it} = \alpha_0 + \delta U_i + \delta U_t + \epsilon_{it} \tag{3.13}$$

$$W_{it} = \alpha_1 + \gamma g_0(X_{it}) + \delta U_i + \delta U_t + \eta_{it} \tag{3.14}$$

$$Y_{it} = \alpha_2 + \beta W_{it} + \gamma m_0(X_{it}) + \delta U_i + \delta U_t + \mu_{it}. \tag{3.15}$$

In the estimation process, methods using fixed effects now employ *both* unit and time fixed effects, while the correlated random effects approach includes *both* unit and time means for treatment

and covariates, in addition to the original covariates. Compared to the baseline setting, no method performs substantially different under two-way fixed effects (Figure 3.9). All methods that cannot fully account for the unobserved heterogeneity in the baseline now perform slightly worse, since there is additional time-varying heterogeneity they can also not account for. DML with correlated random effects seem virtually unaffected by the added dimension.

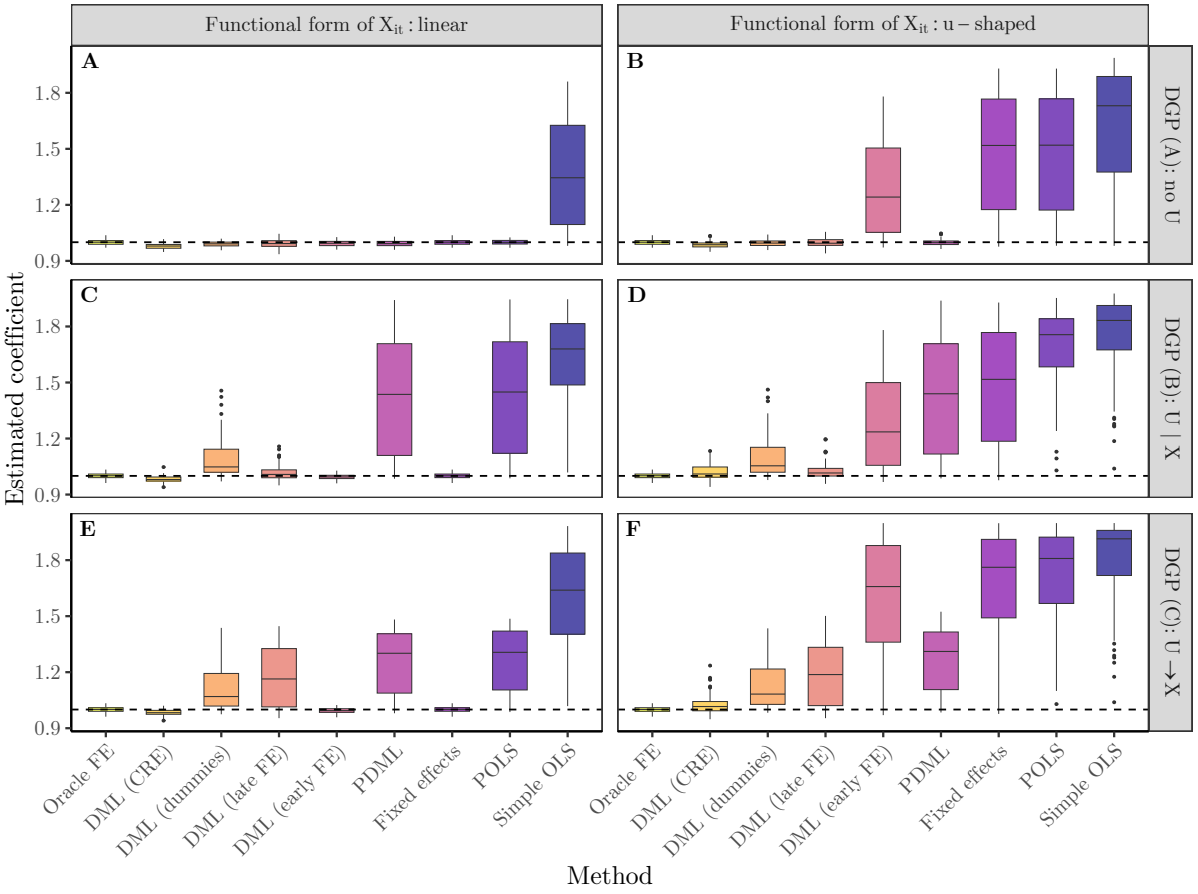


Figure 3.9: Results for DGPs with unobserved heterogeneity in both the unit and the time dimension with $N = 500$ units and $T = 10$ periods. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method. The three rows contain three different DGPs: "$no\,U$" indicates no unobserved heterogeneity, "$U \mid X$" means the unobserved heterogeneity influences treatment and outcome, but not confounders, and "$U \rightarrow X$" means the unobserved heterogeneity also influences the confounders.

Autocorrelation

One relevant difference between cross-sectional and panel data is the potential for autocorrelation (or serial correlation) in the latter, i.e., the error terms of the outcome model could be correlated across time (Wooldridge, 2012, Chapter 10). Serial correlation does not prevent consistency and unbiasedness in OLS under strict exogeneity (Wooldridge, 2012, Chapter 12), but invalidates

the usual OLS standard errors (even though cluster robust standard errors are available for traditional methods). While standard errors are not the focus of our analysis, we want to explore whether serial correlation can also become problematic for the estimated coefficients in DML, where it violates the i.i.d. assumption in the cross-fitting procedure. We already explored the impact of different cross-fitting procedures on the estimates in the presence of autocorrelation in Section 3.4.3. Now, we investigate how different degrees of autocorrelation affect the estimates in our baseline setting with causal structure (C) and nonlinear observed confounding.

To test the methods' sensitivity to autocorrelation, we change the DGP of the outcome model (Equation 3.9) to include serially correlated errors: Now, we simulate μ_{it} according to an AR(1) model, i.e., $\mu_{it} = \rho\mu_{it-1} + e_{it}$, with the ar-coefficient ρ equal to 0, 0.5, or 0.9, implying no, medium, and substantial autocorrelation, respectively. We allow for a longer time series than in our baseline by returning to the setting with $N = 100$ observations and $T = 50$ periods.

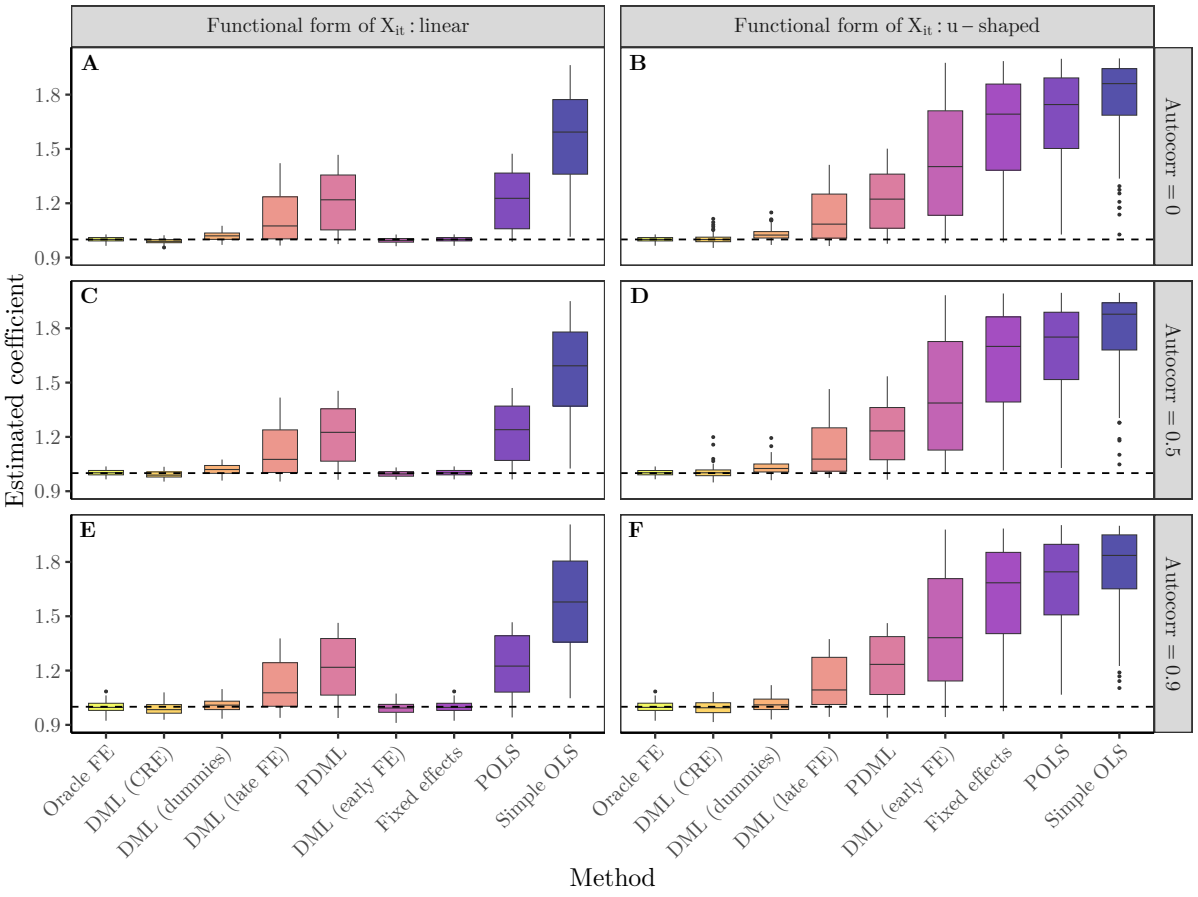


Figure 3.10: Results for different degrees of autocorrelation with $N = 100$ units and $T = 50$ periods. Panel A, B: no autocorrelation; Panel C, D: medium degree of autocorrelation; Panel E, F: strong autocorrelation. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method. The simulated influence of the observed confounders is u-shaped, the causal structure is (C), i.e., $U_i \to X_{it}$.

We find that even larger degrees of autocorrelation do not substantially alter our conclusions about the estimated coefficients (Figure 3.10). Only the distribution of the estimates of the well-performing method becomes slightly wider, while still being centered around very similar values. We conclude that in DML, the degree of autocorrelation is not critical for the accuracy of the estimated coefficients, and will probably only complicate the construction of valid standard errors.

3.5 Discussion

In this article, we explored how we can adapt the double/debiased machine learning framework to deal with unobserved heterogeneity in panel data settings. For this purpose, we considered several intuitive methods to account for unit-specific heterogeneity within DML. If DML can thereby adjust for time-constant unobserved confounding, similarly to the traditional fixed effects estimator, we can relax *two* assumptions when estimating causal effects: (1) the estimation assumption that we have chosen the correct functional forms in a parametric model, since a flexible ML method can in principle learn these within DML, and (2) the identification assumption that there is no unobserved confounding, since accounting for unobserved heterogeneity allows us to rely on the weaker assumption of no *time-varying* unobserved confounding. However, adaptations of DML to panel data settings are not straightforward due to two issues. First, the dependence of observations within the same cluster and/or across time violates assumptions underlying the cross-fitting procedure of DML. Secondly, nonlinear confounding relationships complicate the elimination of unobserved heterogeneity.

While our results show that the choice of cross-fitting procedure is not crucial for the accuracy of the estimated effects, we also find that many of the intuitive methods fail to recover the true effect in simulated data with nonlinear influences of the observed confounders. Although most DML-based methods are superior to estimating a misspecified (i.e., linear) fixed effects model in such settings, most of them cannot fully remove the confounding bias. We demonstrate that the influence of the unobserved unit-specific confounding on the observed time-varying confounders ($U_i \rightarrow X_{it}$) plays a critical role in the viability of some methods: Due to the nonlinear influence of the observed confounders, the unobserved heterogeneity is no longer additively separable and most approaches cannot easily eliminate it. The only DML estimator delivering good estimates across settings is the one using the Mundlak-type correlated random effects approach within

DML: By using the time-means of the treatment and the covariates as additional predictors within DML, this approach can explicitly model the unobserved heterogeneity, even in cases with nonlinear observed confounding. One caveat of this estimator is the need for a sample size that is large relative to the number of observed confounders, which however is not unreasonable in many applications.

We next discuss whether it is warranted to stress the importance of the influence of the unobserved heterogeneity on the observed confounding ($U_i \rightarrow X_{it}$) like we do in this article. Is this path likely to be present in typical applications? We consider two classical causal questions as examples: the evaluation of job programs, and the estimation of the price elasticity of demand for consumer goods. First, one standard problem in econometric textbooks is estimating the effect of a job program (W_{it}) on future earnings of the participants (Y_{it}) (e.g., Wooldridge, 2010, Chapter 10). Important unobserved, but (relatively) time-constant confounders (U_i) could be the ability or motivation of individuals, which could influence both whether they participate in the training and their future earnings. Observed confounders typically consist of individual characteristics like age, sex, years of schooling, marital status, number of hours worked before the training, etc. While some of these are certainly not influenced by the unobserved heterogeneity (age, sex) and would drop out of a fixed effects regression anyway, others can vary over time (schooling, marital status, hours worked) and are plausibly influenced by the ability and/or motivation ($U_i \rightarrow X_{it}$). Secondly, we return to our example from the introduction. Because pricing decisions are very consequential for many products, retailing firms want to know the effect of price (W_{it}) on demand (Y_{it}) (e.g., Bijmolt et al., 2005). However, if the available panel data consists of different stores over time, there could be unobserved heterogeneity (U_i), e.g., due to management quality, the attractiveness of the store location, or demographic characteristics of residents nearby (e.g., Papies et al., 2017; Wooldridge, 2010). At the same time, observed factors like advertising, sales promotions, or prices of competitor products could function as time-varying confounding variables (X_{it}), and are likely also influenced by the unobserved time-constant factors ($U_i \rightarrow X_{it}$). In sum, these two examples illustrate that the influence of U_i on X_{it} is arguably common in practice. As a consequence, researchers should ensure that the estimators they use are capable of dealing with this setting, like we show for DML with CRE.

From our results, we derive the following recommendations: (1) In applications where researchers currently use the traditional fixed effects regression, we recommend additionally using DML with correlated random effects and comparing the estimated coefficients. This robustness

check can indicate potential for functional form misspecification if the estimated effects are very different. (2) When comparing the results, we recommend making the confidence in the DML estimate dependent on the sample size. We encourage a higher confidence in the accuracy of the estimated effect if the sample size is large and the number of observed confounders is small. (3) We recommend *not* splitting by unit within cross-fitting, at least if the goal is to model the unobserved heterogeneity within the ML prediction steps. (4) We discourage interpreting standard errors or confidence intervals stemming from our considered DML estimators. There is still a lack of clarity about how to obtain valid variance estimators in these settings with both cluster and time dependence, especially in combination with the cross-fitting procedure.

Our final recommendation already ties in with the limitations of our study. We focus only on whether the suggested estimators are capable of retrieving the true causal effect in a variety of simulated settings. We do not provide any asymptotic properties of these estimators and are thus not able to construct valid confidence intervals. Also, our considered estimators are motivated by intuitively appealing adaptations of existing panel data methods and not built on Neyman-orthogonal scores like the ones in Chernozhukov et al. (2018). By contrast, Clarke and Polselli (2023) derive a Neyman-orthogonal score for a similar setting, but also show inference to be problematic in a number of situations. Further exploration of the possibilities and limitations of statistical inference in these settings will be crucial for a more widespread adoption of these methods in practical applications. Finally, we are assuming the absence of dynamic effects and effect heterogeneity, and only consider additive fixed effects. Future research could study the consequences of violating these assumptions and propose alternative DML adaptations for these situations.

Chapter 4

It's All in the Data? Potential and Limitations of Causal Discovery in Social Science

Jonathan Fuhr

Statement of contribution

The author confirms sole responsibility for the development of the study, choice of methods, data collection, simulation, analyses, presented results, and manuscript writing.

Acknowledgements

The author gratefully acknowledges David Scheuermann for his valuable feedback on an earlier version of the working paper. Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC number 2064/1 – Project number 390727645. Further, the author acknowledges support by the state of Baden-Württemberg through bwHPC.

Chapter 4 is based on the working paper "Fuhr, J. (2024). It's All in the Data? Potential and Limitations of Causal Discovery in Social Science" (not yet submitted).

Abstract

Traditional causal inference methods in economics and social science assume knowledge about the underlying causal structure of a problem, which researchers subsequently utilize to identify and estimate causal effects. However, recent research in computer science and philosophy has explored under which conditions the causal structure itself is discoverable from observational data. In this paper, I explore the potential of *causal discovery* for typical questions in the social sciences, where the final goal is the estimation of causal effects. For this purpose, I review the fundamental methodology of the field, before focusing on two methods that provide a full pipeline from causal discovery to causal effect estimation. I evaluate the methods on simulated data and apply them to the classical dataset about 401(k) eligibility and net financial assets. My results indicate that discovery of causal structure and subsequent effect estimation is in principle possible, but only with very large sample sizes and under rather restrictive assumptions, which will rarely be justifiable in social science applications. Finally, I discuss the general applicability of causal discovery in social science, concluding that the methods cannot replace theory and domain knowledge, but might be useful for exploratory purposes and initial hypotheses generation in fields with very little prior causal knowledge.

4.1 Introduction

Causal questions of various kinds are at the heart of scientific inquiry: "What is the effect of the minimum wage?", "Do vaccines reduce mortality?", "What is the cause of Alzheimer's disease?", "How will targeted advertising affect demand?", etc. Much of the methodological research around such questions has been concerned with the identification and estimation of causal effects, given data and a causal model (e.g., Hernán and Robins, 2020; Imbens and Rubin, 2015). This approach assumes prior causal information: Researchers derive a (hopefully) plausible causal structure from theory and domain knowledge, either encoded in functional equations or a causal graph. If the assumptions of this causal structure are correct, it contains information about how to identify the causal effect from observed data by indicating sources of exogenous information, variables to adjust for, etc. Given valid identification, researchers can estimate the causal effects of interest from the data using a variety of sophisticated techniques (Angrist and Pischke, 2009).

While the fields of statistics and econometrics have made much progress regarding issues of identification and estimation, another research field has emerged in computer science (e.g., Pearl, 2009; Peters et al., 2017) and philosophy (e.g., Spirtes et al., 2001) that asks an even more ambitious question: Can we actually start earlier in the causal inference workflow and learn something about the causal structure itself directly from the data? If possible, this could prove especially useful if there is a lack of theory about a problem with many components and/or if the theory is ambiguous and conflicting. Additionally, causal reasoning is a key aspect missing in current artificial intelligence applications, which is why the machine learning community dedicates a great amount of resources to learning about and from causal structure (e.g., Schölkopf, 2022).

Methods in this field of *causal discovery* or *causal structure learning* try to learn as much as possible about causal relationships between variables from observational (or experimental) data (e.g., Eberhardt, 2017). The output is typically a causal graph or a set of possible causal graphs that could have generated the observed data. The intuition behind these methods is that the data-generating process under certain conditions leaves a footprint on the observed data, which algorithms in principle can (partly) discover (Glymour et al., 2019).

Since data alone only allows conclusions about correlations and not directions of effects, all causal discovery algorithms rely on additional assumptions (Eberhardt, 2017). Different algorithms impose different assumptions, which might be more or less reasonable, depending on

the application. Whether these assumptions are plausible for realistic applications is subject to an intense debate. Gelman (2011) is "extremely skeptical" of the goal of learning causal structure, especially in social science. His main argument is that in real-world applications, there is a non-zero effect between virtually every pair of variables, which makes methods that rely on conditional independencies infeasible. Imbens (2020) emphasizes that causal discovery asks very different questions than traditional causal inference in economics. Although he states the potential relevance of these questions, he concludes that the methods are currently not feasible for real-world application. On the other hand, Shalizi (2021) argues that assumptions in causal discovery are "comparable to or weaker than assumptions for causal estimation problems", pointing out the strong identification and estimation assumptions necessary for, e.g., propensity score matching. Similarly, Malinsky and Spirtes (2017) propose a causal discovery method as an alternative to propensity score techniques that assume unconfoundedness, an assumption their method does not rely on.

These opposing views warrant a deeper investigation into the relevance and applicability of these methods for real-world causal questions in the social sciences. While there has been plenty of foundational research and theoretical development in causal discovery, it is still not obvious to many applied researchers whether and how these developments can contribute to the typical causal questions in their field. In contrast to standard identification assumptions, it may be more difficult to assess the plausibility of assumptions in causal discovery in the respective domain. So far, a majority of applications of causal discovery algorithms have been in biology, genetics, or related fields (see Glymour et al., 2019). In social science, credible applications are currently very rare. Therefore, the goal of this paper is to explore the usefulness of current causal discovery methods when the final goal is not only the discovery of causal structure, but also the estimation of specific causal effects, which are typically of key interest in social science. While this article does not aim to be a comprehensive treatment of all developments in the field, it reviews, assesses, and applies popular discovery methods that could potentially aid in the process of going from data to causal effects.

To this end, I first provide a basic overview of the main terminology, assumptions, and algorithms in causal discovery (Section 4.2). Then, I focus in more detail on two methods that aim to provide a full pipeline from causal structure discovery to effect estimation. The first is called "intervention-calculus when the DAG is absent" (IDA; Maathuis et al., 2009), for settings where all variables are observable (Section 4.3). The second one is "latent variable IDA" (LV-

IDA; Malinsky and Spirtes, 2017), which allows for unobservable variables (Section 4.4). I assess the performance of the two methods on simulated data, where both the true graph and the true effects are known. In Section 4.5, I demonstrate and evaluate the application of both methods to the classical dataset about 401(k) eligibility and net financial assets (Poterba et al., 1994). Finally, based on my findings, I critically discuss the general applicability of causal discovery methods for typical causal questions in social science (Section 4.6). I focus on the plausibility of assumptions and give my perspective on the future potential of algorithms from this field.

4.2 A brief overview of causal discovery

In this section, I provide a non-comprehensive overview of the concepts, assumptions, and algorithms from causal discovery that are relevant for this article. For more extensive reviews and introductions see Eberhardt (2017), Glymour et al. (2019) (includes applications and practical guidance), Heinze-Deml et al. (2018a) (includes a simulation study comparing different methods), Squires and Uhler (2023) (from a combinatorial perspective), and Vowels et al. (2022) (focuses on continuous optimization methods).

4.2.1 General concepts and terminology

Most causal structure learning methods are based on causal graphs, with the most informative version of a causal graph being a Directed Acyclic Graph (DAG). A DAG consists of nodes, representing variables, and directed edges, representing the flow of causation or the direction of effects. The absence of an arrow indicates that there is no direct causal relationship between two variables. Most causal discovery methods assume *acyclicity*, i.e., the graph cannot contain any cycles, where a path of edges pointing in the same direction ends up at the starting node (e.g., Eberhardt, 2017). In a causal graph, the direct causes of a variable are called its *parents*, while its direct effects are called its *children*. Furthermore, *ancestors* are all variables occurring before a variable on a directed path, and all variables occurring after it are its *descendants*. In Figure 4.1, nodes A and B are parents of X, while W and Y are children of X. A node with two incoming arrows is called a *collider*, e.g., X is a collider on the path from A to B. If additionally (as in Figure 4.1), A and B do not have any direct causal relationship, this part of the graph is a *v-structure* or X is an *unshielded collider*. This special structure is crucial for orienting edges

in many causal discovery algorithms (e.g., Kalisch and Bühlmann, 2014). On the other hand, in relation to W and Y, X is a *common cause* or a *confounder*.

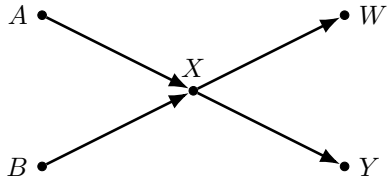


Figure 4.1: A simple DAG to illustrate graph concepts

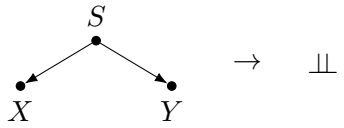
The terms *d-connectivity* and *d-separation* are useful when thinking about the flow of causation in a causal graph. Two nodes/variables in a graph are said to be *d-connected* if there exists a connecting path between them (e.g., $A \to Y$), and *d-separated* if not (Pearl et al., 2016). We say a path is blocked either if there is a collider, which *is not* conditioned on, or if there is a mediator or confounder that *is* conditioned on. E.g., in Figure 4.1, the path $A \to B$ is blocked, if we do not condition on the collider X. However, if we want to block the path $A \to Y$, we need to condition on X.

These rules lie at the core of Pearl's *do-calculus*, which is a formal way to get from any causal graph to an identification statement. That is, the do-calculus states whether a causal effect of interest is identified from the observed variables in the graph, and if so, which variables we need to adjust for, use as instruments, etc. (Pearl, 2009). One of the most important tools of the do-calculus is the *backdoor-criterion*. Given a DAG, it determines which variables one should condition on when estimating the effect of one variable on another (Pearl et al., 2016). In causal discovery, the concept of d-separation helps to connect relationships between variables in a causal graph to conditional independencies in data originating from this graph. Under assumptions stated in the next subsection, this connection enables learning causal graphs from data.

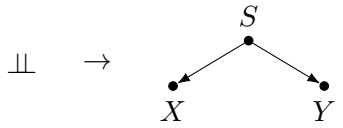
4.2.2 Assumptions

Like all causal inference methods, causal discovery algorithms rely on assumptions, which are often strong and untestable (e.g., Eberhardt, 2017). Depending on the family of methods, different assumptions are necessary, where some might be more plausible in certain applications than others.

Two of the most common assumptions, underlying virtually all constraint-based and score-based structure learning methods, are the *causal Markov* and the *causal faithfulness* assumptions (Figure 4.2). Together, they imply that we can get from d-separation statements in the causal graph to (conditional) independence statements in the data and vice versa (Eberhardt, 2017). The causal Markov assumption states that if two nodes in a causal graph are (conditionally) d-separated, then the corresponding variables in the data are (conditionally) independent (Heinze-Deml et al., 2018a). For example, in Figure 4.1, A is d-separated from Y by X. If the causal Markov condition holds, then, in data generated by this graph, A will also be statistically independent from Y conditional on X. More formally, the causal Markov condition states that every variable in the causal graph is probabilistically independent of its non-descendants given its parents (e.g., Eberhardt, 2017). This assumption is rather weak and difficult to violate in standard settings (Eberhardt, 2017). If two or more graphs share the same d-separation statements – therefore implying the same conditional independencies – they are called *Markov equivalent* (Glymour et al., 2019). The set of DAGs that are Markov equivalent is a *Markov equivalence class (MEC)*. Many causal discovery algorithms discover a MEC of graphs that are compatible with the observed data, rather than a singular unique DAG.



(A) Causal Markov assumption: d-separation in graph implies (conditional) independence in data.



(B) Faithfulness assumption: (conditional) independence in data implies d-separation in the graph.

Figure 4.2: Causal Markov and faithfulness assumptions

The counterpart of the causal Markov assumption is the causal faithfulness assumption. While the Markov assumption links the graph to the data, the faithfulness assumption links the data to the graph and is therefore essential for causal discovery (Figure 4.2). We call a data distribution faithful to a graph, if each (conditional) independence in the data corresponds to a d-separation in the graph (e.g., Eberhardt, 2017). For example, if in Figure 4.1, a conditional independence test on the data tells us that A and Y are independent conditional on X, we assume that A and Y are d-separated by X in the causal graph. It is much easier to encounter a violation of the faithfulness assumption: Two causally related variables will be statistically independent if there are two causal pathways with coefficients that exactly cancel each other out (Eberhardt,

2017). We will discuss potential violations of the faithfulness assumption throughout this paper and in the final discussion.

The next two common assumptions are more familiar, since they are also common in discussions of causal identification. First, the assumption of *acyclicity* does not allow for cycles or feedback loops in the discovered graph (Heinze-Deml et al., 2018a). This is a standard assumption in DAGs (e.g., Pearl et al., 2016), and gets more plausible with increasingly granular (i.e., less aggregated) data. The second assumption is called *causal sufficiency* and implies that there are no unobserved, unmeasured, or latent confounders of any two variables in the causal graph (e.g., Eberhardt, 2017). This assumption is similar to the standard unconfoundedness assumption for causal identification (Imbens and Wooldridge, 2009), but more restrictive, since it applies to any pair of variables, not just the treatment and outcome variable.

Finally, a particular class of causal discovery methods based on a functional causal model makes specific assumptions about functional forms or noise distributions (e.g., Glymour et al., 2019). I will mention these assumptions in the next subsection for the respective methods.

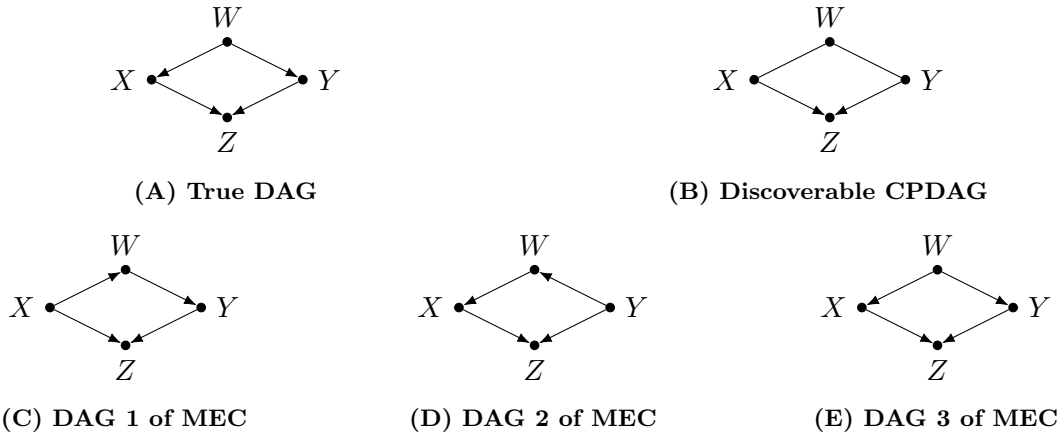
For most of the assumptions discussed, there are methods that try to relax them in some way, albeit often at the cost of less informative results. When introducing different causal discovery algorithms in the following, I point out which assumptions are necessary for each.

4.2.3 Methods

Here, I provide an overview of the most popular causal discovery algorithms in the literature. Since I will use them in the following sections, I explain the PC and FCI algorithms in more detail, while describing alternative methods only briefly. One can classify most causal structure learning algorithms as being either constraint-based, score-based, or based on a functional causal model.

First, constraint-based methods heavily rely on conditional independence tests and therefore must assume Markov and faithfulness. One of the first popular causal discovery algorithms was the *PC algorithm*, short for Peter Spirtes and Clark Glymour, its inventors (Spirtes et al., 2001). It starts with a complete undirected graph and uses conditional independence tests to delete edges. Then, it determines the occurrence of v-structures, which helps to direct the involved edges. Finally, it follows several additional rules that enable orientations of remaining undirected edges. The final output is a Markov equivalence class of possible DAGs. This MEC is often summarized in a CPDAG (Completed Partially Directed Acyclic Graph), which is a

graph where undirected edges imply that different graphs in the MEC do not agree on the edge orientation (e.g., there is at least one graph where $X \to Y$ and one graph where $X \leftarrow Y$). Figure 4.3 illustrates how CPDAGs summarize Markov equivalence classes. In addition to Markov and faithfulness, the PC algorithm also assumes acyclicity and causal sufficiency. While the algorithm itself does not make assumptions about specific functional forms and distributions, the conditional independence tests are simplest for linear relationships and Gaussian distributions (Heinze-Deml et al., 2018a). It is possible to use any conditional independence tests within the PC algorithm, but the most flexible nonparametric tests often require very large sample sizes and are computationally very expensive (e.g., Heinze-Deml et al., 2018b; Shah and Peters, 2020).



(A) True DAG **(B) Discoverable CPDAG**

(C) DAG 1 of MEC **(D) DAG 2 of MEC** **(E) DAG 3 of MEC**

Figure 4.3: How CPDAGs can summarize Markov equivalence classes of DAGs: **(A)** The true, unknown graph. **(B)** The CPDAG that summarizes all DAGs in the MEC. Directed edges imply that all DAGs agree on the orientation, undirected edges represent disagreement. The v-structure $X \to Z \leftarrow Y$ enables the correct orientation of the respective edges in all graphs. **(C)**-**(E)** The three different DAGs in the MEC described by the CPDAG, containing the true DAG **(E)**.

Similarly to Glymour et al. (2019), I illustrate the PC algorithm with a simple example in Figure 4.4. The first panel shows the true causal graph. Figure 4.4B is the starting point of PC, the complete undirected graph. Subsequently, the algorithm deletes edges based on (conditional) independencies. First (Figure 4.4C), it removes the edge $Z - Y$ because $Z \perp\!\!\!\perp Y$ (since the collider W blocks the path if not conditioned on). Conditioning on one variable does not lead to any additional independencies. However, conditioning on W and Y simultaneously shows the independence of Z and X (Figure 4.4D). Then, the v-structure $Z - W - Y$ determines the orientation of the two edges, since Z and Y become dependent when conditioning on W (Figure 4.4E). Finally, we can orient further edges (Figure 4.4F): The edge $W - X$ is directed $W \to X$, because otherwise we would not have found the independence $Z \perp\!\!\!\perp X|W, Y$ in Fig-

ure 4.4D. Then, we orient $Y - X$ as $Y \to X$, since the other direction would lead to a cycle and Z and Y would no longer be unconditionally independent.

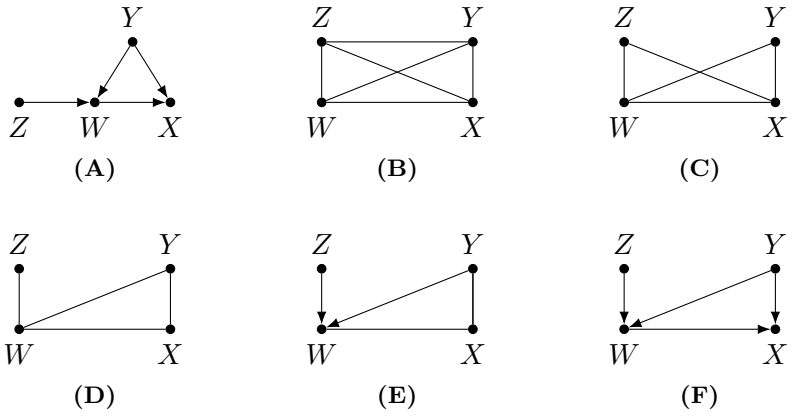


Figure 4.4: Illustration of the PC algorithm. **(A)** True graph. **(B)** Complete undirected graph. **(C)** $Z - Y$ removed because $Z \perp\!\!\!\perp Y$. **(D)** $Z - X$ removed because $Z \perp\!\!\!\perp X|W, Y$. **(E)** v-structure between $Z - W - Y$ determines orientations. **(F)** Further orientations based on conditional independence and acyclicity. Illustration based on Glymour et al. (2019).

Importantly, the PC algorithm depends on the faithfulness assumption being valid in this example. Since there are two directed paths from Y to X ($Y \to X$ and $Y \to W \to X$), faithfulness requires that these paths do not cancel each other out. But there are also more subtle ways how very specific parameter settings can lead to conditional independence tests failing in this example (e.g., all coefficients equal to 1 and simultaneously a variance of 1 in the noise terms of W and Y, see Appendix C.1 for details). Given that faithfulness holds, we uniquely identify the true graph in this stylized example. In most more complex examples, some edges will remain undirected, as no rule can determine the direction. Then, the output is a MEC of multiple DAGs, often summarized as a CPDAG. Given the causal Markov, faithfulness, acyclicity, and sufficiency assumptions and i.i.d. data, the PC algorithm is consistent, i.e., it converges to the true MEC for infinite data (Glymour et al., 2019; Spirtes et al., 2001).

The FCI ("Fast Causal Inference") algorithm is an alternative constraint-based discovery method that drops the causal sufficiency assumption and thus allows for hidden variables (Spirtes et al., 2001), while still being asymptotically consistent in sparse settings (i.e., if the true number of edges is limited) (Colombo et al., 2012). The algorithm uses additional tests and orientation rules and can thereby sometimes detect the presence of unobserved confounding when still assuming causal Markov, faithfulness, and acyclicity. To represent a class of DAGs that can include unobserved variables, FCI needs to makes use of a different graph type called Maximal Ancestral Graphs (MAGs) (e.g., Heinze-Deml et al., 2018a). MAGs display the existence of

one or multiple unobserved common cause(s) between two variables by a bidirected edge (e.g., $X \leftrightarrow Y$). The edge marks in MAGs encode ancestral relationships between two variables (e.g., X and Y): Without knowing the edge mark on the other side, an arrowhead at X implies that X *is not* an ancestor of Y, while a tail mark at X indicates that it *is*. Importantly, a MAG can represent multiple DAGs that have the same ancestral relationships and d-separation properties between the observed variables (Zhang, 2008). This implies that in some cases, an edge $X \rightarrow Y$ ("X is an ancestor of Y") in a MAG can represent a DAG with both $X \rightarrow Y$ and $X \leftrightarrow Y$ (i.e., unobserved confounding in addition to a direct effect). However, there are other cases where we can rule out the additional unobserved confounding by a condition called "edge visibility" (Maathuis and Colombo, 2015; Zhang, 2008), which is crucial for the LV-IDA method (see Section 4.4). Similar to DAGs and CPDAGs, there are typically multiple MAGs in the same Markov equivalence class. Hence, the output of FCI is a Partial Ancestral Graph (PAG), which is a MEC of MAGs. PAGs introduce a circle as an additional edge mark (e.g., $X \circ\!\!\rightarrow Y$), which represents uncertainty about this edge mark in the different MAGs. I illustrate the concepts of MAGs and PAGs and the difference between FCI and the PC algorithm with a simple example in Figure 4.5, where FCI can detect an unobserved confounder. The additional steps of the FCI algorithm make it substantially slower than the PC algorithm (Colombo et al., 2012), which is why multiple faster, but slightly less informative versions are available (e.g., RFCI by Colombo et al. (2012)).

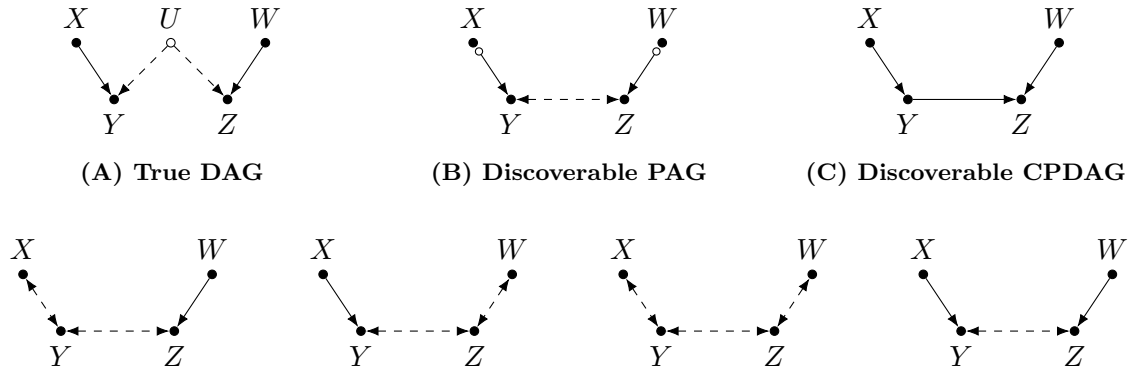


Figure 4.5: Illustration of the outputs of the FCI and PC algorithms in a setting with an unobserved common cause. The example is based on Glymour et al. (2019). **(A)** The true causal graph, where U is an unobserved confounder. **(B)** The PAG learned by FCI. The circles indicate uncertainty about the edge orientation. The bidirected edge between Y and Z correctly posits the existence of a hidden variable. FCI learns about this confounder because conditional independence tests indicate that both Y and Z must function as colliders. **(C)** The CPDAG learned by PC, which falsely discovers an edge $Y \rightarrow Z$. PC assumes sufficiency, which is violated by the existence of U. **(D)-(G)** All MAGs in the MEC described by the PAG in **(B)**, containing the true MAG **(G)**.

Different from constraint-based algorithms, score-based discovery methods search for a graph that maximizes a scoring function (e.g., the Bayesian Information Criterion, BIC) in relation to the data. The most prominent of these algorithms is the Greedy Equivalence Search (GES), which also operates under the assumptions of Markov, acyclicity, sufficiency, and an assumption similar to faithfulness (Chickering, 2002). GES uses a greedy search over different MECs, consisting of a forward phase, where it adds edges based on improvement in score, and a backward phase, where it deletes edges to further improve the score. Even though it uses a very different search method, GES often yields similar results to PC, converges to the same MEC in the large sample limit and is also consistent under some assumptions (Chickering, 2002; Glymour et al., 2019).

In addition to constraint-based and score-based algorithms, there are hybrid algorithms that combine elements of the two. For example, Max-Min Hill Climbing (MMHC) uses constraint-based methods to learn the undirected graph, before determining edge orientations based on a score (Tsamardinos et al., 2006), while GFCI finds a larger graph using GES and then prunes and orients edges with FCI (Ogarrio et al., 2016).

As the third larger category of methods, I discuss algorithms based on functional causal models. These methods start with structural equation models and impose additional assumptions on functional forms or the distribution of the noise term. With these assumptions, they can identify the causal direction even in the two-variable case and estimate unique DAGs instead of a MEC, all without relying on the faithfulness assumption (e.g., Peters et al., 2017). For example, the LiNGAM method assumes "Linear Non-Gaussian Acyclic Models" (Shimizu et al., 2006) and thus can infer the causal direction between two variables if the functional form is linear, the additive noise term is not normally distributed, and there is no unobserved confounder. LiNGAM makes use of the fact that if the true model is $Y = X + \epsilon$, with ϵ being non-Gaussian, the residual of a regression of X on Y (the anti-causal direction) will not be independent of the regressor, while it will be in a model with the causal direction (for a graphical illustration, see, e.g., Glymour et al., 2019). Extensions of the LiNGAM idea include nonlinear additive noise models (Hoyer et al., 2008) as well as post-nonlinear models (Zhang and Hyvärinen, 2009). In nonlinear additive noise models, the noise distribution is not restricted, but the causal relations are assumed to be nonlinear with additive noise: $Y = f(X) + \epsilon$, with f being nonlinear. In this case, the causal direction is identifiable under Markov, acyclicity and sufficiency in all but very rare cases (Eberhardt, 2017). In the post-nonlinear model (PNL), there is an additional nonlin-

ear transformation: $Y = f_2(f_1(X)+\epsilon)$, where f_1 and f_2 are nonlinear functions. The function f_2 denotes sensor or measurement distortion, which occurs frequently in practice (Glymour et al., 2019), and its inclusion has led to good performance of the PNL in causality challenges (Zhang and Hyvärinen, 2009). The PNL has a very general form, of which LiNGAM and the nonlinear additive noise models are special cases (Glymour et al., 2019). While these FCM-based methods do not rely on the faithfulness assumption and can determine unique models, their functional form assumptions are difficult to assess, and they typically require very large sample sizes (Malinsky and Danks, 2018).

In addition to these three larger categories of methods, there are several other algorithms and developments in the field of causal structure learning. First, since learning causal structure from purely observational data is very hard, some methods leverage interventional/experimental data to get closer to the true causal graph. For example, the Greedy Interventional Equivalence Search (GIES) is an adaption of GES that uses known interventions (Hauser and Bühlmann, 2012); and backShift (Rothenhäusler et al., 2015) uses unknown shift interventions to learn invariances. Second, another category of score-based methods views the search for the causal graph as a continuous optimization problem and tries to solve it using deep learning techniques (for an overview, see Vowels et al., 2022). Also, several algorithms allow the inclusion of background knowledge as constraints for the algorithm (see, e.g., Eberhardt, 2017).

In the next two sections, I introduce two methods that build on the PC and FCI algorithms, respectively, and use their outputs to estimate bounds on causal effects from the learned graphs.

4.3 Discovery and effect estimation under sufficiency

4.3.1 The IDA algorithm

Traditionally, much of the causal inference literature in economics and social science has focused on the identification and estimation of causal effects (e.g., Imbens and Rubin, 2015). In other words: Given assumptions about the causal structure of a specific application, is it possible to estimate the effect from observable data (identification) and if yes, which statistical methods should we use to compute the effect (estimation)? In these disciplines, researchers typically derive the causal structure and related assumptions from domain knowledge and established theory. On the other hand, much of the literature on causal discovery focuses on finding the

causal structure represented by a graph, but is less interested in the strength of the discovered causal relationships.

Therefore, a natural development is the combination of causal discovery, identification, and estimation in one consistent pipeline. For this purpose, Maathuis et al. (2009) developed the IDA method, where IDA stands for "intervention-calculus when the DAG is absent" (see also Maathuis et al., 2010). The IDA method consists of three steps. First, it uses the PC algorithm to learn the Markov equivalence class of possible causal graphs, typically summarized in a CPDAG. Then, it enumerates all possible DAGs and applies do-calculus to each graph to identify and estimate the total causal effect of one variable on another. In the original IDA method, estimation occurs by computing a linear regression, where all parents of the treatment in the respective graph enter as covariates ("backdoor-adjustment"). This results in one effect estimate for each possible DAG in the CPDAG. Finally, one can summarize this multiset of effects, e.g., by computing lower or upper bounds on the size of the true effect. For settings with many variables, the number of possible DAGs and thus regressions can be very large, making the estimation computationally expensive. For these cases, Maathuis et al. (2009) introduce a "local" algorithm, which only considers uncertainty about edges in the neighborhood of the treatment variable for determining the parents, and is thus much faster than the "global" version.

In terms of assumptions, the IDA method inherits the reliance on sufficiency, causal Markov, faithfulness, and acyclicity from the PC algorithm. Additionally, it assumes that all variables follow a multivariate normal distribution, which implies linear relationships between the variables and enables the use of linear regression for estimating the effects (Maathuis et al., 2009). The final assumption is equal variance of all regressors, which is only necessary if the goal is a direct comparison of the effects of different variables. Given these assumptions, Maathuis et al. (2009) prove that IDA is asymptotically consistent even in high-dimensional settings (more covariates than observations), if the true number of edges between the variables is limited ("sparsity").

Contrary to other methods in the field of causal discovery, Maathuis et al. (2009) motivate their development of IDA with a specific application, and other applications, mainly in biology, have followed with remarkable success. The authors' initial goal was to infer the effects of different genes on the vitamin production in a specific bacterium. In their setting, only observational, but very high-dimensional data (71 observations, 4088 covariates/genes) is available. Here, experimental interventions on all genes are infeasible, but alternative purely associational measures of variable importance lack a causal foundation. Therefore, Maathuis et al. (2009) aim

to estimate the lower bound on the effect of each gene. Subsequently, researchers can experimentally test the genes with the largest estimated effect. The authors compare their approach with the results obtained by using lasso as an associational ML approach, and find a very different set of genes with the largest effects. However, in the absence of a ground truth, they are unable to evaluate which of the approaches performs better. To deal with this limitation, in a follow-up study, Maathuis et al. (2010) present an experimental validation of IDA in an application where they estimate the effect of different genes in a species of yeast. They have access to a large number of "true" effects from interventional data, and assess whether IDA can identify the strongest of these effects from observational data. Their results show that IDA substantially outperforms random guessing, but also associational ML approaches such as lasso and elastic net. When further developing IDA and estimating the effects of genes on the time to flowering of a specific plant, Stekhoven et al. (2012) also find very positive results of evaluating IDA with interventional data: New experiments, based on the IDA results, confirmed previously unknown effects of four genes, and validation on previously conducted experiments again demonstrated superior performance compared to lasso and elastic net. Beyond these early applications and validations, researchers have applied IDA in various further settings, mostly in disciplines related to biology or medicine. Although some of these applications also work with genetic data and are therefore very high-dimensional, others operate in settings where the number of observations is much larger than the number of variables (e.g., Ehrmann et al., 2019; Kalisch et al., 2010). These "low-dimensional" settings are also more common in social science applications, where we typically have access to much more observations than variables. Generally, causal discovery methods and conditional independence tests only benefit from larger sample sizes (e.g., Glymour et al., 2019), although it might be more difficult to argue for the sufficiency assumption when only few variables are considered.

Since the assumptions underlying IDA are strong, several papers have proposed extensions of IDA that relax some of these assumptions. Some try relaxing the assumption of normally distributed variables (e.g., Nandy et al., 2017; Teramoto et al., 2014), others the assumption of causal sufficiency (e.g., Frot et al., 2019; Malinsky and Spirtes, 2017). Additionally, further extensions allow adding background knowledge to IDA, which can help to decrease the number of DAGs in the equivalence class (e.g., Fang and He, 2020; Perkovic et al., 2017). Lastly, Stekhoven et al. (2012) and Taruttis et al. (2015) propose variations of IDA that make the results more stable in high-dimensional settings. In Section 4.4, I will introduce the LV-IDA

method (Malinsky and Spirtes, 2017), which drops the sufficiency assumption, in more detail. However, in this section, I continue by assessing the standard IDA method in two steps: First, I evaluate how causal discovery methods like the PC algorithm used in IDA can recover causal structure from simulated data; then, I examine how accurately IDA can estimate effects from data originating from randomly generated causal graphs.

4.3.2 Discovering structure from simulated data

Since IDA relies on the PC algorithm discovering the causal structure in the first step, I initially assess how close the learned graphs are to the true DAG in a variety of simulation settings. While Maathuis et al. (2009) use the PC algorithm, in principle one can apply any causal discovery method that returns a CPDAG. Hence, I compare three of the most popular algorithms: PC, GES, and MMHC (e.g., Heinze-Deml et al., 2018a). Also, I include two naive baselines: one approach based on random guessing, the other based on marginal correlation.

For data generation and the implementation of most algorithms, I rely on the *pcalg* package in R (Kalisch et al., 2023). First, I generate the true random DAG with J nodes, where each node is connected to a subsequent node with probability p (the density parameter). For each edge, I draw the edge weight/coefficient randomly from a uniform distribution $U(0.1, 1)$. From this DAG and using linear relationships, I generate data with N observations following a multivariate normal distribution. Finally, I randomly shuffle the order of the variables in the data to ensure that the variable ordering does not contain information about the causal ordering.

In the next step, the causal discovery algorithms use this data to learn about the causal structure. In the PC algorithm, I test for zero partial correlation using a conditional independence test for normally distributed random variables with the significance level $\alpha = .01$. This value is common in the literature (e.g., Maathuis et al., 2009), and in my simulations, other values did not lead to substantially different results. For the score-based Greedy Equivalence Search (GES), I use the BIC score and the recommended defaults for other settings. For the hybrid method MMHC, I use the implementation from the *bnlearn* package with default options. I also implement two non-causal naive comparisons. The first is purely random guessing of a DAG with the same number of nodes and an edge probability $p = .5$. The second alternative is based on pairwise marginal correlation, which retains an edge if the correlation between a pair of variables is significant at the 1% level. Since correlation is an non-directional measure, this method always returns an undirected graph.

I follow Heinze-Deml et al. (2018a) in the way I compare the output of the different methods: For each pair of variables, I record whether one variable is the parent of the other (query *isParent*) or a possible parent (query *isPosParent*) in the learned graph. The query *isParent* is true if this relation holds in all DAGs of the CPDAG. If there are conflicting directions for the edge in question, *isParent* is false, but the query *isPosParent* is true. If the variable is not a parent in any of the DAGs, both *isParent* and *isPosParent* are false (for more details, see Heinze-Deml et al., 2018a). For each of these queries, I compute a true positive rate (TPR) and a false positive rate (FPR) and use those as the final evaluation metrics. Choosing this metric over, e.g., the structural hamming distance, has two reasons: First, knowing the parents of a particular node is crucial for our final goal of effect estimation, since the parents form a valid adjustment set (e.g., Pearl et al., 2016). Given the other IDA assumptions, if the discovered parents are correct, each effect estimate will be unbiased. If only the set of possible parents is correct, IDA's multiset will still at least contain one unbiased effect estimate. The second benefit of using these queries for evaluation is that they enable the comparison of CPDAGs with PAGs arising from settings where we do not assume sufficiency (see Section 4.4).

For each combination of simulation parameters, I generate 500 datasets and report the average rates across all results. In the main simulation setting, I hold the number of variables ($J = 7$) and the density parameter ($p = .4$) constant, while varying the number of observations between 100 and 1,000,000. The results show that GES and PC are indeed capable of learning the true MEC as the sample size grows, but the naive methods fail to learn causal structure (Figure 4.6). More specifically, for the query *isParent*, GES and PC find more correct parents as the sample size increases, ending up with around 63% true positives (Figure 4.6A) and close to 0% false positives in large samples (Figure 4.6C). MMHC performs similarly in terms of TPR, but incurs more false positives. The reason is that in this implementation of MMHC, the method returns a DAG instead of a CPDAG, and thus must direct all edges, while PC and GES can leave edges undirected if the MEC contains DAGs with conflicting directions. This is also why MMHC has identical rates for the query *isPosParent*, but PC and GES get close to a TPR of 1 for very large sample sizes (Figure 4.6B), which means that the discovered CPDAG contains the true DAG. However, we also see that the methods do not discover a unique DAG, since the TPR for the *isParent* query is far from 1 (Figure 4.6A) and the FPR for *isPosParent* indicates that other DAGs in the MEC contain wrongly directed edges (Figure 4.6D). Finally, the naive methods do not deliver useful information about the correct graph: On the one hand, random

guessing always leads to rates around 25%, on the other hand, the undirected correlation can only answer the query *isPosParent*, but generates far too many false positives.

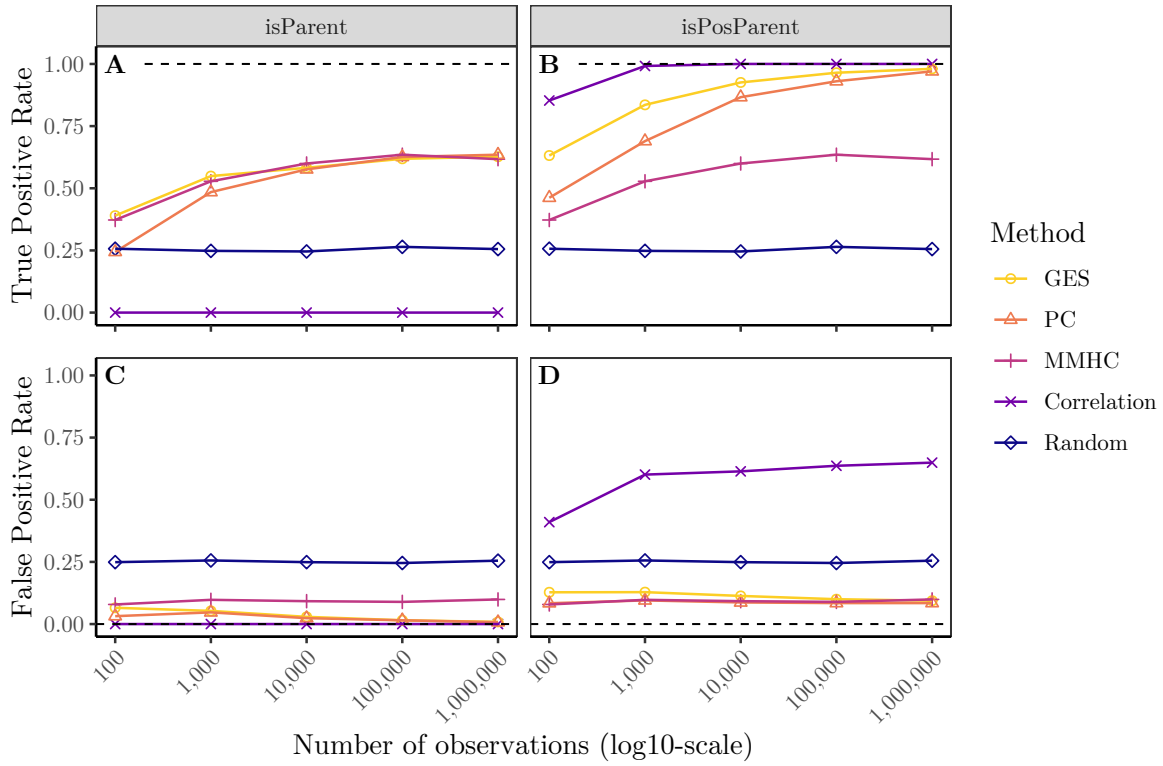


Figure 4.6: Results for the discovery of parents (query *isParent*, directed edge in CPDAG) and possible parents (query *isPosParent*, undirected edge in CPDAG) for different sample sizes. The dashed lines indicate the optimal values: a true positive rate of 1 and a false negative rate of 0. The number of variables is fixed at $J = 7$ and the density of the true graph at $p = .4$. Each point is the average rate across 500 simulated graphs and datasets.

In two additional settings reported in Appendix C.2, I hold the number of observations constant at 10,000 and vary the number of variables and the density of the graph, respectively. The performance of the causal discovery methods for the *isPosParent* query decreases as the number of variables increase (Figure C.2, Appendix C.2). For the query *isParent*, PC and GES perform best for graphs with 5-13 nodes. For settings with more variables, GES predicts more causal relationships compared to PC, which leads to more true, but also substantially more false positives, while the FPR is relatively constant for PC. In terms of density of the true graph, PC and GES again seem to work best for the *isParent* query for medium values (i.e., $p = .3 - .7$) (Figure C.3, Appendix C.2). For *isPosParent*, the performance of all methods decreases for denser graphs. In both queries, the causal discovery methods become more susceptible for false positives as the density increases, with GES again incurring larger FPRs compared to PC.

Overall, in my simulations, the score-based GES and the constraint-based PC algorithms outperform the hybrid MMHC method. GES and PC perform very similarly, but PC seems less prone to false positives. Both methods depend on the sample size being very large relative to the number of variables. As the sample size increases, GES and PC are capable of finding the correct set of possible parents and therefore the correct Markov equivalence class, which is the decisive input for IDA. Since no other method clearly dominates the PC algorithm in my simulations, I use it over the alternatives in my application of IDA.

4.3.3 Estimating effects from the discovered graphs

The goal of the IDA method is to estimate (bounds for) the total causal effect of one variable on another from an initially unknown graph. In the following, I assess how well the set of possible effects estimated by IDA can recover the true causal effect in various simulation settings. In my applications, IDA uses the output of the PC algorithm (i.e., a CPDAG) and the local method to estimate a multiset of possible total effects. For comparison, I also compute coefficient estimates from two OLS regressions: First, only regressing the outcome on the treatment ("Simple OLS"), and second, additionally adjusting for all other observed variables ("Full OLS"). Simple OLS will be biased if any of the remaining variables acts as a confounder, i.e., is a parent of both the treatment and the outcome. By contrast, OLS with all variables will be biased if any of the other variables is a mediator or collider ("bad control") on the path of treatment to outcome.

The data generation follows the same pattern as in the earlier simulations. Additionally, I randomly draw the identity of treatment and outcome variables. Given the causal ordering of all variables, I draw the treatment from the first $J-1$ variables: X_j^{treat} where $j \sim U\{1, ..., J-1\}$, and the outcome from all variables after the treatment: X_i^{out} where $i \sim U\{j+1, ..., J\}$. IDA returns a multiset of several possible causal effects, which I summarize in three different ways: the minimum, i.e., the lower bound ("IDA (min)"), the mean, i.e., the average of possible estimates ("IDA (mean)"), and the estimate in the set that is closest to the true effect ("IDA (best)"). While the latter is unknown in practice, it demonstrates whether IDA recovers the true effect within its multiset in my simulations. For each IDA summary measure and the two OLS methods, I report the absolute deviation from the true total effect, i.e., the absolute error or bias.

When holding the number of variables ($J = 7$) and the graph density ($p = .4$) constant while varying the number of observations, I find that the best estimate within IDA successfully

recovers the true effect for large sample sizes (Figure 4.7). Since the PC algorithm learns the true equivalence class in large samples (see Figure 4.6), at least one graph will contain the correct parents for adjustment, which IDA then uses (in addition to distributional assumptions) to estimate an unbiased effect. This result also implies that bounds derived from the minimum and maximum effect in the multiset will be correct. On the other end, the OLS methods have a much wider distribution at all sample sizes, since they at times either miss a confounder (Simple OLS) or adjust for a mediator/collider (Full OLS). However, deriving a feasible point estimate from IDA is challenging as well. In large samples, the distributions of the minimum and mean estimates seem to be concentrated closer to zero bias compared to the OLS estimates, especially Simple OLS. Nevertheless, due to many true effects being (close to) zero, all distributions are heavily left-skewed and some contain a large number of outliers (e.g., IDA (min)). Hence, a point estimate derived from IDA is not always clearly superior to OLS, and the choice of summary measure might depend on the application, e.g., the minimum is appropriate when the goal is the comparison of variable importances (Maathuis et al., 2009).

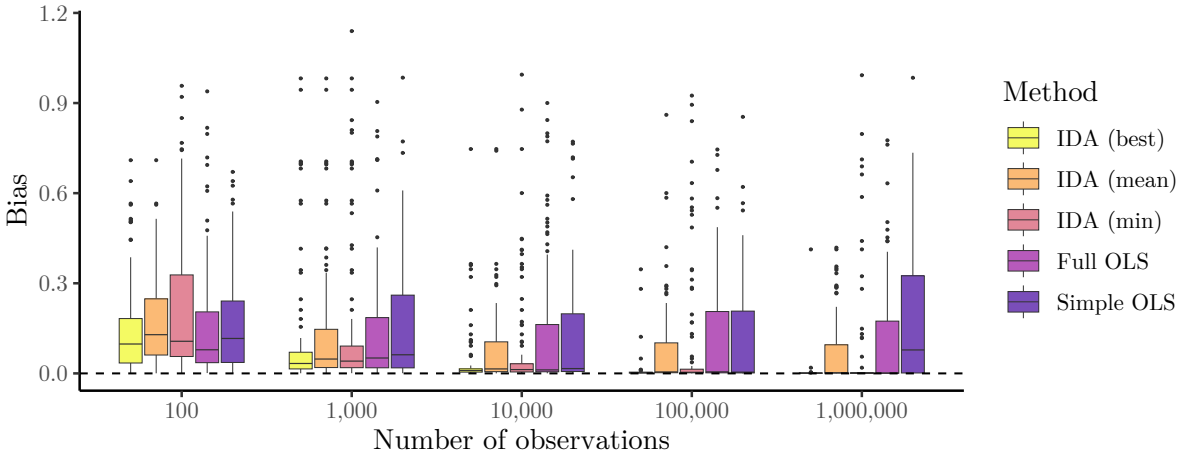


Figure 4.7: Results for causal effect estimation from unknown graphs for different sample sizes. The dashed line indicates the optimal bias of 0. The number of variables is fixed at $J = 7$ and the density of the true graph at $p = .4$. The boxplots show the distribution of the bias in the estimated coefficient across 100 simulated graphs and datasets. The IDA versions differ in how I summarize the multiset of effects: "best" (infeasible in practice) takes the effect closest to the true effect, "min" takes the minimum, "mean" takes the average of the multiset. I excluded the 11 largest outliers in terms of bias for better readability.

I report results for varying the number of variables and the density of the graph in Appendix C.3. In my simulations, IDA's benefits over OLS are especially pronounced for medium numbers of variables (e.g., $J = 7 - 13$, Figure C.4, Appendix C.3). For fewer variables, the probability of a confounder, mediator, or collider variable between treatment and outcome is low,

making the OLS methods feasible. For larger numbers of variables, the PC algorithm struggles to estimate the true Markov equivalence class, which leads to larger bias in the IDA estimates. However, if more observations were available, PC and thus IDA could also handle larger numbers of variables, and the benefits over OLS would be more visible due to the larger likelihood of missing or wrong adjustment variables. Similar observations hold when varying the density of the true graph (Figure C.5, Appendix C.3): IDA's advantage is largest for medium densities (i.e., $p = .4 - .6$), since low densities lead to few mistakes in OLS, and larger densities complicate the discovery of the CPDAG with the PC algorithm in limited sample sizes. However, even for large densities, IDA is not outperformed by these versions of OLS.

In sum, if the IDA assumptions hold and the number of observations is large relative to the number of variables and the graph density, the method can estimate reliable bounds on causal effects without knowing the underlying DAG. In several settings, IDA substantially outperforms OLS with no or all variables included, but is rarely outperformed by these alternatives. However, there are two concerns to consider when interpreting these results. First, in many applications, researchers will have some knowledge about the underlying graph and select their variables accordingly, avoiding blindly using all or no variables within OLS. This more realistic benchmark could outperform IDA, but IDA might be a valuable alternative if researchers know very little about the true graph. Secondly, the assumptions underlying this version of IDA are very strong (e.g., causal sufficiency and multivariate Gaussianity) and most likely do not hold in many applications. Thus, in the next section, I explore the consequences of violating the arguably strongest assumption, causal sufficiency, for both causal discovery and IDA.

4.4 Discovery and effect estimation without sufficiency

4.4.1 The LV-IDA algorithm

Malinsky and Spirtes (2017) propose an alternative to IDA that relaxes the causal sufficiency assumption, called "latent variable IDA" (LV-IDA). They argue that sufficiency is often not plausible in applications in social science and biomedicine, where some important confounding variables could be unobservable or unmeasured. LV-IDA retains the remaining assumptions of IDA, but allows for latent confounders by building on FCI instead of the PC algorithm for causal discovery. That is, it uses FCI to learn the PAG, cleverly enumerates the MAGs in the equivalence class, and tries to estimate the effect for each of the MAGs. Since MAGs learned by

FCI can contain bidirected edges and edges that are not visible (indicating potential unobserved confounding between two variables), some of the effects may not be causally identifiable via (generalized) backdoor adjustment. In these cases, LV-IDA will return a set of possible effects that includes "NA", denoting an unidentifiable effect. Malinsky and Spirtes (2017) state asymptotic consistency of LV-IDA, which here means that the estimated multiset contains either the true effect or "NA", if the true effect is not identifiable in the MAG. Similarly to IDA, the authors develop a faster "local" version of the algorithm that prioritizes finding the (possible) parents over enumerating all possible MAGs.

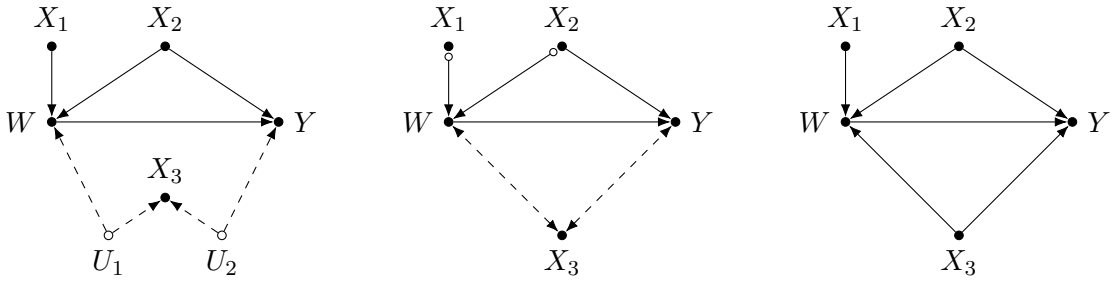
In a recent application, Lee et al. (2023) use LV-IDA to estimate effects of several variables on postoperative length of stay for patients undergoing cardiac surgery. Their data consists of 2,610 observations and 27 potential causes of the outcome of interest. Their results quantify effects of variables expected to be causes, suggest specific mediation mechanisms for further exploration, and demonstrate differences in results when using standard OLS instead of LV-IDA. This application is closer to many problems in social science in terms of dimensionality and assumptions and is a good example for how researchers can use causal discovery and subsequent effect estimation in practice. Also, throughout their article, the authors acknowledge assumptions and limitations like linearity, unidentifiable effects, and potential selection bias.

In the remainder of this section, I demonstrate and evaluate causal discovery methods and (LV-)IDA for cases where the sufficiency assumption does not hold. First, I illustrate the advantages of FCI and LV-IDA in a stylized setting where they can outperform alternative methods. Then, I introduce more randomness in the DGPs and evaluate how often settings like the first occur, or whether alternative methods are superior in other settings.

4.4.2 Discovery and estimation in a stylized setting

I begin by introducing a stylized example of a causal structure with latent variables where FCI and LV-IDA outperform PC and IDA (Figure 4.8). In this example, the goal is to learn the causal structure in order to estimate the effect of W on Y. X_1, X_2, and X_3 are further variables potentially relevant for adjustment. Given the true causal structure (Figure 4.8A), we know that we should adjust for X_2, which is a confounder; can, but do not have to, adjust for X_1; and *should not* adjust for X_3, which acts as a collider and would thus open a biasing path. The structure in the lower half of the graph is called an "M-bias" and implies that X_3 is a bad control, even though it could be a pre-treatment variable (e.g., Cinelli et al., 2024). X_1 is

not important for identification via the backdoor-criterion, but is crucial for causal discovery: It introduces additional v-structures, which helps to orient further edges. Additionally, the relationship $X_1 \to W$ makes the edge $W \to Y$ "visible" (Zhang, 2008), meaning that there cannot exist an unobserved confounder between W and Y, which enables identification via the generalized backdoor criterion (Maathuis and Colombo, 2015).



(A) True DAG **(B) Discoverable PAG** **(C) Discoverable CPDAG**

Figure 4.8: Stylized example of a causal graph where FCI and LV-IDA can outperform PC and IDA. The example is inspired by Cheng et al. (2024). The final goal is estimation of the effect of W on Y when the true DAG is unknown. **(A)** The true causal graph, where U_1 and U_2 are unobserved confounders, X_2 is an observed confounder, X_3 is a bad control. **(B)** The PAG learned by FCI in large samples. The circles indicate uncertainty about the edge orientation. The bidirected and dashed edges posit the existence of hidden variables. **(C)** The CPDAG learned by PC, which cannot detect unobserved confounders.

If the true causal structure is unknown, we first must learn it from the data before we can estimate the effect of W on Y. Given enough data, FCI learns the most important features of the true graph (Figure 4.8B). It correctly infers the unobserved confounders between X_3 and W/Y, respectively, and only indicates uncertainty about the edge marks between W and X_1/X_2, respectively. For the next step of effect estimation, it is crucial that the discovered structure implies the proper adjustment sets, which it does: All MAGs in the PAG correctly agree that X_2 is a good control and X_3 is a collider.

By contrast, the CPDAG discovered by the PC algorithm in large samples does not correctly identify X_3 as a bad control, because the underlying sufficiency assumption does not hold (Figure 4.8C). As a consequence, effects estimated based on this graph will be biased. Even though the number of wrongly inferred (possible) parents is small, the two false positives are enough to influence the adjustment set and thus bias the effect estimates (for a comparison of the causal discovery performance of these two methods, see Figure C.6 in Appendix C.4).

To demonstrate how the discovered causal structure affects the finally estimated coefficients, I generate data according to the causal structure of Figure 4.8A and compare the performance of LV-IDA, IDA, and OLS. All variables are jointly Gaussian, the true effect is 1, and I draw

all other coefficients randomly from a uniform distribution $U(0.1, 3)$. The relatively large upper limit of the distribution increases the average strength of the relationships and thereby facilitates the structure discovery. Except for the unobservable U_1 and U_2, all variables serve as input to the discovery and estimation methods. I apply IDA as in the previous section. For FCI, I choose $\alpha = .01$ as significance level for the conditional independence tests. Finally, I use the local LV-IDA algorithm[1] to estimate the effect of W on Y, given the FCI output. I summarize the multisets of IDA and LV-IDA by taking the mean; if the latter multiset contains an unidentifiable effect, the summary is also "NA".

For large sample sizes, LV-IDA successfully recovers the true causal effect in this setting (Figure 4.9). For smaller samples, FCI is unable to consistently discover the appropriate causal structure, which leads to biased or unidentifiable ("NA") effects in LV-IDA. The different versions of OLS are biased in different directions: Simple OLS incurs an upward bias, because it misses the confounder X_2; OLS with all variables is downwardly biased because it adjusts for the bad control X_3. As the sample size increases, IDA judges all variables to be good controls and thus performs similarly to Full OLS. In summary, in this special setting and if sufficient data is available, only LV-IDA can recover the true causal effect by both adjusting for the confounder and not adjusting for the bad control.

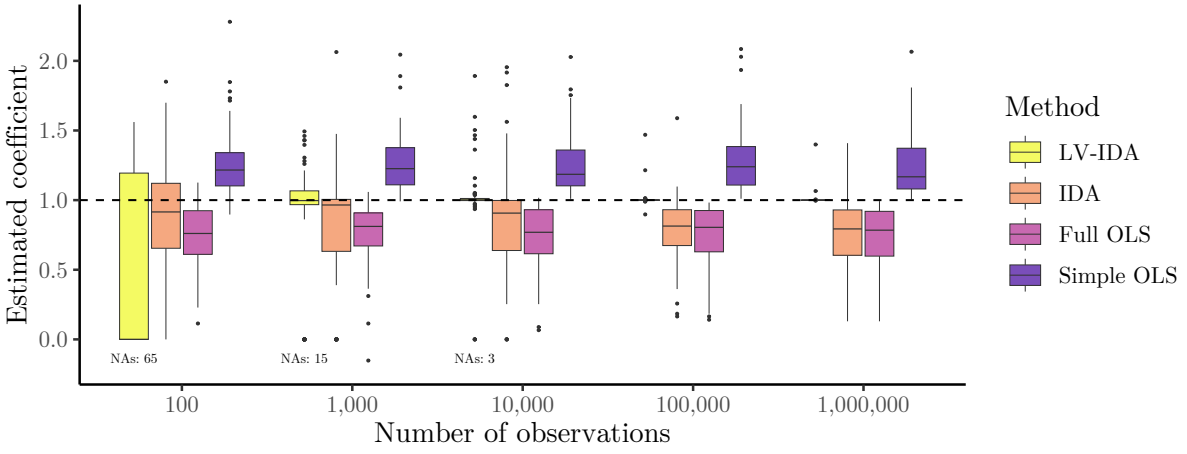


Figure 4.9: Results for causal effect estimation from an unknown graph with unobserved confounders for different sample sizes. The dashed line indicates the true causal effect of 1. The boxplots show the distribution of the estimated coefficients across 100 simulated datasets generated from the graph in Figure 4.8A. The number of "NA" values shows how many of the multisets of LV-IDA contained an unidentifiable effect. For all multisets without "NA" values (in both LV-IDA and IDA), the estimated coefficient is the average value.

[1] I use the R implementation of LV-IDA by Daniel Malinsky: `https://github.com/dmalinsk/lv-ida` (Retrieved June 18, 2024).

4.4.3 Estimating effects from randomly simulated graphs

In practice, the previous stylized setting might just be one special case, and other causal structures could prove to be more challenging for FCI and LV-IDA. Thus, in the following, I assess the performance of causal discovery and estimation methods on randomly generated graphs with unobserved confounding. For the data generation, I build upon the simulations of Section 4.3, but make the following adaptations: In the main simulations, I generate the initial graphs with 9 instead of 7 variables, which increases the probability of confounders between variable pairs. Then, I select the identity of unobserved confounders by randomly drawing with probability .5 from all variables that have at least two children and no parents (following Colombo et al., 2012), before I exclude the selected variables from the observed dataset. As a result, in most settings, there exist between one and three confounders in each graph, half of which are unobservable for the discovery and estimation methods. Finally, I draw the identity of the treatment and outcome variables from all observable variables as before.

For the analysis, I first assess the performance of different causal discovery methods in these settings with (potentially) unobserved confounders. I compare the PC algorithm, which assumes causal sufficiency, with the FCI algorithm, which does not make this assumption. As a third (infeasible) method, I include an oracle version of PC that has access to the unobserved confounder(s).[2] The results show that the unobserved confounding negatively influences the performance of PC using only the observable data compared to the oracle version, as it incurs fewer true positives and more false positives for both queries (Figure 4.10). On the other hand, FCI is more conservative than the feasible PC method: It detects substantially fewer true positives, but is also less prone to falsely detect possible parents. How consequential missed true parents or falsely found parents are for the subsequent causal effect estimation depends on the exact structure of the graph: A missed parent only introduces bias if it lies on a backdoor path from the treatment to the outcome, whereas a wrongly detected parent only biases the total effect if it functions as a mediator or bad control between treatment and outcome. Hence, in the second step, I assess the performance of effect estimation methods based on PC and FCI.

For causal effect estimation from unknown graphs, I use the same methods as in the stylized example, but add another summary measure for IDA and LV-IDA: In addition to the mean

[2] I can only evaluate FCI with the *isPosParent* query, since edges in PAGs only imply ancestral and not parental relationships (see also Heinze-Deml et al., 2018a).

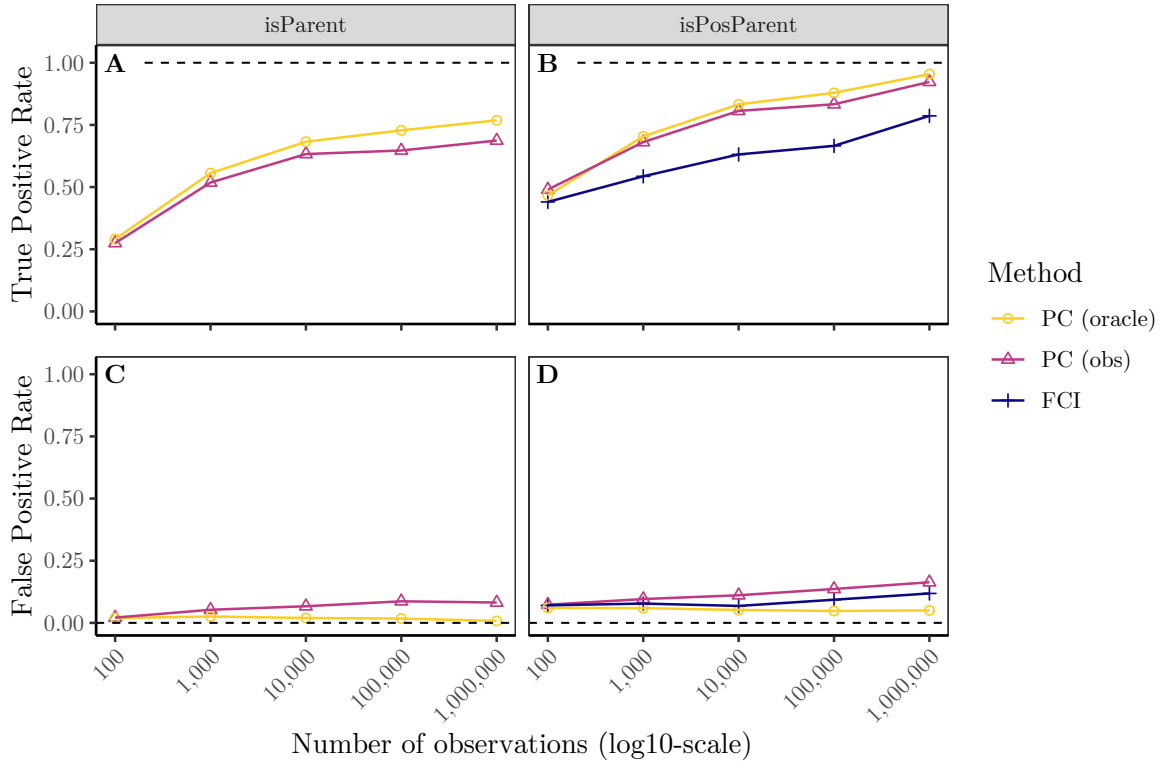


Figure 4.10: Results for the discovery of parents (query *isParent*, directed edge in CPDAG) and possible parents (query *isPosParent*, undirected (directed or circle) edge in CPDAG (PAG)) for different sample sizes under causal insufficiency. The dashed lines indicate the optimal values: a true positive rate of 1 and a false negative rate of 0. The number of variables is fixed at $J = 9$ (which may contain unobserved confounders) and the density of the true graph at $p = .4$. Each point is the average rate across 100 simulated graphs and datasets. "oracle" means PC has access to all variables; "obs" is the version of PC using only observable variables. PAGs only imply ancestral relationships, hence FCI cannot answer the *isParent* query.

effect of the multiset, I also report the accuracy of the best effect estimate within the multiset (Figure 4.11). I display the overall number of unidentifiable effects across simulation runs separately below the graph. The number in brackets indicates how many of these effects are indeed unidentifiable in the true DAG, i.e., in how many cases a variable necessary for backdoor adjustment is unobserved. For LV-IDA, I compute the "mean" measure after removing any "NA" values from the multiset, whereas I only plot a numeric "best" estimate if the multiset does not already contain a true "NA" value.

The results show that LV-IDA struggles to accurately estimate the effects from randomly generated DAGs (Figure 4.11). Across sample sizes, a large number of multisets include an unidentifiable effect, even though just a relatively small number of these (~22-52%) are in fact unidentifiable with the observed data, given the true DAG. With respect to the identifiable effects, even the distribution of the best LV-IDA estimates is wide for most sample sizes, although

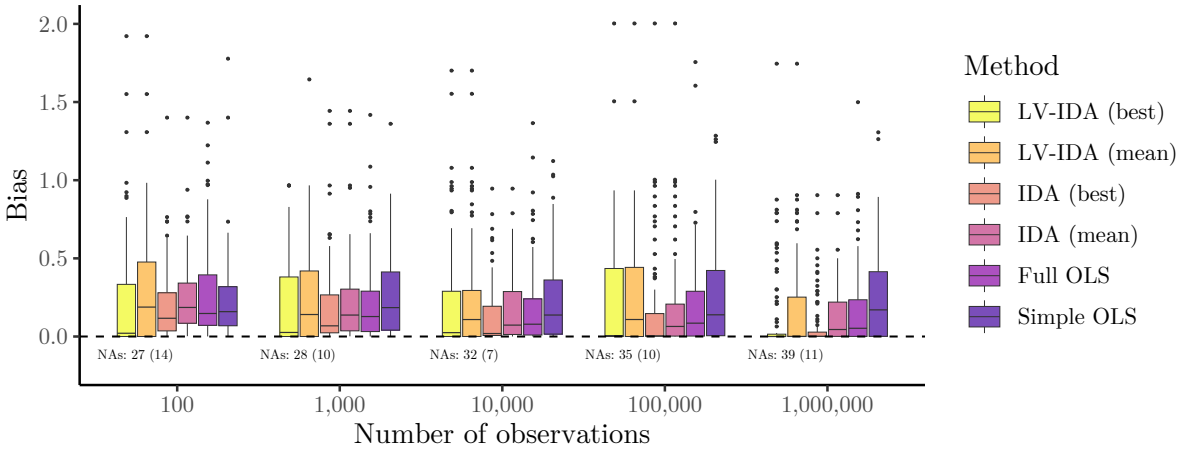


Figure 4.11: Results for causal effect estimation from unknown graphs for different sample sizes under causal insufficiency. The dashed lines indicates the optimal bias of 0. The number of variables is fixed at $J = 9$ and the density of the true graph at $p = .4$. The boxplots show the distribution of the bias in the estimated coefficient across 100 simulated graphs and datasets. The (LV-)IDA versions differ in how I summarize the multiset of effects: "best" takes the effect closest to the true effect, "mean" takes the average of the multiset. The number of "NA" values shows how many of the multisets of LV-IDA contained an unidentifiable effect, with the number in brackets indicating how many of these effects were truly unidentifiable with the observed data. I excluded the largest outlier in terms of bias for better readability.

the median bias is close to 0. Only for the largest sample size, the median bias of both LV-IDA measures is close to zero and the distribution of the best estimate is relatively narrow, indicating substantially more accurate estimates than the OLS methods. Nevertheless, even at this large sample size, many of the multisets incorrectly include an unidentifiable effect (presumably because of invisible edges). Despite the fact that the simulated settings often violate the sufficiency assumption, the standard IDA method often performs comparably to LV-IDA. The best IDA estimate improves as the sample size increases and is similar to the best LV-IDA estimate at the largest sample size, even though the results of both methods display various outliers with larger degrees of bias.

These findings suggest that the implemented sufficiency violations are not more consequential for IDA than for the other methods. Some of the unobserved confounders will not be on paths between treatment and outcome. Other unobserved confounders between treatment and outcome will bias the effects estimated with IDA, but also those of other methods, or might lead to an unidentifiable effect (in LV-IDA). Additionally, the risk of IDA including a collider or mediator is small, since false positives for PC are relatively rare, and a wrongly detected parent would in fact need to be a child *and* have a relationship to the outcome to cause any bias. A more complex setting of a bad control like the M-bias in our stylized example is rather unlikely to occur randomly. Hence, the inferior performance of FCI in the true positives outweighs the lower false-

positive rates, taking away most of the theoretical advantages of LV-IDA in random settings. Similarly to our results, Malinsky and Spirtes (2017) find that in higher-dimensional data, LV-IDA does not outperform IDA in randomly simulated graphs with unobserved confounders. They explain this by confounded variable pairs having weaker covariances in most simulations, making it both harder for FCI to detect the confounding and less consequential for IDA to miss a confounder.

4.5 Application

One classical social science application for causal inference methods is the question of the effect of 401(k) pension plan eligibility on net financial assets (Poterba et al., 1994, 1995). In much of the literature, the identification argument is that we can take 401(k) eligibility as exogenous after conditioning on a number of observable variables, among which income is the most important one (see also Chernozhukov et al., 2018).

When researchers use this identification strategy, they typically include all other observed variables for adjustment in their estimation model. By contrast, I take an alternative approach, where I use causal discovery methods to learn the underlying causal structure, before I employ (LV-)IDA to find the appropriate adjustment sets and estimate a set of possible effects. In the process, I evaluate the plausibility of the learned causal structure and compare how the effect estimates relate to the ones obtained by alternative estimation methods.

In addition to the variables for 401(k) eligibility and net financial assets, the data (made available by Chernozhukov et al. (2018)) contains measures for age, income, family size, and education, as well as indicators for marriage, two-earners status, defined benefit pension, IRA participation, and home ownership. The dataset thus consists of 11 variables and has 9,915 observations. A naive simple OLS regression, not adjusting for any covariates, results in an average treatment effect estimate of \$19,599. By contrast, adjusting for all observed covariates linearly within OLS attenuates the effect estimate to \$5,896. Instead of linear adjustment via OLS, Chernozhukov et al. (2018) use double/debiased machine learning (DML) for flexible confounding adjustment, which results in estimates from \$6,830 to \$9,247, depending on the model and ML method employed. In the following, I retain the linearity assumption of OLS, but start without any knowledge of the underlying causal structure, try to discover the graph, and subsequently estimate the effects.

4.5.1 401(k): Discovery and estimation under causal sufficiency

In a first step, I make the – admittedly strong – assumption of causal sufficiency, thereby ruling out unobserved confounding between any two variables in the underlying graph. This enables the use of the PC algorithm for causal discovery and of IDA for estimation. My first observation is that the results are highly dependent on the ordering of the variables in the dataset. This is a known issue of the PC algorithm, which is why Colombo and Maathuis (2014) developed a fully order-independent version of PC, which I use for my analysis. Applying the PC algorithm with the parameter $\alpha = .01$ leads to a CPDAG where *401(k) eligibility* $\leftarrow$ *net financial assets* (see Figure C.7, Appendix C.6), which is the reverse direction from what we would expect, and implausible, because net financial assets are measured later. Naturally, this would also imply an effect size of 0, since *401(k) eligibility* is no cause of *net financial assets* in this graph. When setting $\alpha = .05$, the PC algorithm becomes more uncertain about this edge direction and leaves it undirected, allowing for both directions within its Markov equivalence class (Figure 4.12). In fact, many of the edges remain undirected, leading to a relatively large equivalence class.

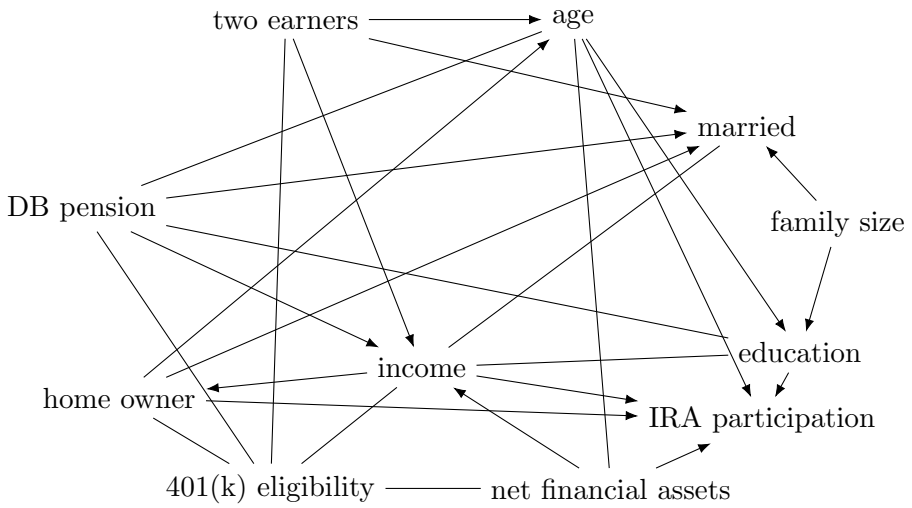


Figure 4.12: CPDAG discovered by order-independent PC algorithm with $\alpha = .05$. The undirected edges indicate uncertainty about the edge orientation.

However, even a cursory inspection of the resulting CPDAG reveals some implausible edges. For instance, *age* is an effect of being a home owner and having two earners, whereas we would not expect age to be caused by any other variables. Other edges, e.g., *income* $\rightarrow$ *home owner* are more plausible. Additionally, the CPDAGs returned by the PC algorithm for this application are "invalid", since some of the graphs in the equivalence class contain cycles and are thus not

"acyclic". This can be the consequence of hidden variables or sampling errors in finite data (Kalisch et al., 2023).

While the global IDA algorithm does not work on cyclic graphs, the local version does, since it only uses the local information about (possible) parents and can thus ignore cycles. The multiset returned by the local IDA algorithm for the effect of 401(k) eligibility on net financial assets is {19,559; 4,697; 19,536; 16,151; 0; 19,534; 5,607; 4,210; 0; 5,561}, which contains 0 (when *401(k) eligibility* ← *net financial assets*), but also the OLS estimate without any adjustment (19,559) and estimates close to OLS with adjustment for all variables (e.g., 5,607).

Overall, the implausible results in the discovered graph should lead us to critically question the appropriateness and usefulness of PC and IDA for this causal question and dataset. Most likely, the underlying assumptions, mainly causal sufficiency and linearity, are not valid for this application. If this is the case, we would expect implausible results as output of PC. Even if the discovered graph was plausible, the large multiset estimated by IDA is hardly informative, as it covers a very large range of possible effects.

4.5.2 401(k): Discovery and estimation without causal sufficiency

We can address at least one of the limitations of PC and IDA by using FCI and LV-IDA instead. That is, we no longer must assume that there is no unobserved confounder between any two variables. When working with the 401(k) eligibility data, Chernozhukov et al. (2024) consider various causal structures, all of which contain at least one latent cause somewhere in the graph. As examples for potentially unobserved variables they mention firm characteristics, an employer match amount, and other not further specified confounders. Depending on the exact position of these latent variables in the causal graph, they may or may not hinder causal identification.

For learning the causal structure, I now use the order-independent version of FCI, first with the parameter α set to .01. The resulting PAG (Figure 4.13) indicates the presence of latent variables between multiple pairs of variables (bidirected edges), as well as uncertainty in several edge directions (circles). For the focal relationship between 401(k) eligibility and net financial assets, the PAG is uncertain of both edge marks, allowing for a directed edge in either direction or a fully confounded relationship. Surprisingly, the relationship of both variables with the income node is supposedly fully confounded, making income a bad control (similar to the stylized example in Figure 4.8B). This is contrary to the original argument in Poterba et al. (1994), which views income as the most important confounding variable. Other edge directions

are more plausible compared to the previous CPDAG; e.g., *married* is now an ancestor of *family size*, and having *two earners* no longer is a cause of *age*.

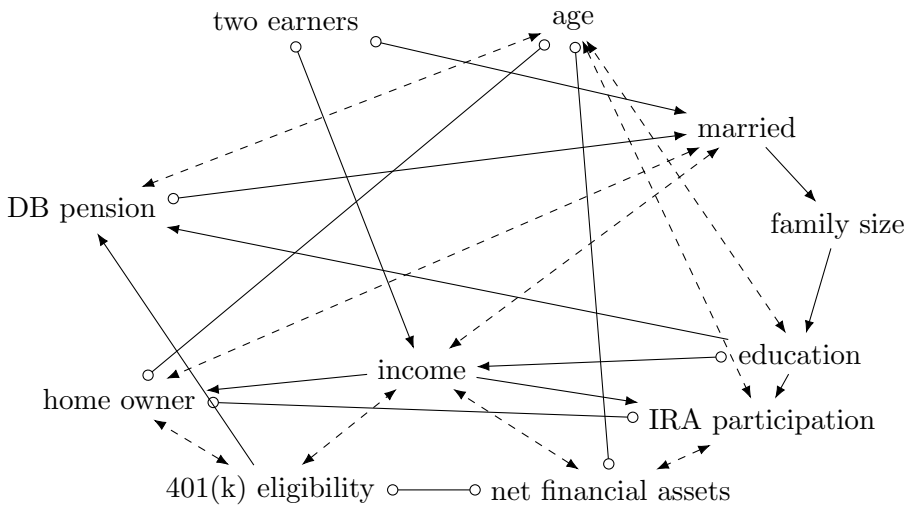


Figure 4.13: PAG discovered by order-independent FCI algorithm with $\alpha = .01$. The circles indicate uncertainty about the edge orientation. The bidirected and dashed edges posit the existence of hidden variables.

In the next step, estimating possible effects of 401(k) eligibility on net financial assets with the local LV-IDA algorithm results in the multiset {0; 0; 0; 0; 19,559; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 19,559; 19,559; 19,559}. It does not contain any unidentifiable effects, since in every MAG, the relationship is either fully confounded or the directed edge is visible. The many zero effects result from MAGs with a bidirected edge between treatment and outcome, or a directed edge from outcome to treatment. The other larger estimates are the same as for simple OLS, originating from MAGs that suggest that there are no backdoor paths to block.

When setting $\alpha = .05$, the PAG found by FCI contains a bidirected edge between 401(k) eligibility and net financial assets, which implies that this relation is fully confounded without a directed path between the variables, leading to a single effect estimate of 0 in LV-IDA (see Figure C.8, Appendix C.6).

While FCI and LV-IDA no longer rely on the sufficiency assumption, these results still raise questions about the appropriateness of the methods and their assumptions for this application. A substantially larger sample size might improve the discovery process, but there are likely additional issues as well. For example, my implementation of the methods assumes linear relationships between all variables. By contrast, Chernozhukov et al. (2018) use DML for effect estimation in this application because they suspect some nonlinearities (although their final estimates are broadly consistent with the original ones). Additionally, violations of the faithfulness

assumption could lead to mistakes in the conditional independence tests, but are hard to assess in this example. I will discuss these limitations and potential remedies in more detail in the following discussion.

4.6 Discussion

4.6.1 Summary and limitations

This article has explored the use of causal discovery methods in the social sciences, when the final goal is not only learning about causal structure, but also using the structure to estimate causal effects. In my simulations, I found that given causal sufficiency, appropriate conditional independence tests, and large sample sizes relative to the number of variables and the graph density, discovery algorithms like PC or GES were able to find the correct Markov equivalence class of DAGs. As a consequence, the multiset of possible effects estimated by IDA contained – among others – the true effect. Also, in many settings, the bounds of the multiset were more informative than estimates obtained from naive regression methods unaware of any causal structure. If, however, the number of variables or graph density was high or the sample size small, the methods became much less informative. Without causal sufficiency, the alternative methods FCI (for discovery) and LV-IDA (for estimation) were less prone to detect erroneous adjustment variables, but also missed some important ones. Additionally, effect estimation with LV-IDA often led to unidentifiable effects, even if the effect was identifiable in the true underlying DAG. When applying both pairs of methods to data about the effect of 401(k) eligibility on net financial assets, I found a strong sensitivity to parameter choices, and several suggested relationships that were counterintuitive. Also, the estimation results of (LV-)IDA were largely implausible or uninformative compared to the original results obtained from models built with domain knowledge.

These rather negative findings are based on a specific selection of simulation settings, methods, and implementations, which limits the ability to directly generalize from my results to the field of causal discovery as a whole. First, throughout my simulations, I assume continuous, Gaussian variables and linear relationships with homogeneous effects. Future studies could assess how other variable types, distributions, and nonlinear relationships would influence the accuracy of the results. For such settings, there is a variety of very flexible nonparametric conditional independence tests (e.g., Heinze-Deml et al., 2018b; Shah and Peters, 2020), although

they typically need many observations and significant computing resources. In fact, there are extensions of IDA that try to relax the Gaussianity and linearity assumptions (e.g., Mahmoudi and Wit, 2018; Nandy et al., 2017; Teramoto et al., 2014). Furthermore, given a causal discovery method that uses nonparametric conditional independence tests, an interesting modification of IDA could be to replace OLS as estimation method by a more flexible, ML-based estimation approach. For example, one could use PC with nonparametric conditional independence tests for causal discovery, then enumerate all possible graphs, derive the appropriate adjustment sets, and flexibly adjust for these variables with DML (Chernozhukov et al., 2018, see Chapter 2 of this thesis), resulting in a multiset of possible effects under substantially weakened assumptions in discovery and estimation.

Second, in the 401(k) application above, the assumption of linear relationships could also be one explanation for some of the implausible results. Further research could analyze how using more flexible methods and (nonparametric) conditional independence tests for mixtures of discrete and continuous data would alter the results, although the sample size might limit the flexibility to some degree. Additionally, some implementations of causal discovery algorithms allow a fine-grained inclusion of background knowledge, i.e., researchers can force or forbid certain relationships between variables (e.g., edges going into a variable such as age), which could further improve the performance and avoid some implausible results.

Finally, I limited my analysis to a selection of causal discovery and estimation methods. For example, I did not apply any discovery methods based on functional causal models (e.g., Peters et al., 2017) or continuous optimization (e.g., Vowels et al., 2022). While the former set of methods typically return a single DAG, they crucially rely on assumptions about specific functional forms and/or distributions, which I deem hard to assess in practice. Additionally, there are several alternative methods to (LV-)IDA, attempting to construct a full pipeline from causal structure learning to effect estimation in various settings, often under even stronger or more questionable assumptions (for a review, see Cheng et al., 2024).

4.6.2 Causal discovery as a tool for applications in the social sciences

Given my results and insights along the way, what do I conclude about causal discovery as an instrument for real-world data in the social sciences? Is it right to be critical of attempts to discover causal structure from purely observational data, or are the necessary assumptions comparable to the ones made by traditional identification and estimation methods? In the

following, I lay out my central observations from literature, simulations, and applications, before I summarize my current perspective on the applicability of causal structure learning methods.

First, a focal theme – as for all causal inference methods – is the necessity of strong, typically untestable assumptions. Although causal discovery methods in some sense relax the assumption that researchers know the true causal structure before they estimate effects, they replace this assumption with others that are not necessarily weaker. For example, the assumption of causal sufficiency is much stronger than the traditional assumption of unconfoundedness, since it rules out latent confounding between any pair of variables, not only between treatment and outcome. The assumption of faithfulness seems very difficult to assess. While an exact canceling out of causal pathways appears unlikely at first, finite samples might already prevent the detection of small differences between the paths, which Uhler et al. (2013) argue can happen surprisingly often. Alternative methods based on functional causal models have the advantage that they avoid making a faithfulness assumption. However, they instead rely on assumptions about functional forms and/or noise distributions, which seem difficult to assess with domain knowledge in the social sciences. The assumptions discussed above are only a selection, and researchers must be aware of the specific ones made by the method they want to employ. Then, in the context of their particular application, they can reason whether these assumptions are plausible. I agree with Peters et al. (2017, Preface) that this will only be the case in rather limited situations with the currently available methods.

A second theme is the need for large sample sizes (very roughly: $N > 10,000$ observations for $J < 10$ variables) for virtually all causal discovery methods. In general, methods based on functional causal models require even more observations than constraint-based or score-based methods (Malinsky and Danks, 2018), but the flexibility of nonparametric conditional independence tests in constraint-based approaches also crucially depends on the sample size (Shah and Peters, 2020).

Thirdly, the most established and applied causal discovery algorithms are developed for settings with cross-sectional, i.i.d. data, whereas much of the data in social science has a time series or panel structure (for a brief overview on causal discovery with time series, see, e.g., Spirtes and Zhang, 2016). While the time structure provides some benefits because of known constraints (the future cannot cause the past), it also comes with unique challenges (Malinsky and Danks, 2018).

A fourth observation is that many causal discovery (and estimation) methods such as PC and IDA have received attention and application success primarily in biology and the medical domain (e.g., Cheng et al., 2024; Maathuis et al., 2009), but are largely unknown in the social sciences. I presume this is due to the fact that in many biological systems, researchers can measure many factors, but often do not have a clear notion of how these factors causally relate. By contrast, in social science, some variables are inherently not measurable (making causal sufficiency implausible), while at the same time, researchers can more confidently construct the crucial parts of a causal model from theory and domain knowledge. Additionally, the scarcity of true zero effects contradicts the usefulness of conditional independence tests (Gelman, 2011).

Finally, traditional causal discovery methods are typically not concerned with quantifying statistical uncertainty in the structure learning process. This is even the case for methods like IDA, which use the same data for discovery and estimation, which invalidates the usual OLS standard errors (Witte et al., 2020). For this reason, the literature typically cautions against interpreting the standard errors of standard procedures (e.g., Lee et al., 2023) or suggests alternative approaches for valid post-discovery inference (e.g., Chang et al., 2024).

In summary, my current perspective on causal discovery methods is that they do not represent a fully data-driven alternative for theory and domain knowledge, since learning from purely observational data has substantial limitations when it comes to causal questions. The assumptions made by traditional identification and estimation methods often seem weaker, or at least more straightforward to assess and discuss for specific applications. Nevertheless, causal discovery algorithms might complement conventional causal inference methods in specific settings. Their application might be most appropriate in new areas or fields where many factors are measured, but there is little prior knowledge about causal relationships between these factors. There, causal discovery could serve as a tool for building new hypotheses, which researchers can subsequently validate with experiments or conventional causal methods (Dawid, 2010; Maathuis et al., 2009).

Chapter 5

Discussion

The availability of massive amounts of data and the development of increasingly powerful machine learning algorithms has influenced research in many scientific fields. Data-driven methods can often find complex patterns in large and unstructured datasets, enabling accurate predictions in settings where traditional statistical methods struggle. However, most ML methods are not intrinsically capable of answering causal questions, which is arguably the main goal of many scientific disciplines, especially the social sciences. To draw accurate causal inferences from observational data, researchers need to rely on assumptions about causal structure, identification, and estimation. They typically ground these assumptions in the theory and expert knowledge from the respective field. Nevertheless, a recent literature has suggested that using ML could relax some of the assumptions researchers conventionally make in causal inference. If feasible, this has the potential to make causal research more credible by relying on weaker assumptions compared to traditional methods.

The goal of this dissertation was to explore the promise and pitfalls of some of these new developments for applied research in the social sciences. For this purpose, I reviewed the methodology, empirically evaluated the performance of various novel approaches, adapted them to relevant new settings, and illustrated their application to datasets from social science. In the following, I briefly summarize the findings of each of the previous three chapters, connect the common insights in an overarching discussion, and give an outlook on potential future developments, before finally concluding the thesis.

5.1 Summary

In Chapter 2, we reviewed and evaluated the "double/debiased machine learning" (DML) framework by Chernozhukov et al. (2018), aiming to illustrate its potential and limitations and to guide applied researchers in the many choices they face when applying it. In extensive simulations, we compared different ML methods within DML and demonstrated how DML with flexible ML algorithms like gradient boosting can appropriately adjust for a variety of (non-)linear confounding relationships. This enables researchers to relax functional form assumptions often made in the process of causal estimation. However, we also highlighted that the method still crucially depends on standard assumptions in the steps of causal structure and causal identification. Furthermore, we provided evidence for how the predictive accuracy in the treatment and outcome models can help researchers to choose between ML algorithms. In our application, DML with

flexible ML algorithms consistently estimated absolutely larger effects of air pollution on housing prices, suggesting the presence of nonlinear confounding not easily captured by a parametric model. Lastly, we demonstrated the impact of further decisions in the implementation of DML and derived actionable recommendations.

The goal of Chapter 3 was to adapt DML – originally developed for cross-sectional data – to settings with panel data. We first discussed the challenges of adapting DML to this data structure in the presence of nonlinear observed confounding in addition to unobserved heterogeneity. In our subsequent simulations, various adjustments of DML's cross-fitting procedure did not affect the accuracy of the point estimates. When we considered multiple intuitively appealing adaptations of DML to remove the unobserved heterogeneity, we found that most approaches failed in the more complex data-generating processes. However, we discovered that an adaptation that uses predictors based on the correlated random effects approach within DML performed well across settings. The advantage of this approach against others depended on the sample size being large relative to the number of observed confounders. We also demonstrated that the relationship between the unobserved heterogeneity and the observed confounding crucially affects the estimates of alternative methods. Our suggested approach allows researchers to relax two assumptions in the causal inference workflow. First, in the causal identification step, we can relax the assumption of no unobserved confounding to no unobserved *time-varying* confounding, as the panel structure allows the removal of unobserved time-constant heterogeneity. Secondly, similar to the original DML approach, we can relax assumptions in the estimation process about the functional forms of the observed confounding.

Chapter 4 explored how much we can learn about causal structure from observational data in social science. After surveying the foundational concepts of causal discovery, I reviewed, evaluated, and applied two methods that aim to discover structure and subsequently estimate causal effects. The results demonstrated that learning causal structure with such methods is a very hard problem, only possible in very large samples and under rather restrictive assumptions. I argued that in many social science applications, these assumptions will be stronger and less plausible than alternative assumptions derived from theory, but might be a reasonable starting point in fields with very little prior domain knowledge.

5.2 Overarching themes

In the introduction of this thesis, I raised the question of whether data-driven ML methods can contribute to solving not only correlational and predictive, but also causal research problems. The subsequent three chapters set out to explore answers to this question by focusing on specific methods that use ML at different points in the causal inference workflow. For the remainder of this discussion, I will take a step back to address the question in the bigger picture, using both my results and other insights from the recent literature. While the three studies of this dissertation do not comprehensively cover all developments of the field, they contain at least four common and interrelated themes, which I will discuss in the following. Table 5.1 summarizes the core aspects of these themes.

Table 5.1: Characteristics of the causal inference workflow with respect to theory, ML, and assumptions

	causal structure ⟶	**causal identification** ⟶	**causal estimation**
Need for theory	high	high	low/moderate
Potential for ML contribution	low	low/moderate	high
Strength of altern. assumptions	strong	strong	weaker

1. Great potential for ML to relax estimation assumptions

First, Chapters 2 and 3 have shown that if we use ML within appropriate frameworks, it can help to relax estimation assumptions, e.g., assumptions about functional forms of confounding relationships. In addition to DML, other approaches have also successfully relaxed estimation assumptions by incorporating ML. For example, Wager and Athey (2018) proposed "causal forests", which enable the flexible estimation of heterogeneous effects without assuming and modeling the specific heterogeneity beforehand. Given this potential of flexibly learning functional relationships when estimating causal effects, some researchers have criticised the persisting traditional practice of specifying fully parametric models, even in the absence of strong theoretical arguments for particular functional forms (e.g., Dang et al., 2023). Imposing these statistical assumptions is often not necessary, and a misspecified parametric model can result in substantially biased estimates (e.g., Wooldridge, 2012, Chapter 9). Indeed, our results in Chapter 2

demonstrate both the consequences of functional form misspecifications, and the potential of ML-based estimation approaches, even for small sample sizes. Hence, I would similarly recommend that applied researchers use data-driven estimation approaches for settings where we can understand and show their feasibility, as discussed in Chapter 2. However, Chapter 3 has demonstrated that it is not always straightforward to replace parametric models with ML-based methods. There is active research for going beyond settings where covariate adjustment in cross-sectional data is sufficient, with progress in the identification strategies of instrumental variables, difference-in-differences, and regression discontinuity designs (see Chernozhukov et al., 2024).

2. Limited potential for ML to relax structure and identification assumptions

While using ML for estimation within other identification strategies is promising, there is arguably limited potential of data-driven approaches to directly relax assumptions about causal structure and identification. Chapter 2 showed that ML-based estimation approaches such as DML still rely on causal structure and identification assumptions, e.g., the absence of unobserved confounding and correct classification of good and bad controls. In the context of causal identification, I currently see the main potential of ML in its incorporation within various existing identification strategies, as opposed to it enabling new identification results. We considered adaptations of DML for settings where identification is possible with panel data (Chapter 3), others have worked on the strategies mentioned above (see Chernozhukov et al., 2024).

However, researchers have also suggested more ambitious methods that directly attempt to relax identification assumptions. One prominent example I spent some time exploring is the "deconfounder" approach proposed by Wang and Blei (2019). This method attempts to relax the unconfoundedness assumption in settings with many causes/treatment variables. The basic idea is that an unobserved confounder influencing many causes leaves an imprint on these observed variables, which factor models from unsupervised ML can recover. Subsequently, researchers can adjust for this recovered substitute confounder and thereby relax the unconfoundedness assumption, now merely assuming the absence of unobserved confounders that influence only a single cause. While the approach sounds promising and would be valuable for many social science applications, several researchers have criticized it for implicitly relying on assumptions that are as strong or stronger than unconfoundedness (e.g., Ogburn et al., 2019), and have shown naive regression-based approaches to perform similarly under weaker assumptions (Grimmer et al., 2023). In another development, Burauel (2023) proposed a test to evaluate the validity of

instrumental variables based on principles from the causal discovery literature. A feasible test could support and facilitate theoretical arguments about instrument validity. The approach relies on linearity, constant treatment effects, high-dimensional covariates, and more technical assumptions. An independent assessment of the boundary conditions of this method is still missing, but it is currently not obvious whether the additionally imposed assumptions are weaker than arguments for instrument validity from theory. While there are certainly further approaches that try to utilize ML for causal identification, I am currently not aware of any approaches that can substantially relax identification assumptions to make applied causal research more credible.

Similarly, Chapter 4 has shown and discussed the difficulty of learning causal structure in a mainly data-driven way. While causal discovery is in principle possible, it depends on strong assumptions that will often be questionable or at least more difficult to assess than arguments from theory.

To summarize, from my current perspective, using ML for the steps of causal structure or causal identification will probably always rely on assumptions that are equally strong or stronger than those imposed by traditional methods. Therefore, applied researchers should be hesitant when facing new developments that claim data-driven causal identification or structure learning under substantially weaker assumptions.

3. Exchanging and assessing different assumptions

The third theme emerges from the previous two: There is no "assumption-free" causal inference from observational data. Every method relies on assumptions, but these assumptions might differ between different approaches. In a sense, much methodological development in causal inference is about "shifting around" assumptions or trading a set of assumptions for alternative ones. Thus, the crucial question for applied researchers is how they can compare and assess the different assumptions in the context of their application.

This is often simpler for estimation assumptions than for identification and structure assumptions, which are typically not testable from observational data. For example, in Chapter 2, we proposed a simple test for evaluating functional form assumptions within DML: We could compare the predictive accuracy of a parametric model within DML with the predictive accuracy of a flexible ML method, ultimately choosing the one with smaller prediction error. Note, however, that this relies on the identification assumptions being correct and is only a relative measure, i.e., we can compare two alternative predictive methods but cannot easily judge how close each

is to finding the correct functional form. Nevertheless, as years of research and applications in ML have shown, we can consider the assumption that a flexible ML method fits an unknown functional form to be substantially weaker than assuming a correctly specified parametric model (e.g., Dang et al., 2023).

By contrast, assessing assumptions about causal structure or identification is much harder and crucially relies on theory and expert knowledge. When applying methods with specific assumptions, researchers need to explain why they consider these assumptions to be plausible in the context of their particular application (e.g., Athey, 2019). This is most convincing when articulating a consistent theory based on findings from previous literature, which other researchers can subsequently challenge if necessary. Within a certain application, some assumptions might be more plausible than others, and researchers should choose their approach such that they can justify the underlying assumptions. While the classical approach of making, assessing, and comparing assumptions based on theory is straightforward for some assumptions, it might be more obscure for others. For example, in the 401(k) application of Chapter 4, theory (or rather common sense) tells us that home ownership causing a person's age is implausible. Also, in many instrumental variable settings, it is much easier to argue for the (conditional) exogeneity of the instrument than for that of the treatment variable. However, other assumptions seem more difficult to assess even with significant domain knowledge. For example, when assessing the faithfulness assumption in causal discovery, we could argue from theory whether there might be both a positive and a negative path between two nodes, but it will be hard to make statements about whether they might cancel each other out. Also, arguing about the various paths between all pairs of nodes from domain knowledge – thereby essentially constructing a causal graph – practically renders the whole enterprise of causal discovery superfluous. Similarly, assumptions in functional model based causal discovery about specific underlying functional forms and noise distributions seem difficult to assess from theory.

Another important approach for assumption assessment is to quantify how severe violations of the main assumptions would need to be to invalidate the conclusions of a particular study via *sensitivity analysis* (e.g., Cinelli and Hazlett, 2020), which was recently also combined with debiased ML methods (Chernozhukov et al., 2022a). Finally, the only way to really test assumptions about causal structure or identification is to conduct randomized experiments, which is unfortunately often not feasible (Athey, 2019).

4. The need for transparency of assumptions in development and application

The final overarching theme is the importance of clear and transparent communication both when developing and when applying causal inference methods. Since all causal research relies on assumptions, researchers should explicitly point out the substantial assumptions underlying their approach. This way, readers can independently judge whether they deem those assumptions and the subsequent results to be credible. Transparency about assumptions is arguably much more important in causal inference compared to (supervised) ML, where objective performance measurements (e.g., test set error) are easier to obtain. Moreover, there is often a gap in the communication between the researchers developing new methodology and the researchers applying it. While some assumptions appear obvious to statisticians, applied researchers might fail to realize their importance if they are not explicitly stated. For example, when using ML to adjust for confounding, applied researchers might not be aware of the risk of adjusting for bad controls or missing an unobserved confounder (Chapter 2). Similarly, they could naively apply the original DML algorithm to panel data, unaware that it assumes cross-sectional data and requires adaptation for panel data settings (Chapter 3). In their critiques of the deconfounder, Ogburn et al. (2019), Grimmer et al. (2023), and others point out the strong assumptions underlying the approach, many of which are not explicit in the original paper. Hence, researchers developing new causal inference methods should explicitly state the underlying assumptions, ideally with terminology accessible to applied researchers. Then, applied researchers should transparently communicate what they assume by using a particular method for their application and argue for the plausibility of those assumptions.

5.3 Outlook

Before I conclude, I briefly consider whether and how further developments in ML could alter the conclusions of this thesis. I have argued that data-driven methods are currently valuable for estimation, but limited in their ability to substantially assist in defining causal structure and achieving causal identification. However, it is conceivable that the advent of large language models (e.g., OpenAI et al., 2024) could also have an impact on these two steps (e.g., Kıcıman et al., 2024). The text corpora underlying these models could contain the theory and domain knowledge necessary to substantiate causal inference assumptions. Researchers (or other algorithms) could thus ask an LLM to summarize the current state of the literature around a

particular problem, query it for specifics about the causal structure implied by previous theory, and request an assessment of the plausibility of specific assumptions that would enable causal identification. Such an approach would require the models to be trained on carefully curated high-quality expert literature. Also, it would likely not replace the thorough thinking of an expert researcher, who should still assess the output of the LLM, fill in potential gaps, and maybe extend the current theory to the specific application under consideration. Thus, if we define "observational data" to include text data generated by human authors, it seems more feasible than ever to learn more than correlations from data. Such an approach still critically relies on human input and domain knowledge, requires appropriate prompting, sometimes fails unexpectedly (Kıcıman et al., 2024), and will thus not constitute the end of theory, but it is an interesting avenue of future research to consider.

5.4 Conclusion

In conclusion, purely data-driven approaches have clear limitations when it comes to answering causal questions, there is still no "end of theory" (Anderson, 2008) in sight. However, principles and algorithms from machine learning have become valuable tools to support traditional causal inference approaches and make their estimation results more credible (e.g., Athey, 2019). Hence, theory-based traditional causal inference and data-driven ML are not competing methodologies, but should complement each other to deliver more robust answers to causal questions (Leavitt et al., 2021). This thesis has demonstrated both the potential and the limitations of data-driven approaches to causal inference, hopefully bringing clarity and guidance to applied researchers for how to interact with these new developments.

References

Agrawal, J. and Kamakura, W. A. (1999). Country of origin: A competitive advantage? *International Journal of Research in Marketing*, 16(4), 255–267.

Alley, M., Biggs, M., Hariss, R., Herrmann, C., Li, M. L., and Perakis, G. (2022). Pricing for Heterogeneous Products: Analytics for Ticket Reselling. *Manufacturing & Service Operations Management*, 25(2), 409–426.

Anderson, C. (2008). The End of Theory: The Data Deluge Makes the Scientific Method Obsolete. *Wired*. Retrieved August 20, 2024, from https://www.wired.com/2008/06/pb-theory/.

Angrist, J. D. and Pischke, J.-S. (2009). *Mostly Harmless Econometrics: An Empiricist's Companion*. Princeton: Princeton University Press, 1st edition.

Athey, S. (2017). Beyond prediction: Using big data for policy problems. *Science*, 355(6324), 483–485.

Athey, S. (2019). The Impact of Machine Learning on Economics. In A. Agrawal, J. Gans, and A. Goldfarb (Eds.), *The Economics of Artificial Intelligence: An Agenda* (pp. 507–552). Chicago: University of Chicago Press.

Athey, S. and Imbens, G. W. (2017). The State of Applied Econometrics: Causality and Policy Evaluation. *The Journal of Economic Perspectives*, 31(2), 3–32.

Athey, S., Tibshirani, J., and Wager, S. (2019). Generalized random forests. *Annals of Statistics*, 47(2), 1148–1178.

Azoulay, P., Greenblatt, W. H., and Heggeness, M. L. (2021). Long-term effects from early exposure to research: Evidence from the NIH "Yellow Berets". *Research Policy*, 50(9), 104332.

Beach, E. F. (1949). The Use of Polynomials to Represent Cost Functions. *The Review of Economic Studies*, 16(3), 158–169.

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

Belloni, A., Chernozhukov, V., and Hansen, C. (2014). High-Dimensional Methods and Inference on Structural and Treatment Effects. *Journal of Economic Perspectives*, 28(2), 29–50.

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

Bijmolt, T. H. A., van Heerde, H. J., and Pieters, R. G. M. (2005). New Empirical Generalizations on the Determinants of Price Elasticity. *Journal of Marketing Research*, 42(2), 141–156.

Bilancini, E., Boncinelli, L., Di Paolo, R., Menicagli, D., Pizziol, V., Ricciardi, E., and Serti, F. (2022). Prosocial behavior in emergencies: Evidence from blood donors recruitment and retention during the COVID-19 pandemic. *Social Science & Medicine*, 314, 115438.

Björkegren, D. and Grissen, D. (2020). Behavior Revealed in Mobile Phone Usage Predicts Credit Repayment. *The World Bank Economic Review*, 34(3), 618–634.

Bodory, H., Huber, M., and Lafférs, L. (2022). Evaluating (weighted) dynamic treatment effects by double machine learning. *The Econometrics Journal*, 25(3), 628–648.

Burauel, P. F. (2023). Evaluating Instrument Validity using the Principle of Independent Mechanisms. *Journal of Machine Learning Research*, 24(176), 1–56.

Card, D. (1999). The Causal Effect of Education on Earnings. In O. C. Ashenfelter and D. Card (Eds.), *Handbook of Labor Economics*, volume 3 (pp. 1801–1863). Elsevier.

Cárdenas, D., Lattimore, F., Steinberg, D., and Reynolds, K. J. (2022). Youth well-being predicts later academic success. *Scientific Reports*, 12(1), Article 2134.

Chamberlain, G. (1982). Multivariate regression models for panel data. *Journal of Econometrics*, 18(1), 5–46.

Chamberlain, G. (1984). Chapter 22 Panel data. In *Handbook of Econometrics*, volume 2 (pp. 1247–1318). Elsevier.

Chan, Z. T. and Meunier, S. (2022). Behind the screen: Understanding national support for a foreign investment screening mechanism in the European Union. *The Review of International Organizations*, 17(3), 513–541.

Chang, N.-C. (2020). Double/debiased machine learning for difference-in-differences models. *The Econometrics Journal*, 23(2), 177–191.

Chang, T.-H., Guo, Z., and Malinsky, D. (2024). Post-selection inference for causal effects after causal discovery. arXiv:2405.06763 [stat].

Chen, J.-e., Huang, C.-H., and Tien, J.-J. (2021). Debiased/Double Machine Learning for Instrumental Variable Quantile Regressions. *Econometrics*, 9(2), 1–18.

Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., Chen, K., Mitchell, R., Cano, I., Zhou, T., Li, M., Xie, J., Lin, M., Geng, Y., Li, Y., and Yuan, J. (2023). xgboost: Extreme Gradient Boosting.

Cheng, D., Li, J., Liu, L., Liu, J., and Le, T. D. (2024). Data-Driven Causal Effect Estimation Based on Graphical Causal Modelling: A Survey. *ACM Computing Surveys*, 56(5), 127:1–127:37.

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

Chernozhukov, V., Cinelli, C., Newey, W., Sharma, A., and Syrgkanis, V. (2022a). Long Story Short: Omitted Variable Bias in Causal Machine Learning. National Bureau of Economic Research.

Chernozhukov, V., Hansen, C., Kallus, N., Spindler, M., and Syrgkanis, V. (2024). Applied Causal Inference Powered by ML and AI. arXiv:2403.02467 [cs, econ, stat].

Chernozhukov, V., Kasahara, H., and Schrimpf, P. (2021). Causal impact of masks, policies, behavior on early covid-19 pandemic in the U.S. *Journal of Econometrics*, 220(1), 23–62.

Chernozhukov, V., Newey, W. K., and Singh, R. (2022b). Automatic Debiased Machine Learning of Causal and Structural Effects. *Econometrica*, 90(3), 967–1027.

Chiang, H. D., Kato, K., Ma, Y., and Sasaki, Y. (2022). Multiway Cluster Robust Double/Debiased Machine Learning. *Journal of Business & Economic Statistics*, 40(3), 1046–1056.

Chickering, D. M. (2002). Optimal Structure Identification With Greedy Search. *Journal of Machine Learning Research*, 3(Nov), 507–554.

Cinelli, C., Forney, A., and Pearl, J. (2024). A Crash Course in Good and Bad Controls. *Sociological Methods & Research*, 53(3), 1071–1104.

Cinelli, C. and Hazlett, C. (2020). Making Sense of Sensitivity: Extending Omitted Variable Bias. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 82(1), 39–67.

Clarke, P. and Polselli, A. (2023). Double Machine Learning for Static Panel Models with Fixed Effects. arXiv:2312.08174 [cs, econ, stat].

Colombo, D. and Maathuis, M. H. (2014). Order-independent constraint-based causal structure learning. *The Journal of Machine Learning Research*, 15(1), 3741–3782.

Colombo, D., Maathuis, M. H., Kalisch, M., and Richardson, T. S. (2012). Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, 40(1), 294–321.

Cornfield, J., Haenszel, W., Hammond, E. C., Lilienfeld, A. M., Shimkin, M. B., and Wynder, E. L. (2009). Smoking and lung cancer: recent evidence and a discussion of some questions. *International Journal of Epidemiology*, 38(5), 1175–1191.

Cunningham, S. (2021). *Causal Inference: The Mixtape*. Yale University Press.

Dang, L. E., Gruber, S., Lee, H., Dahabreh, I. J., Stuart, E. A., Williamson, B. D., Wyss, R., Díaz, I., Ghosh, D., Kıcıman, E., Alemayehu, D., Hoffman, K. L., Vossen, C. Y., Huml, R. A., Ravn, H., Kvist, K., Pratley, R., Shih, M.-C., Pennello, G., ..., and Petersen, M. (2023). A causal roadmap for generating high-quality real-world evidence. *Journal of Clinical and Translational Science*, 7(1), 1–12.

Daniel, F., Corporation, M., Weston, S., and Tenenbaum, D. (2022a). doParallel: Foreach Parallel Adaptor for the 'parallel' Package.

Daniel, F., Ooi, H., Calaway, R., Microsoft, and Weston, S. (2022b). foreach: Provides Foreach Looping Construct.

Dawid, A. P. (2010). Beware of the DAG! In *Proceedings of Workshop on Causality: Objectives and Assessment at NIPS 2008*, volume 6 (pp. 59–86).: PMLR.

Decarolis, F. and Giorgiantonio, C. (2022). Corruption red flags in public procurement: new evidence from Italian calls for tenders. *EPJ Data Science*, 11(1), 1–38.

Diaz, I. (2020). Machine learning in the estimation of causal effects: targeted minimum loss-based estimation and double/debiased machine learning. *Biostatistics*, 21(2), 353–358.

Dickson, M. C., Nguyen, M. M., Patel, C., Grabich, S. C., Benson, C., Cothran, T., and Skrepnek, G. H. (2023). Adherence, Persistence, Readmissions, and Costs in Medicaid Members with Schizophrenia or Schizoaffective Disorder Initiating Paliperidone Palmitate Versus Switching Oral Antipsychotics: A Real-World Retrospective Investigation. *Advances in Therapy*, 40(1), 349–366.

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

Eberhardt, F. (2017). Introduction to the foundations of causal discovery. *International Journal of Data Science and Analytics*, 3(2), 81–91.

Ehrmann, C., Bickenbach, J., and Stucki, G. (2019). Graphical modelling: a tool for describing and understanding the functioning of people living with a health condition. *European Journal of Physical and Rehabilitation Medicine*, 55(1), 131–135.

Ellickson, P. B., Kar, W., and Reeder, J. C. (2022). Estimating Marketing Component Effects: Double Machine Learning from Targeted Digital Promotions. *Marketing Science*, 42(4), 704–728.

Fang, Z. and He, Y. (2020). IDA with Background Knowledge. In *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 124 (pp. 270–279).: PMLR.

Farbmacher, H., Huber, M., Laffers, L., Langen, H., and Spindler, M. (2022). Causal mediation analysis with double machine learning. *The Econometrics Journal*, 25(2), 277–300.

Farrell, M. H. (2015). Robust inference on average treatment effects with possibly more covariates than observations. *Journal of Econometrics*, 189(1), 1–23.

Felderer, B., Kueck, J., and Spindler, M. (2023). Using Double Machine Learning to Understand Nonresponse in the Recruitment of a Mixed-Mode Online Panel. *Social Science Computer Review*, 41(2), 461–481.

Feuerriegel, S., Frauen, D., Melnychuk, V., Schweisthal, J., Hess, K., Curth, A., Bauer, S., Kilbertus, N., Kohane, I. S., and van der Schaar, M. (2024). Causal machine learning for predicting treatment outcomes. *Nature Medicine*, 30(4), 958–968.

Frot, B., Nandy, P., and Maathuis, M. H. (2019). Robust Causal Structure Learning with Some Hidden Variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 81(3), 459–487.

Gaujoux, R. (2023). doRNG: Generic Reproducible Parallel Backend for 'foreach' Loops.

Gelman, A. (2011). Causality and Statistical Learning. *American Journal of Sociology*, 117(3), 955–966.

Glymour, C., Zhang, K., and Spirtes, P. (2019). Review of Causal Discovery Methods Based on Graphical Models. *Frontiers in Genetics*, 10, Article 524.

Goller, D. (2023). Analysing a built-in advantage in asymmetric darts contests using causal machine learning. *Annals of Operations Research*, 325, 649–679.

Gordon, B. R., Moakler, R., and Zettelmeyer, F. (2022). Close Enough? A Large-Scale Exploration of Non-Experimental Approaches to Advertising Measurement. *Marketing Science*, 42(4), 768–793.

Grimmer, J., Knox, D., and Stewart, B. (2023). Naive regression requires weaker assumptions than factor models to adjust for multiple cause confounding. *Journal of Machine Learning Research*, 24(182), 1–70.

Hansen, D. (2020). The effectiveness of fiscal institutions: International financial flogging or domestic constraint? *European Journal of Political Economy*, 63, Article 101879.

Harrison, D. and Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5(1), 81–102.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media, 2nd edition.

Hauser, A. and Bühlmann, P. (2012). Characterization and greedy learning of interventional Markov equivalence classes of directed acyclic graphs. *The Journal of Machine Learning Research*, 13(1), 2409–2464.

Heinze-Deml, C., Maathuis, M. H., and Meinshausen, N. (2018a). Causal Structure Learning. *Annual Review of Statistics and Its Application*, 5(1), 371–391.

Heinze-Deml, C., Peters, J., and Meinshausen, N. (2018b). Invariant Causal Prediction for Nonlinear Models. *Journal of Causal Inference*, 6(2), Article 20170016.

Hernán, M. A. and Robins, J. M. (2020). *Causal Inference: What If*. Chapman & Hall/CRC.

Holtz, D., Zhao, M., Benzell, S. G., Cao, C. Y., Rahimian, M. A., Yang, J., Allen, J., Collis, A., Moehring, A., Sowrirajan, T., Ghosh, D., Zhang, Y., Dhillon, P. S., Nicolaides, C., Eckles, D., and Aral, S. (2020). Interdependence and the cost of uncoordinated responses to COVID-19. *Proceedings of the National Academy of Sciences*, 117(33), 19837–19843.

Hoyer, P., Janzing, D., Mooij, J. M., Peters, J., and Schölkopf, B. (2008). Nonlinear causal discovery with additive noise models. In *Advances in Neural Information Processing Systems*, volume 21: Curran Associates, Inc.

Huber, M., Meier, J., and Wallimann, H. (2022). Business analytics meets artificial intelligence: Assessing the demand effects of discounts on Swiss train tickets. *Transportation Research Part B: Methodological*, 163, 22–39.

Huntington-Klein, N. (2022). *The Effect: An Introduction to Research Design and Causality*. Chapman and Hall/CRC.

Hünermund, P., Louw, B., and Caspi, I. (2023). Double machine learning and automated confounder selection: A cautionary tale. *Journal of Causal Inference*, 11(1), Article 20220078.

Imbens, G. W. (2004). Nonparametric Estimation of Average Treatment Effects Under Exogeneity: A Review. *The Review of Economics and Statistics*, 86(1), 4–29.

Imbens, G. W. (2020). Potential Outcome and Directed Acyclic Graph Approaches to Causality: Relevance for Empirical Practice in Economics. *Journal of Economic Literature*, 58(4), 1129–1179.

Imbens, G. W. and Angrist, J. D. (1994). Identification and Estimation of Local Average Treatment Effects. *Econometrica*, 62(2), 467–475.

Imbens, G. W. and Rubin, D. B. (2015). *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. Cambridge: Cambridge University Press.

Imbens, G. W. and Wooldridge, J. M. (2009). Recent Developments in the Econometrics of Program Evaluation. *Journal of Economic Literature*, 47(1), 5–86.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2021). *An Introduction to Statistical Learning: with Applications in R*. New York: Springer, 2nd edition.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., ..., and Hassabis, D. (2021). Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873), 583–589.

Kalisch, M. and Bühlmann, P. (2014). Causal Structure Learning and Inference: A Selective Review. *Quality Technology & Quantitative Management*, 11(1), 3–21.

Kalisch, M., Fellinghauer, B. A., Grill, E., Maathuis, M. H., Mansmann, U., Bühlmann, P., and Stucki, G. (2010). Understanding human functioning using graphical models. *BMC Medical Research Methodology*, 10(1), 14.

Kalisch, M., Hauser, A., Maechler, M., Colombo, D., Entner, D., Hoyer, P., Hyttinen, A., Peters, J., Andri, N., Perkovic, E., Nandy, P., Ruetimann, P., Stekhoven, D., Schuerch, M., Eigenmann, M., Henckel, L., and Mooij, J. (2023). pcalg: Methods for Graphical Models and Causal Inference.

Kıcıman, E., Ness, R., Sharma, A., and Tan, C. (2024). Causal Reasoning and Large Language Models: Opening a New Frontier for Causality. arXiv:2305.00050 [cs, stat].

Kitchin, R. (2014). Big Data, new epistemologies and paradigm shifts. *Big Data & Society*, 1(1), 1–12.

Knaus, M. C. (2021). A double machine learning approach to estimate the effects of musical practice on student's skills. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 184(1), 282–300.

Knaus, M. C. (2022). Double machine learning-based programme evaluation under unconfoundedness. *The Econometrics Journal*, 25(3), 602–627.

Kuppelwieser, T. and Wozabal, D. (2021). Liquidity costs on intraday power markets: Continuous trading versus auctions. *Energy Policy*, 154, Article 112299.

Laan, M. J. v. d. and Rubin, D. (2006). Targeted Maximum Likelihood Learning. *The International Journal of Biostatistics*, 2(1), 1–38.

Leavitt, K., Schabram, K., Hariharan, P., and Barnes, C. M. (2021). Ghost in the Machine: On Organizational Theory in the Age of Machine Learning. *Academy of Management Review*, 46(4), 750–777.

Lee, J. J. R., Srinivasan, R., Ong, C. S., Alejo, D., Schena, S., Shpitser, I., Sussman, M., Whitman, G. J. R., and Malinsky, D. (2023). Causal determinants of postoperative length of stay in cardiac surgery using causal graphical learning. *The Journal of Thoracic and Cardiovascular Surgery*, 166(5), e446–e462.

Liaw, A. and Wiener, M. (2002). Classification and Regression by randomForest. *R News*, 2(3), 18–22.

Liu, M., Zhang, Y., and Zhou, D. (2021). Double/debiased machine learning for logistic partially linear model. *The Econometrics Journal*, 24(3), 559–588.

Loiseau, N., Trichelair, P., He, M., Andreux, M., Zaslavskiy, M., Wainrib, G., and Blum, M. G. B. (2022). External control arm analysis: an evaluation of propensity score approaches, G-computation, and doubly debiased machine learning. *BMC Medical Research Methodology*, 22(1), Article 335.

Lundberg, I. (2024). The Gap-Closing Estimand: A Causal Approach to Study Interventions That Close Disparities Across Social Categories. *Sociological Methods & Research*, 53(2), 507–570.

Maathuis, M. H. and Colombo, D. (2015). A generalized back-door criterion. *The Annals of Statistics*, 43(3), 1060–1088.

Maathuis, M. H., Colombo, D., Kalisch, M., and Bühlmann, P. (2010). Predicting causal effects in large-scale systems from observational data. *Nature Methods*, 7(4), 247–248.

Maathuis, M. H., Kalisch, M., and Bühlmann, P. (2009). Estimating high-dimensional intervention effects from observational data. *The Annals of Statistics*, 37(6A), 3133–3164.

Mahmoudi, S. M. and Wit, E. C. (2018). Estimating Causal Effects from Nonparanormal Observational Data. *The International Journal of Biostatistics*, 14(2), Article 20180030.

Malinsky, D. and Danks, D. (2018). Causal discovery algorithms: A practical guide. *Philosophy Compass*, 13(1), Article e12470.

Malinsky, D. and Spirtes, P. (2017). Estimating bounds on causal effects in high-dimensional and possibly confounded systems. *International Journal of Approximate Reasoning*, 88, 371–384.

Marra, G. and Wood, S. N. (2011). Practical variable selection for generalized additive models. *Computational Statistics & Data Analysis*, 55(7), 2372–2387.

Mayer-Schönberger, V. and Cukier, K. (2013). *Big Data: A Revolution that Will Transform how We Live, Work, and Think*. Houghton Mifflin Harcourt.

Mazzocchi, F. (2015). Could Big Data be the end of theory in science? *EMBO Reports*, 16(10), 1250–1255.

McAfee, A. and Brynjolfsson, E. (2012). Big Data: The Management Revolution. *Harvard Business Review*, 90(10), 60–68.

McConnell, K. J. and Lindner, S. (2019). Estimating treatment effects with machine learning. *Health Services Research*, 54(6), 1273–1282.

Mullainathan, S. and Spiess, J. (2017). Machine Learning: An Applied Econometric Approach. *Journal of Economic Perspectives*, 31(2), 87–106.

Mundlak, Y. (1978). On the Pooling of Time Series and Cross Section Data. *Econometrica*, 46(1), 69–85.

Naimi, A. I. and Westreich, D. J. (2014). Big Data: A Revolution That Will Transform How We Live, Work, and Think. *American Journal of Epidemiology*, 179(9), 1143–1144.

Nandy, P., Maathuis, M. H., and Richardson, T. S. (2017). Estimating the effect of joint interventions from observational data in sparse high-dimensional settings. *The Annals of Statistics*, 45(2), 647–674.

Nelson, J. P. (1978). Residential choice, hedonic prices, and the demand for urban air quality. *Journal of Urban Economics*, 5(3), 357–369.

Ogarrio, J. M., Spirtes, P., and Ramsey, J. (2016). A Hybrid Causal Search Algorithm for Latent Variable Models. In *Proceedings of the Eighth International Conference on Probabilistic Graphical Models*, volume 52 (pp. 368–379).: PMLR.

Ogburn, E. L., Shpitser, I., and Tchetgen, E. J. T. (2019). Comment on "Blessings of Multiple Causes". *Journal of the American Statistical Association*, 114(528), 1611–1615.

OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., ..., and Zoph, B. (2024). GPT-4 Technical Report. arXiv:2303.08774 [cs].

Papies, D., Ebbes, P., and Van Heerde, H. J. (2017). Addressing Endogeneity in Marketing Models. In P. S. H. Leeflang, J. E. Wieringa, T. H. Bijmolt, and K. H. Pauwels (Eds.), *Advanced Methods for Modeling Markets* (pp. 581–627). Cham: Springer International Publishing.

Parpouchi, M., Moniruzzaman, A., and Somers, J. M. (2021). The association between experiencing homelessness in childhood or youth and adult housing stability in Housing First. *BMC Psychiatry*, 21(1), Article 138.

Pearl, J. (2009). *Causality*. Cambridge: Cambridge University Press, 2nd edition.

Pearl, J., Glymour, M., and Jewell, N. P. (2016). *Causal Inference in Statistics: A Primer*. Chichester, West Sussex: Wiley, 1st edition.

Pearl, J. and Mackenzie, D. (2019). *The Book of Why: The New Science of Cause and Effect*. London: Penguin, 1st edition.

Perkovic, E., Kalisch, M., and Maathuis, M. (2017). Interpreting and Using CPDAGs With Background Knowledge. In *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence*.

Peters, J., Janzing, D., and Schölkopf, B. (2017). *Elements of Causal Inference: Foundations and Learning Algorithms*. Adaptive Computation and Machine Learning series. Cambridge, MA, USA: MIT Press.

Poterba, J. M., Venti, S. F., and Wise, D. A. (1994). 401(k) Plans and Tax-Deferred Saving. In D. A. Wise (Ed.), *Studies in the Economics of Aging* (pp. 105–142). Chicago, IL: University of Chicago Press.

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

Powell, J. L. (1994). Estimation of semiparametric models. In R. F. Engle and D. L. McFadden (Eds.), *Handbook of Econometrics*, volume 4 (pp. 2443–2521). Elsevier.

Qiu, M., Zigler, C., and Selin, N. E. (2022). Statistical and machine learning methods for evaluating trends in air quality under changing meteorological conditions. *Atmospheric Chemistry and Physics*, 22(16), 10551–10566.

R Core Team (2023). R: A Language and Environment for Statistical Computing.

Robins, J. M. and Rotnitzky, A. (1995). Semiparametric Efficiency in Multivariate Regression Models with Missing Data. *Journal of the American Statistical Association*, 90(429), 122–129.

Robins, J. M., Rotnitzky, A., and Zhao, L. P. (1994). Estimation of Regression Coefficients When Some Regressors Are Not Always Observed. *Journal of the American Statistical Association*, 89(427), 846–866.

Robinson, P. M. (1988). Root-N-Consistent Semiparametric Regression. *Econometrica*, 56(4), 931–954.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (pp. 234–241). Cham: Springer International Publishing.

Rosen, S. (1974). Hedonic Prices and Implicit Markets: Product Differentiation in Pure Competition. *Journal of Political Economy*, 82(1), 34–55.

Rosenbaum, P. R. and Rubin, D. B. (1983). The Central Role of the Propensity Score in Observational Studies for Causal Effects. *Biometrika*, 70(1), 41–55.

Rossi, P. E. (2014). Invited Paper: Even the Rich Can Make Themselves Poor: A Critical Examination of IV Methods in Marketing Applications. *Marketing Science*, 33(5), 655–672.

Rothenhäusler, D., Heinze, C., Peters, J., and Meinshausen, N. (2015). BACKSHIFT: Learning causal cyclic graphs from unknown shift interventions. In *Advances in Neural Information Processing Systems*, volume 28 (pp. 1513–1521).

Schölkopf, B. (2022). Causality for Machine Learning. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, volume 36 (pp. 765–804). New York, NY: Association for Computing Machinery, 1st edition.

Semenova, V. and Chernozhukov, V. (2021). Debiased machine learning of conditional average treatment effects and other causal functions. *The Econometrics Journal*, 24(2), 264–289.

Semenova, V., Goldman, M., Chernozhukov, V., and Taddy, M. (2023). Inference on heterogeneous treatment effects in high-dimensional dynamic panels under weak dependence. *Quantitative Economics*, 14(2), 471–510.

Shah, R. D. and Peters, J. (2020). The hardness of conditional independence testing and the generalised covariance measure. *The Annals of Statistics*, 48(3), 1514–1538.

Shalizi, C. (2021). Commentary on Kun Zhang's presentation "Learning and Using Causal Representations" [Online seminar discussion]. https://www.youtube.com/watch?v=_MVi6XzOdD0 [Recording, retrieved July 22, 2024]; https://drive.google.com/file/d/16kiv2IMiqHqaPt58NzFvNNICw5qFGCHp/view [Slides, retrieved July 22, 2024].

Shimizu, S., Hoyer, P. O., Hyvärinen, A., and Kerminen, A. (2006). A Linear Non-Gaussian Acyclic Model for Causal Discovery. *Journal of Machine Learning Research*, 7(72), 2003–2030.

Shmueli, G. (2010). To Explain or to Predict? *Statistical Science*, 25(3), 289–310.

Simon, N., Friedman, J. H., Hastie, T., and Tibshirani, R. (2011). Regularization Paths for Cox's Proportional Hazards Model via Coordinate Descent. *Journal of Statistical Software*, 39(5), 1–13.

Skoufias, E. and Vinha, K. (2021). Child stature, maternal education, and early childhood development in Nigeria. *PLOS ONE*, 16(12), 1–17.

Spirtes, P., Glymour, C., and Scheines, R. (2001). *Causation, Prediction, and Search*. The MIT Press.

Spirtes, P. and Zhang, K. (2016). Causal discovery and inference: concepts and recent methodological advances. *Applied Informatics*, 3(1), Article 3.

Squires, C. and Uhler, C. (2023). Causal Structure Learning: A Combinatorial Perspective. *Foundations of Computational Mathematics*, 23(5), 1781–1815.

StataCorp (2019). Stata Statistical Software: Release 16.

Stekhoven, D. J., Moraes, I., Sveinbjörnsson, G., Hennig, L., Maathuis, M. H., and Bühlmann, P. (2012). Causal stability ranking. *Bioinformatics*, 28(21), 2819–2823.

Taruttis, F., Spang, R., and Engelmann, J. C. (2015). A statistical approach to virtual cellular experiments: improved causal discovery using accumulation IDA (aIDA). *Bioinformatics*, 31(23), 3807–3814.

Teramoto, R., Saito, C., and Funahashi, S.-i. (2014). Estimating causal effects with a non-paranormal method for the design of efficient intervention experiments. *BMC Bioinformatics*, 15(1), Article 228.

Tsamardinos, I., Brown, L. E., and Aliferis, C. F. (2006). The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65(1), 31–78.

Uhler, C., Raskutti, G., Bühlmann, P., and Yu, B. (2013). Geometry of the faithfulness assumption in causal inference. *The Annals of Statistics*, 41(2), 436–463.

Vansteelandt, S., Dukes, O., Van Lancker, K., and Martinussen, T. (2022). Assumption-Lean Cox Regression. *Journal of the American Statistical Association*, 119(545), 475–484.

Varian, H. R. (2014). Big Data: New Tricks for Econometrics. *Journal of Economic Perspectives*, 28(2), 3–28.

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. New York: Springer, 4th edition.

Vowels, M. J., Camgoz, N. C., and Bowden, R. (2022). D'ya Like DAGs? A Survey on Structure Learning and Causal Discovery. *ACM Computing Surveys*, 55(4), 82:1–82:36.

Wager, S. and Athey, S. (2018). Estimation and Inference of Heterogeneous Treatment Effects using Random Forests. *Journal of the American Statistical Association*, 113(523), 1228–1242.

Wang, Y. and Blei, D. M. (2019). The Blessings of Multiple Causes. *Journal of the American Statistical Association*, 114(528), 1574–1596.

Witte, J., Henckel, L., Maathuis, M. H., and Didelez, V. (2020). On efficient adjustment in causal graphs. *The Journal of Machine Learning Research*, 21(1), Article 246.

Wood, S. N. (2017). *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, 2nd edition.

Wooldridge, J. (2012). *Introductory Econometrics: A Modern Approach*. Mason, OH: Cengage Learning, Inc, 5th edition.

Wooldridge, J. M. (2010). *Econometric Analysis of Cross Section and Panel Data*. Cambridge, MA, USA: MIT Press, 2nd edition.

Yamane, T. and Kaneko, S. (2021). Is the younger generation a driving force toward achieving the sustainable development goals? Survey experiments. *Journal of Cleaner Production*, 292, Article 125932.

Yang, J.-C., Chuang, H.-C., and Kuan, C.-M. (2020). Double machine learning with gradient boosting and its application to the Big N audit quality effect. *Journal of Econometrics*, 216(1), 268–283.

Zhang, J. (2008). Causal Reasoning with Ancestral Graphs. *Journal of Machine Learning Research*, 9(47), 1437–1474.

Zhang, K. and Hyvärinen, A. (2009). On the identifiability of the post-nonlinear causal model. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09 (pp. 647–655). Arlington, VA: AUAI Press.

Zivich, P. N. and Breskin, A. (2021). Machine Learning for Causal Inference: On the Use of Cross-fit Estimators. *Epidemiology*, 32(3), 393–401.

Appendix A

Appendix Chapter 2

A.1 Literature selection

For the literature review, we identified 46 papers published up to March 2023, which cite the initial proposal of DML in Chernozhukov et al. (2018), and apply the method to real-world data. From these, we excluded papers that contained too little information about the implementation of DML or the data used, either because it was not reported or because we could not access the full paper. At a minimum, the authors had to report which ML algorithm they used and their sample size. We also excluded papers that use the double selection method by Belloni et al. (2014), but call it DML, since it does not fit the definition of DML in Chernozhukov et al. (2018). If the study contained multiple different specifications, we recorded the lowest sample size n and the largest number of raw covariates p to get a conservative estimate of the dimensionality. If the authors did not state number of covariates explicitly, we counted all covariates mentioned in the text. Typically, the authors transformed categorical covariates into dummy variables. We counted p after that transformation; in cases where the number of levels of a categorical variable is unclear, we guessed it from the available information. The covariate dimension p always refers to the raw covariates, before (some of) the authors added nonlinearities through polynomials, interactions or other transformations.

Table A.1: Published papers applying DML

Year	Author	Title	Journal	Discipline
2018	Chernozhukov et al.	Double/debiased machine learning for treatment and structural parameters	The Econometrics Journal	Statistics / Econometrics
2019	McConnell and Lindner	Estimating treatment effects with machine learning	Health Services Research	Healthcare / Medicine
2020	Chang	Double/debiased machine learning for difference-in-differences models	The Econometrics Journal	Statistics / Econometrics
2020	Dube et al.	Monopsony in Online Labor Markets	American Economic Review: Insights	Economics
2020	Hansen	The effectiveness of fiscal institutions: International financial flogging or domestic constraint?	European Journal of Political Economy	Political Science
2020	Holtz et al.	Interdependence and the cost of uncoordinated responses to COVID-19	Proceedings of the National Academy of Sciences	Economics
2020	Yang et al.	Double machine learning with gradient boosting and its application to the Big N audit quality effect	Journal of Econometrics	Economics
2021	Azoulay et al.	Long-term effects from early exposure to research: Evidence from the NIH "Yellow Berets"	Research Policy	Healthcare / Medicine
2021	Chan and Meunier	Behind the screen: Understanding national support for a foreign investment screening mechanism in the European Union	The Review of International Organizations	Political Science
2021	Chen et al.	Debiased/Double Machine Learning for Instrumental Variable Quantile Regressions	Econometrics	Statistics / Econometrics
2021	Chernozhukov et al.	Causal impact of masks, policies, behavior on early covid-19 pandemic in the U.S.	Journal of Econometrics	Economics
2021	Knaus	A double machine learning approach to estimate the effects of musical practice on student's skills	Journal of the Royal Statistical Society: Series A (Statistics in Society)	Economics
2021	Kuppelwieser and Wozabal	Liquidity costs on intraday power markets: Continuous trading versus auctions	Energy Policy	Economics
2021	Liu et al.	Double/debiased machine learning for logistic partially linear model	The Econometrics Journal	Statistics / Econometrics
2021	Parpouchi et al.	The association between experiencing homelessness in childhood or youth and adult housing stability in Housing First	BMC Psychiatry	Healthcare / Medicine
2021	Semenova and Chernozhukov	Debiased machine learning of conditional average treatment effects and other causal functions	The Econometrics Journal	Statistics / Econometrics
2021	Skoufias and Vinha	Child stature, maternal education, and early childhood development in Nigeria	PLOS ONE	Interdisciplinary
2021	Yamane and Kaneko	Is the younger generation a driving force toward achieving the sustainable development goals? Survey experiments	Journal of Cleaner Production	Sociology
2022	Alley et al.	Pricing for Heterogeneous Products: Analytics for Ticket Reselling	Manufacturing & Service Operations Management	Economics
2022	Bilancini et al.	Prosocial behavior in emergencies: Evidence from blood donors recruitment and retention during the COVID-19 pandemic	Social Science & Medicine	Interdisciplinary
2022	Bodory et al.	Evaluating (weighted) dynamic treatment effects by double machine learning	The Econometrics Journal	Statistics / Econometrics
2022	Cárdenas et al.	Youth well-being predicts later academic success	Scientific Reports	Interdisciplinary
2022	Chiang et al.	Multiway Cluster Robust Double/Debiased Machine Learning	Journal of Business & Economic Statistics	Statistics / Econometrics
2022	Decarolis and Giorgiantonio	Corruption red flags in public procurement: New evidence from Italian calls for tenders	EPJ Data Science	Economics
2022	Ellickson et al.	Estimating Marketing Component Effects: Double Machine Learning from Targeted Digital Promotions	Marketing Science	Economics
2022	Farbmacher et al.	Causal mediation analysis with double machine learning	The Econometrics Journal	Statistics / Econometrics
2022	Goller	Analyzing a built-in advantage in asymmetric darts contests using causal machine learning	Annals of Operations Research	Sports Research
2022	Gordon et al.	Close Enough? A Large-Scale Exploration of Non-Experimental Approaches to Advertising Measurement	Marketing Science	Economics
2022	Huber et al.	Business analytics meets artificial intelligence: Assessing the demand effects of discounts on Swiss train tickets	Transportation Research Part B: Methodological	Economics
2022	Knaus	Double machine learning-based programme evaluation under unconfoundedness	The Econometrics Journal	Statistics / Econometrics
2022	Loiseau et al.	External control arm analysis: An evaluation of propensity score approaches, G-computation, and doubly debiased machine learning	BMC Medical Research Methodology	Healthcare / Medicine
2022	Lundberg	The Gap-Closing Estimand: A Causal Approach to Study Interventions That Close Disparities Across Social Categories	Sociological Methods & Research	Sociology
2022	Qiu et al.	Statistical and machine learning methods for evaluating trends in air quality under changing meteorological conditions	Atmospheric Chemistry and Physics	Geoscience
2022	Vansteelandt et al.	Assumption-Lean Cox Regression	Journal of the American Statistical Association	Statistics / Econometrics
2023	Dickson et al.	Adherence, Persistence, Readmissions, and Costs in Medicaid Members with Schizophrenia or Schizoaffective Disorder Initiating Paliperidone Palmitate Versus Switching Oral Antipsychotics: A Real-World Retrospective Investigation	Advances in Therapy	Healthcare / Medicine
2023	Felderer et al.	Using Double Machine Learning to Understand Nonresponse in the Recruitment of a Mixed-Mode Online Panel	Social Science Computer Review	Sociology

A.2 Notes for replication

In this appendix section, we note additional settings in our code not mentioned in the main text that should ensure replicability of our results.

For all computations, we use the random seed 42 and compute in parallel on 50 cores on the HPC bwUniCluster. For parallel computing, we use the package *foreach* (Daniel et al., 2022b) in combination with the *doParallel* backend (Daniel et al., 2022a) and rely on the package *doRNG* (Gaujoux, 2023) for replicable random number generation on multiple cores.

A.3 Figures

Comparing baseline results for different numbers of simulation iterations

Figure A.1 shows results for the "baseline" simulation setting for 50 to 1,000 simulation iterations. The overall results are very stable. As expected, adding more iterations creates more outliers, especially for the methods not adjusting well for confounding. Based on these results, we choose 100 simulation iterations for the remainder of our simulations, as the resulting distribution seems fairly representative, while using computing resources efficiently.

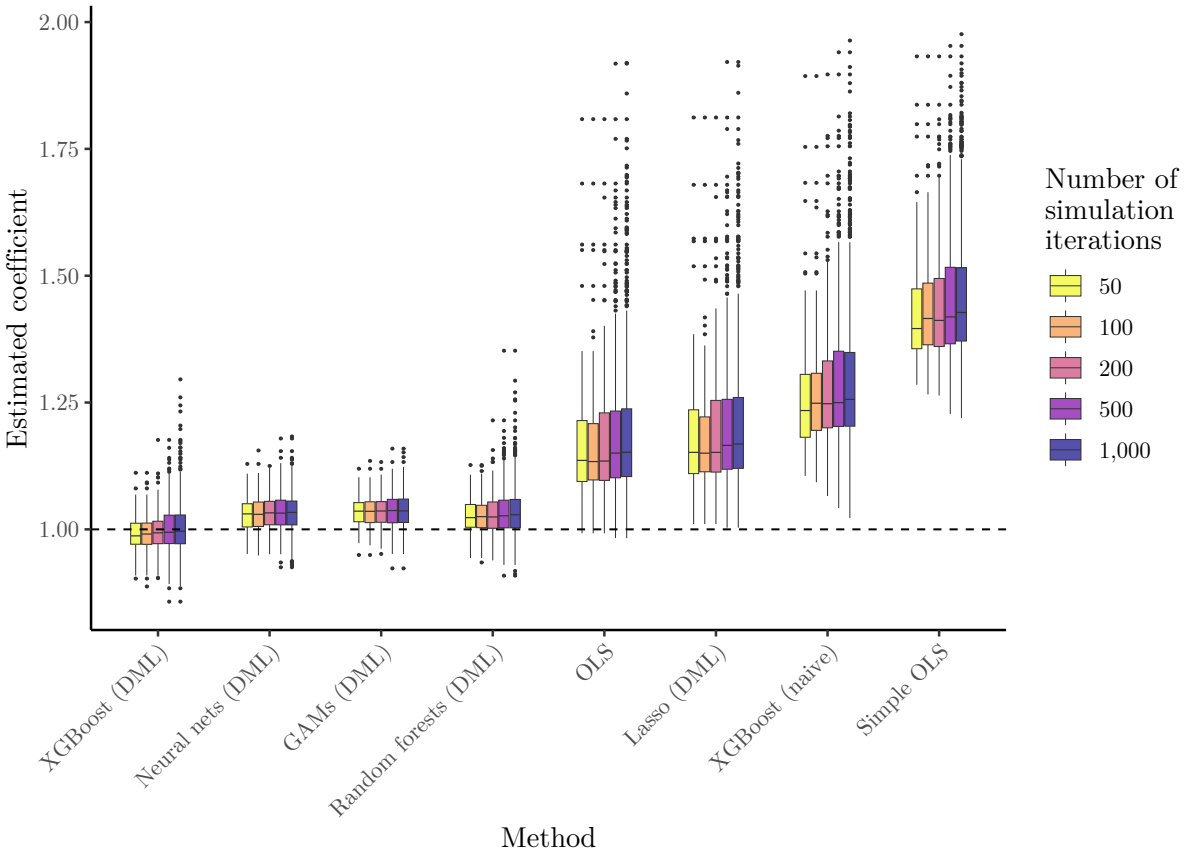


Figure A.1: Comparing baseline results for different numbers of simulation iterations

A.4 Application notes

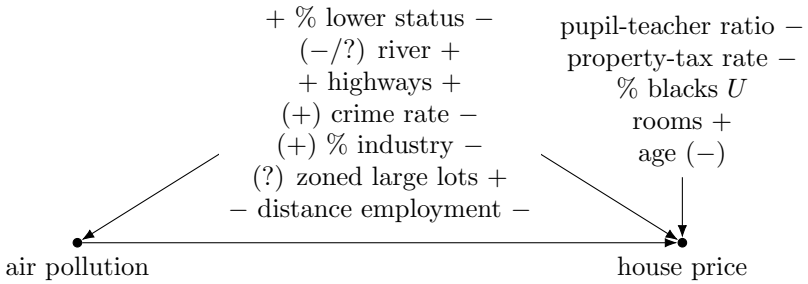


Figure A.2: Causal structure argued for in Harrison and Rubinfeld (1978), including hypothesized effect signs. +/− are explicitly stated, (+)/(−) are implicitly inferred. *U* refers to a U-shaped parabolic relationship.

Table A.2: Variable descriptions (replication of Table IV in Harrison and Rubinfeld (1978))

Variable	Definition	Source
MV	Median value of owner-occupied homes.	1970 U. S. Census
RM	Average number of rooms in owner units. RM represents spaciousness and, in a certain sense, quantity of housing. It should be positively related to housing value. The RM^2 form was found to provide a better fit than either the linear or logarithmic forms.	1970 U. S. Census
AGE	Proportion of owner units built prior to 1940. Unit age is generally related to structure quality.	1970 U. S. Census
B	Black proportion of population. At low to moderate levels of B, an increase in B should have a negative influence on housing value if Blacks are regarded as undesirable neighbors by Whites. However, market discrimination means that housing values are higher at very high levels of B. One expects, therefore, a parabolic relationship between proportion Black in a neighborhood and housing values.[1]	1970 U. S. Census
$LSTAT$	Proportion of population of that is lower status $= \frac{1}{2}$ (proportion of adults without some high school education and proportion of male workers classified as laborers). The logarithmic specification implies that socioeconomic status distinctions mean more in the upper brackets of society than in lower classes.	1970 U. S. Census
$CRIM$	Crime rate by town. Since $CRIM$ gauges the threat to well-being that households perceive in various neighborhoods of the Boston metropolitan area (assuming that crime rates are generally proportional to people's perceptions of danger) it should have a negative effect on housing values.	FBI (1970)
ZN	Proportion of a town's residential land zoned for lots greater than 25,000 square feet. Since such zoning restricts construction of small lot houses, we expect ZN to be positively related to housing values. A positive coefficient may also arise because zoning proxies the exclusivity, social class, and outdoor amenities of a community.	Metropolitan Area Planning Commission (1972)
$INDUS$	Proportion non-retail business acres per town. $INDUS$ serves as a proxy for the externalities associated with industry-noise, heavy traffic, and unpleasant visual effects, and thus should affect housing values negatively.	Vogt, Ivers and Associates [33]

[1] Comment by JF: This is a rather controversial variable (see, e.g., https://medium.com/@docintangible/racist-data-destruction-113e3eff54a8). With the authors' functional form, it is in a way modeling systemic racism. This is problematic if the model were to be used to set prices for homes based on this variable. However, since the original authors (and we) are interested in the effect of air pollution on housing prices, this variable is only included to adjust for how past racism might bias this effect. In this sense, it describes and adjusts for the discrimination *at that time*; we do not suggest to use it to predict prices, and its use is no endorsement of systemic racism. An additional problem is that the available data does not contain the "black" variable in its raw form, that is, the black proportion of the population. It only contains the parabolic transformation, which is a non-invertible function, so we cannot transform it back to the original raw values. One can argue against this functional form, but we expect using the variable with a wrong functional form (which the ML methods might even improve) is still more informative than not using it at all. Because of this, we use the transformed variable ("B_trans") throughout our analysis.

TAX	Full value property tax rate ($/$10,000). Measures the cost of public services in each community. Nominal tax rates were corrected by local assessment ratios to yield the full value tax rate for each town. Intratown differences in the assessment ratio were difficult to obtain and thus not used. The coefficient of this variable should be negative.	Massachusetts Taxpayers Foundation (1970)
$PTRATIO$	Pupil-Teacher ratio by town school district. Measures public sector benefits in each town. The relation of the pupil-teacher ratio to school quality is not entirely clear, although a low ratio should imply each student receives more individual attention. We expect the sign on $PTRATIO$ to be negative.	Massachusetts Dept. of Education (1971-1972)
$CHAS$	Charles River dummy: =1 if tract bounds the Charles River; =0 if otherwise. $CHAS$ captures the amenities of a riverside location and thus the coefficient should be positive.	1970 U. S. Census Tract maps
DIS	Weighted distances to five employment centers in the Boston region. According to traditional theories of urban land rent gradients, housing values should be higher near employment centers. DIS is entered in logarithm form; the expected sign is negative.	Schanre [29]
RAD	Index of accessibility to radial highways. The highway access index was calculated on a town basis. Good road access variables are needed so that auto pollution variables do not capture the locational advantages of roadways. RAD captures other sorts of locational advantages besides nearness to workplace. It is entered in logarithmic form; the expected sign is positive.	MIT Boston Project
NOX	Nitrogen oxide concentrations in pphm (annual average concentration in parts per hundred million).	TASSIM
$PART$	Particulate concentrations in mg/hcm^3 (annual average concentration in milligrams per hundred cubic meters)	TASSIM

Table A.3: Descriptive statistics of application

Variable	Minimum	Q1	Median	Mean	Q3	Maximum	SD
medv	5000	17025	21200	22532	25000	50000	9197
nox	3.85	4.49	5.38	5.54	6.24	8.71	1.15
crim	0.006	0.082	0.256	3.613	3.677	88.976	8.601
zn	0.00	0.00	0.00	11.36	12.50	100.00	23.32
indus	0.46	5.19	9.69	11.13	18.10	27.74	6.86
chas	0.00	0.00	0.00	0.07	0.00	1.00	0.25
rm	3.56	5.88	6.20	6.28	6.62	8.78	0.70
age	2.90	45.02	77.50	68.57	94.07	100.00	28.14
dis	1.12	2.10	3.20	3.79	5.18	12.12	2.10
rad	1.00	4.000	5.00	9.54	24.00	24.00	8.70
tax	187.0	279.00	330.0	408.2	666.0	711.0	168.5
ptratio	12.60	17.40	19.05	18.45	20.20	22.00	2.16
B_trans	0.0003	0.3754	0.3914	0.3567	0.3962	0.3969	0.0913
lstat	0.0173	0.0695	0.1136	0.1265	0.1696	0.3797	0.0714

Appendix B

Appendix Chapter 3

B.1 Cross-fitting techniques illustrations and results

B.1.1 Illustration of splitting procedures

Figure B.1: Illustration of cross-fitting when splitting by time into folds with adjacent periods. Data is split by time into K folds (here, $K = 10$). Each $\mathcal{M}$ is one fold of data. We train the two ML models on the folds within the golden box. We make predictions with these models and estimate the effects on the fold printed in bold within the gray box.

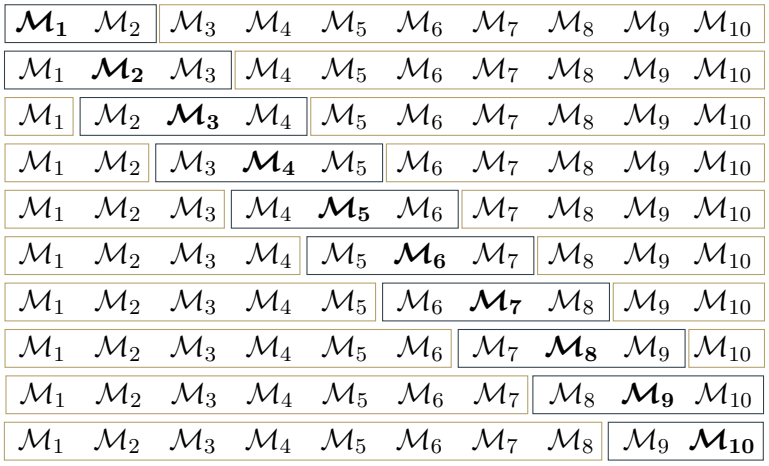


Figure B.2: Illustration of "neighbors-left-out cross-fitting" (Semenova et al., 2023). Data is split by time into K folds (here, $K = 10$). Each $\mathcal{M}$ is one fold of data. We train the two ML models on the folds within the golden box. We make predictions with these models and estimate the effects on the fold printed in bold. The other folds within the grey box are in the immediate neighborhood of the bold fold and excluded from both training and estimation.

B.1.2 Results for cross-fitting with different N and T

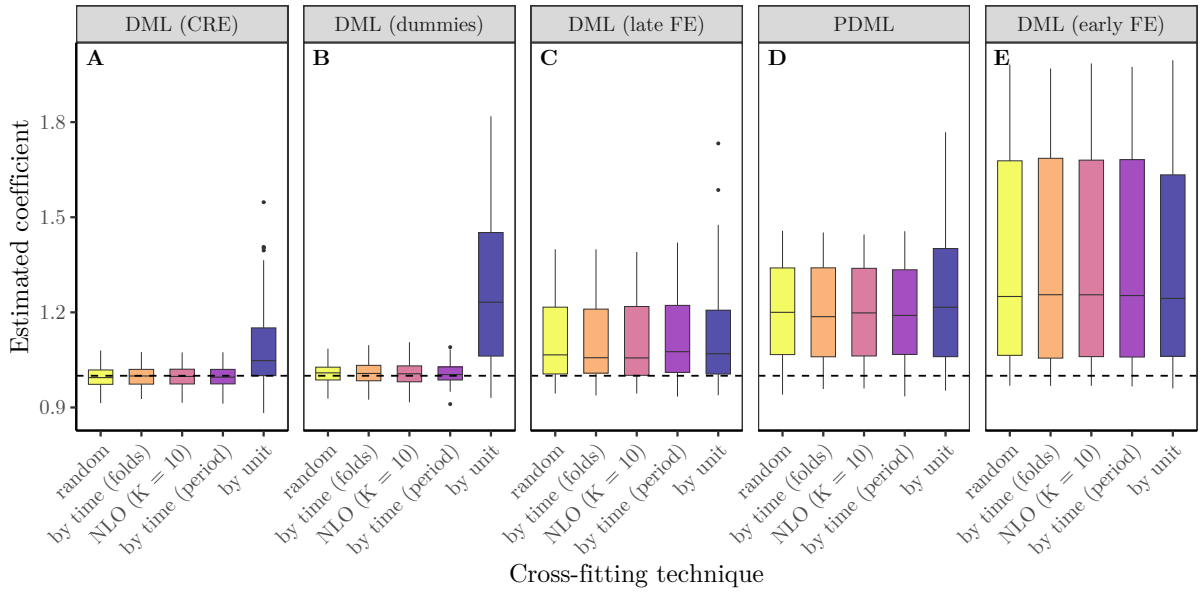


Figure B.3: Results for utilizing different cross-fitting techniques (Table 3.2) within various DML estimators for $\underline{N = 10}$ and $\underline{T = 500}$. The vertical axis depicts the estimated coefficient. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method. Data is generated according to DGP (C), with one observed confounder, u-shaped functional forms and a large degree of autocorrelation ($\rho = .9$). NLO: neighbors-left-out cross-fitting.

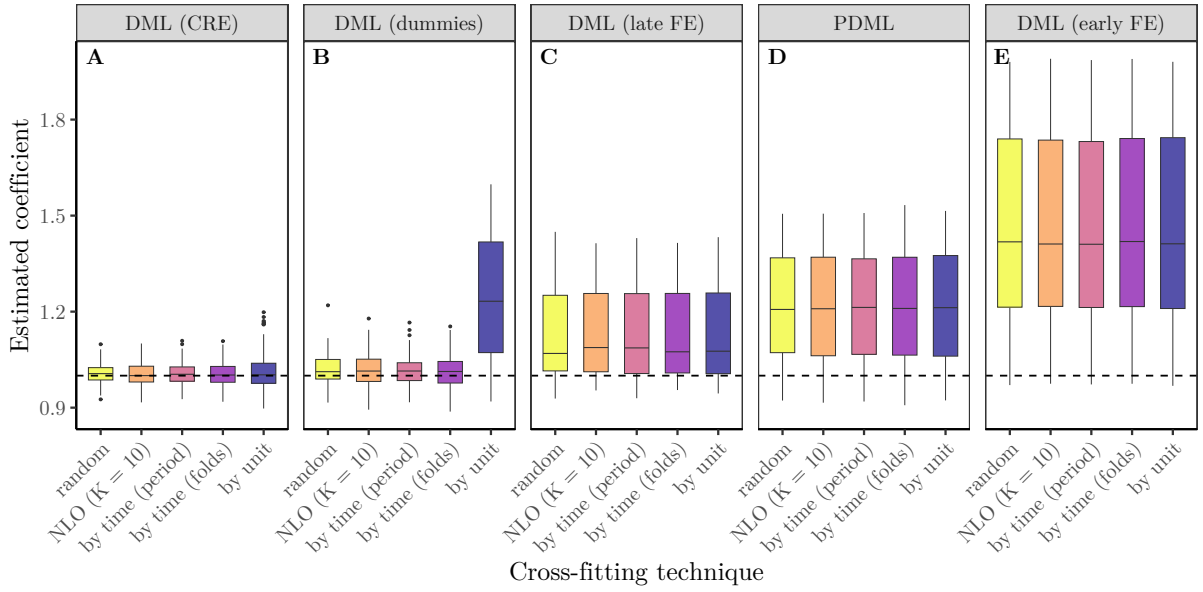


Figure B.4: Results for utilizing different cross-fitting techniques (Table 3.2) within various DML estimators for $\underline{N = 250}$ and $\underline{T = 20}$. The vertical axis depicts the estimated coefficient. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method. Data is generated according to DGP (C), with one observed confounder, u-shaped functional forms and a large degree of autocorrelation ($\rho = .9$). NLO: neighbors-left-out cross-fitting.

B.2 Computational efficiency of different approaches

In addition to their varying estimation performance, the different considered estimation approaches also strongly differ in their computational efficiency (Table B.1). Most notably, DML with dummies is computationally much more expensive than the alternatives, especially in settings where the number of units and thus dummy variables gets large. In our baseline setting, the estimation for a single dataset with $N = 500$ units and $T = 10$ periods, using 5-fold cross-fitting, is approximately 42.8 times slower compared to the second slowest method (DML with CRE). This effect is less pronounced in settings with fewer units (and thus fixed effects).

Table B.1: Computation times (in seconds) for different approaches and different data structures

Method	$N = 500$ / $T = 10$	$N = 100$ / $T = 50$	$N = 50$ / $T = 100$	$N = 10$ / $T = 500$
DML (early FE)	4.56	5.10	4.54	5.61
PDML	6.38	6.78	6.56	6.72
DML (late FE)	6.40	6.71	6.63	6.89
DML (CRE)	7.69	8.22	7.79	7.89
DML (dummies)	329.43	47.67	23.56	13.09

Note: N: number of units, T: number of periods. Reported execution times are the averages of five iterations for each method-dataset combination. We simulated the data according to DGP (C) of the baseline simulation setting. For this simple comparison, we computed each method on a standard laptop with an Intel Core i5-8365U 4-core CPU with 1.60 GHz and 16 GB of RAM.

B.3 Further settings and results

B.3.1 Results for intermediate numbers of N and T

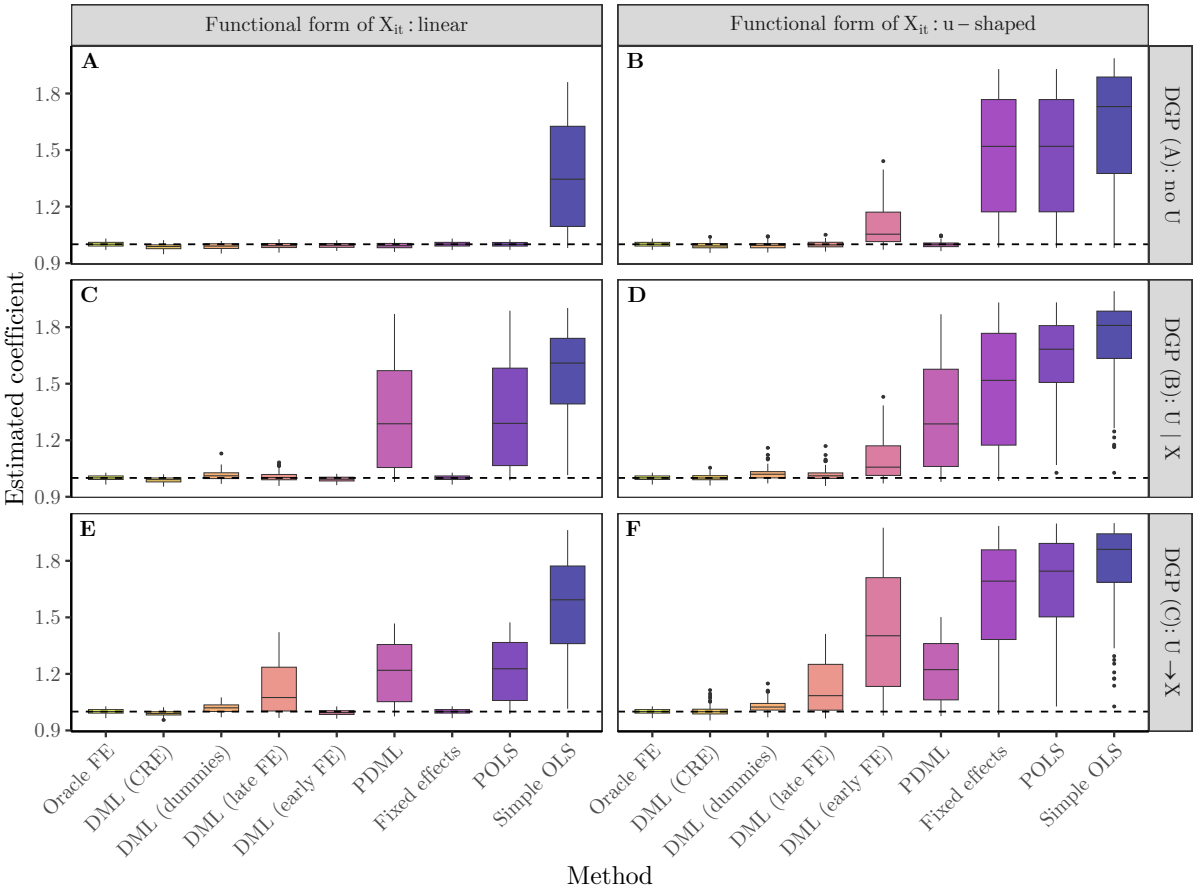


Figure B.5: Results for the setting with $\underline{N = 100}$ units and $\underline{T = 50}$ periods. The horizontal axis displays the different methods from Table 3.4. The vertical axis depicts the estimated coefficient. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method. The three rows contain three different DGPs: "$no\,U$" indicates no unobserved heterogeneity, "$U \mid X$" means the unobserved heterogeneity influences treatment and outcome, but not confounders, and "$U \rightarrow X$" means the unobserved heterogeneity also influences the confounders.

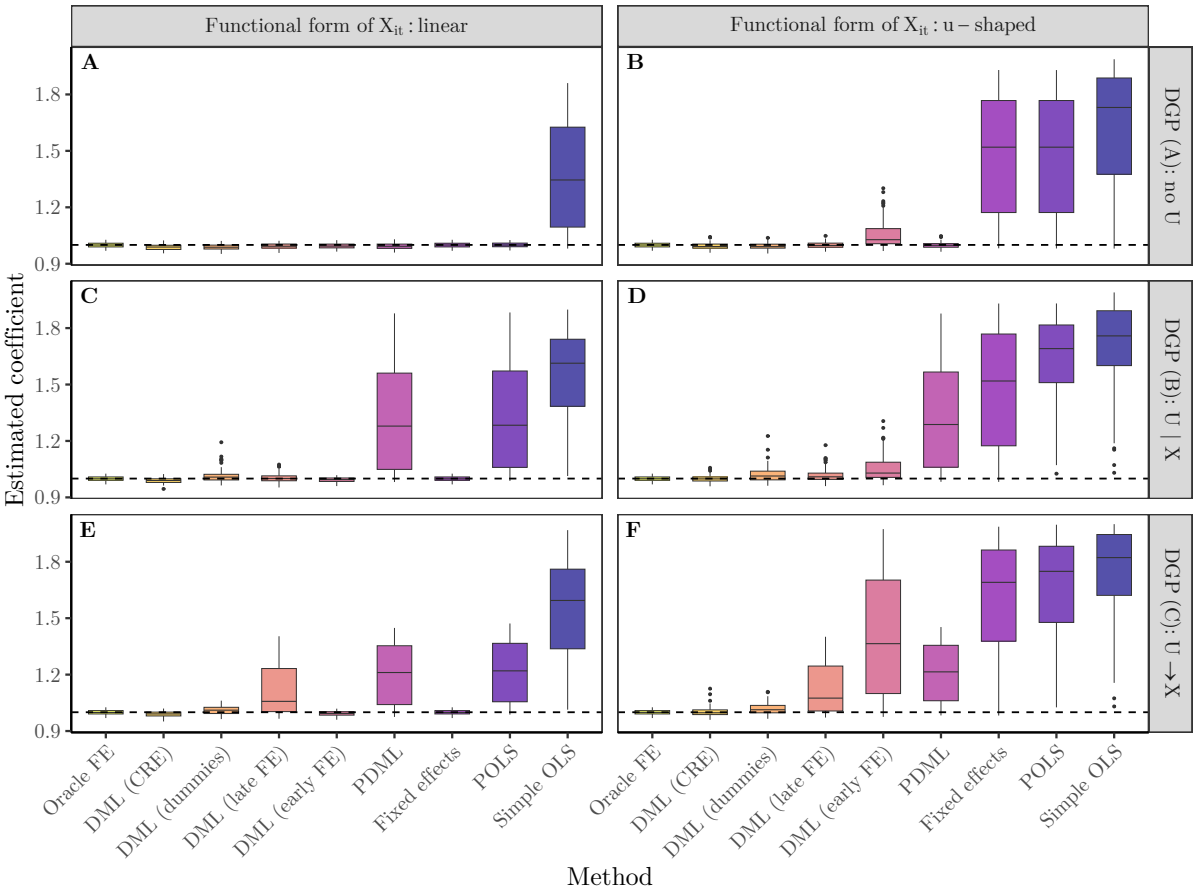


Figure B.6: Results for the setting with $\underline{N = 50}$ units and $\underline{T = 100}$ periods. The horizontal axis displays the different methods from Table 3.4. The vertical axis depicts the estimated coefficient. The dashed line marks the true causal effect ($\beta = 1$). The boxplots show the distribution of estimated coefficients across 100 simulated datasets for each method. The three rows contain three different DGPs: "$no\,U$" indicates no unobserved heterogeneity, "$U \mid X$" means the unobserved heterogeneity influences treatment and outcome, but not confounders, and "$U \rightarrow X$" means the unobserved heterogeneity also influences the confounders.

B.3.2 Varying the number of observed confounders in linear settings or with more periods

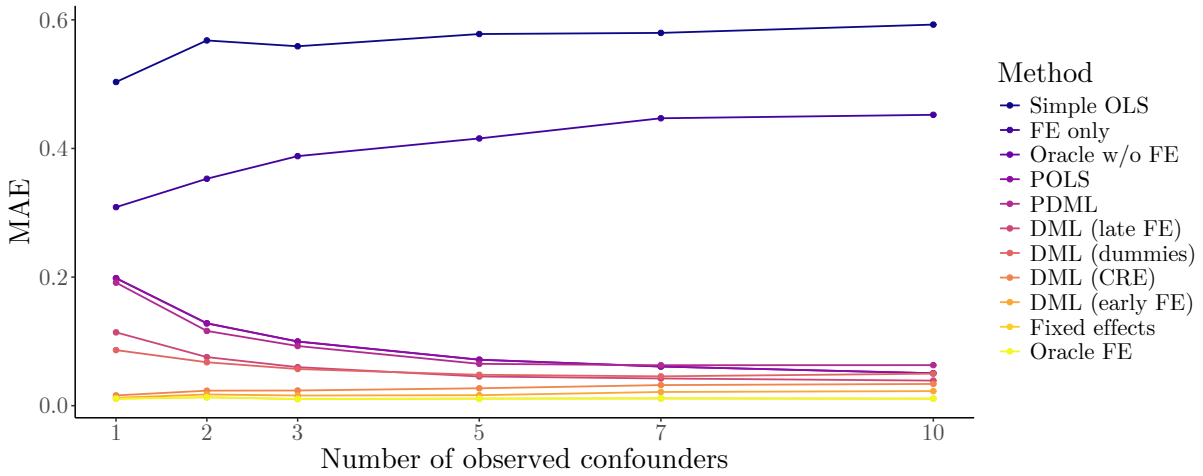


Figure B.7: Mean absolute error in the estimated coefficient across 100 simulations by number of observed confounders. The simulated confounding influence is linear, the causal structure is (C), i.e., $U_i \to X_{it}$. $N = 500$, $T = 10$.

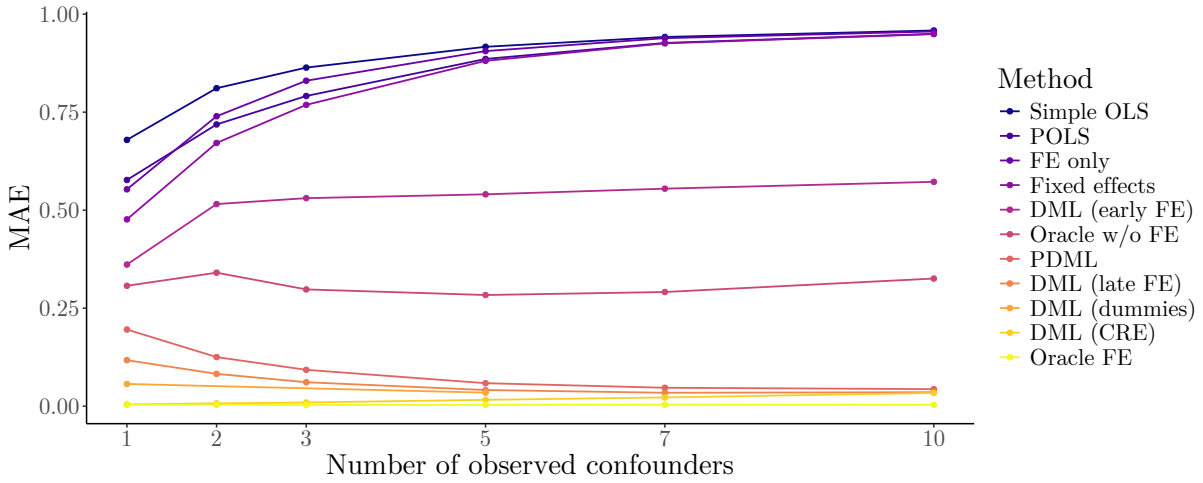


Figure B.8: Mean absolute error in the estimated coefficient across 100 simulations by number of observed confounders. The simulated confounding influence is u-shaped, the causal structure is (C), i.e., $U_i \to X_{it}$. $N = 500$, $T = 100$. We computed DML (dummies) only for 1 and 5 confounders due to unreasonably high computational costs for this data size.

B.3.3 Increasing sample size by increasing the number of periods

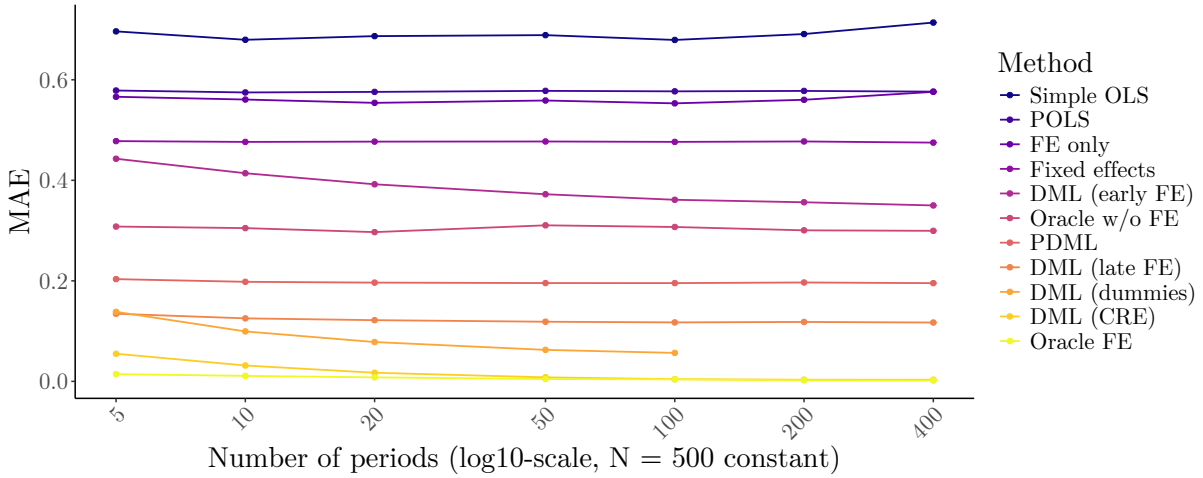


Figure B.9: Mean absolute error in the estimated coefficient across 100 simulations for **1** observed confounder by the number of periods. The number of units is fixed at $N = 500$. The simulated confounding influence is u-shaped, the causal structure is (C), i.e., $U_i \to X_{it}$. We computed DML (dummies) only for up to $T = 100$, as it becomes computationally too costly for larger values.

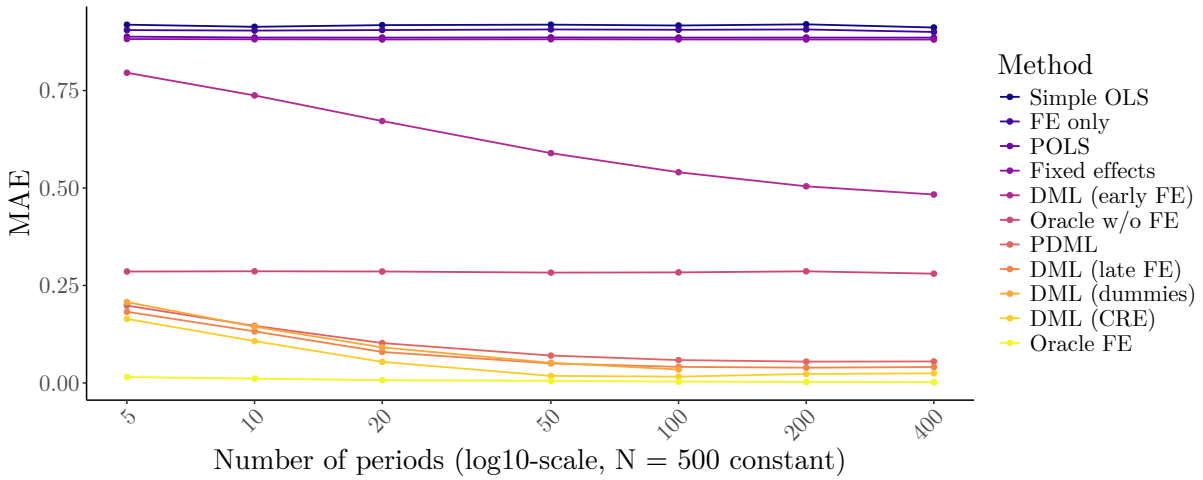


Figure B.10: Mean absolute error in the estimated coefficient across 100 simulations for **5** observed confounders by the number of periods. The number of units is fixed at $N = 500$. The simulated confounding influence is u-shaped, the causal structure is (C), i.e., $U_i \to X_{it}$. We computed DML (dummies) only for up to $T = 100$, as it becomes computationally too costly for larger values.

Appendix C

Appendix Chapter 4

C.1 Subtle violations of faithfulness

Figure C.1: How a subtle violation of faithfulness/specific parameter combinations can make the PC algorithm fail. **(A)** True graph, where all edge weights are 1 and the noise term of each variable follows a standard normal distribution. **(B)** The CPDAG estimated by the PC algorithm, which is missing the edge between W and Y, and is unable to direct any edges. This is just one special case. As soon as the edge weights take any value other than 1 (e.g., 1.1), the PC algorithm discovers the correct DAG. Also, even if all edge weights remain at 1, slightly increasing or decreasing the variance of the noise term of either W or Y (such that it is no longer exactly 1) also makes the graph discoverable.

C.2 Causal discovery under sufficiency

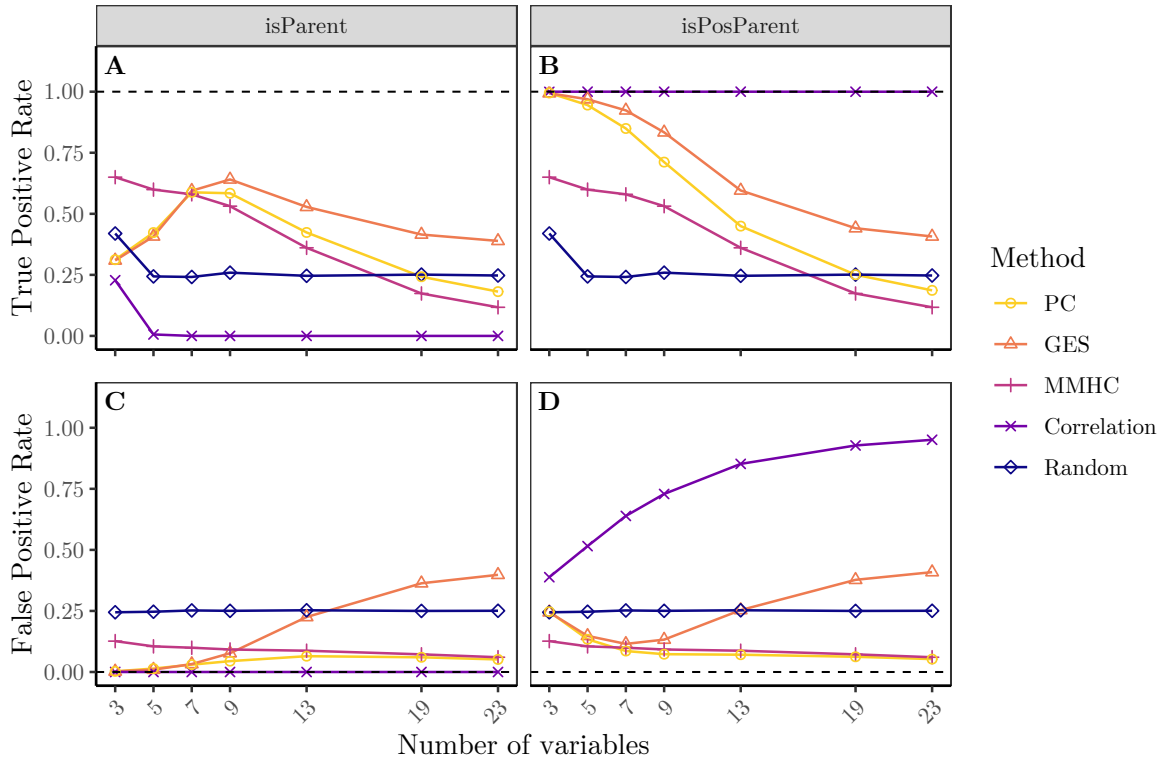


Figure C.2: Results for the discovery of parents (query *isParent*, directed edge in CPDAG) and possible parents (query *isPosParent*, undirected edge in CPDAG) for different numbers of variables. The dashed lines indicate the optimal values: a true positive rate of 1 and a false negative rate of 0. The sample size is fixed at $N = 10,000$ and the density of the true graph at $p = .4$. Each point is the average rate across 500 simulated graphs and datasets.

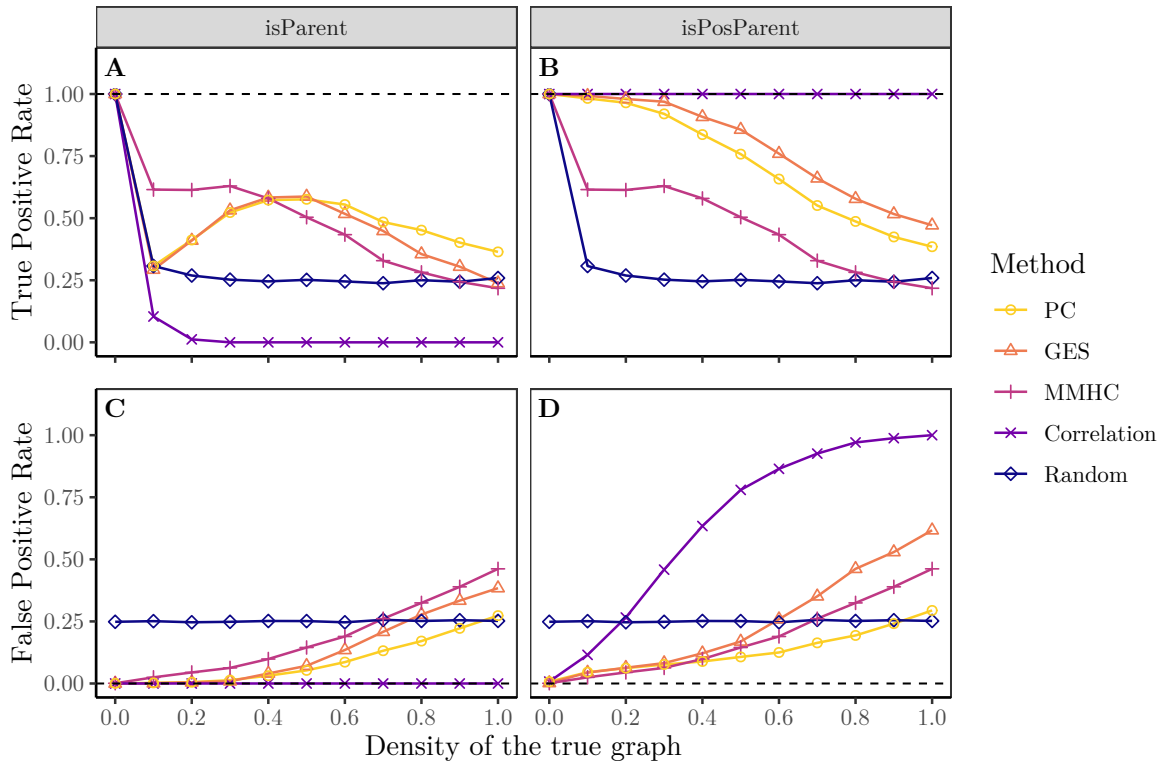


Figure C.3: Results for the discovery of parents (query *isParent*, directed edge in CPDAG) and possible parents (query *isPosParent*, undirected edge in CPDAG) for different degrees of density in the true graph. The dashed lines indicate the optimal values: a true positive rate of 1 and a false negative rate of 0. The sample size is fixed at $N = 10,000$ and the number of variables at $J = 7$. Each point is the average rate across 500 simulated graphs and datasets.

C.3 Causal estimation from an unknown graph under sufficiency

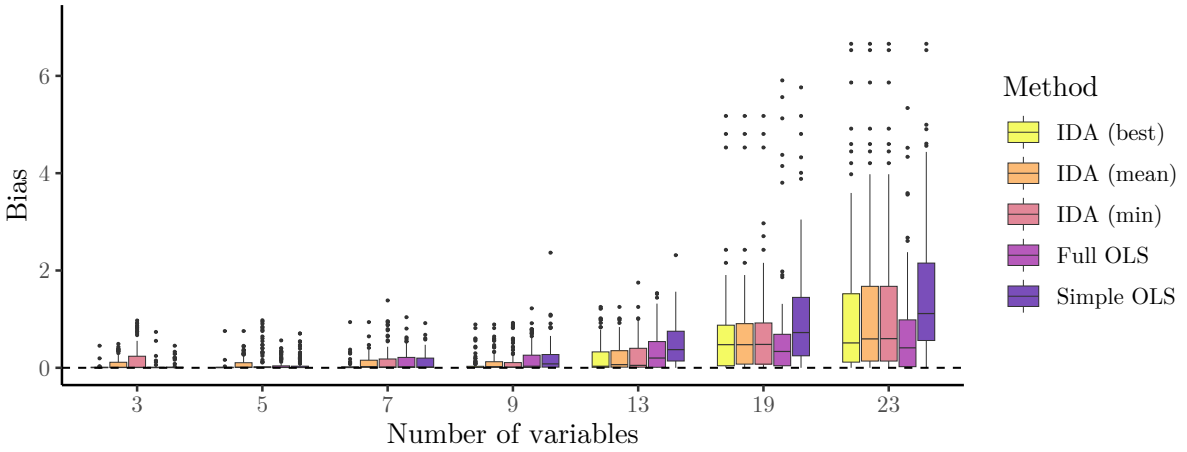


Figure C.4: Results for causal effect estimation from unknown graphs for different numbers of variables. The dashed line indicates the optimal bias of 0. The sample size is fixed at $N = 10,000$ and the density of the true graph at $p = .4$. The boxplots show the distribution of the bias in the estimated coefficient across 100 simulated graphs and datasets. The IDA versions differ in how I summarize the multiset of effects: "best" (infeasible in practice) takes the effect closest to the true effect, "min" takes the minimum, "mean" takes the average of the multiset. I excluded the 15 largest outliers in terms of bias for better readability.

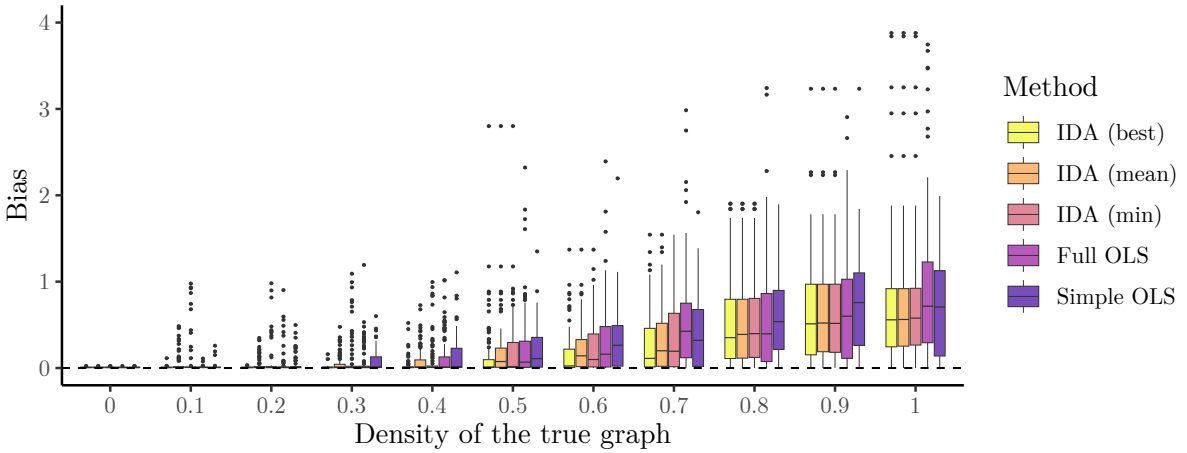


Figure C.5: Results for causal effect estimation from unknown graphs for different degrees of density in the true graph. The dashed line indicates the optimal bias of 0. The sample size is fixed at $N = 10,000$ and the number of variables at $J = 7$. The boxplots show the distribution of the bias in the estimated coefficient across 100 simulated graphs and datasets. The IDA versions differ in how I summarize the multiset of effects: "best" (infeasible in practice) takes the effect closest to the true effect, "min" takes the minimum, "mean" takes the average of the multiset. I excluded the 10 largest outliers in terms of bias for better readability.

C.4 Causal discovery under insufficiency

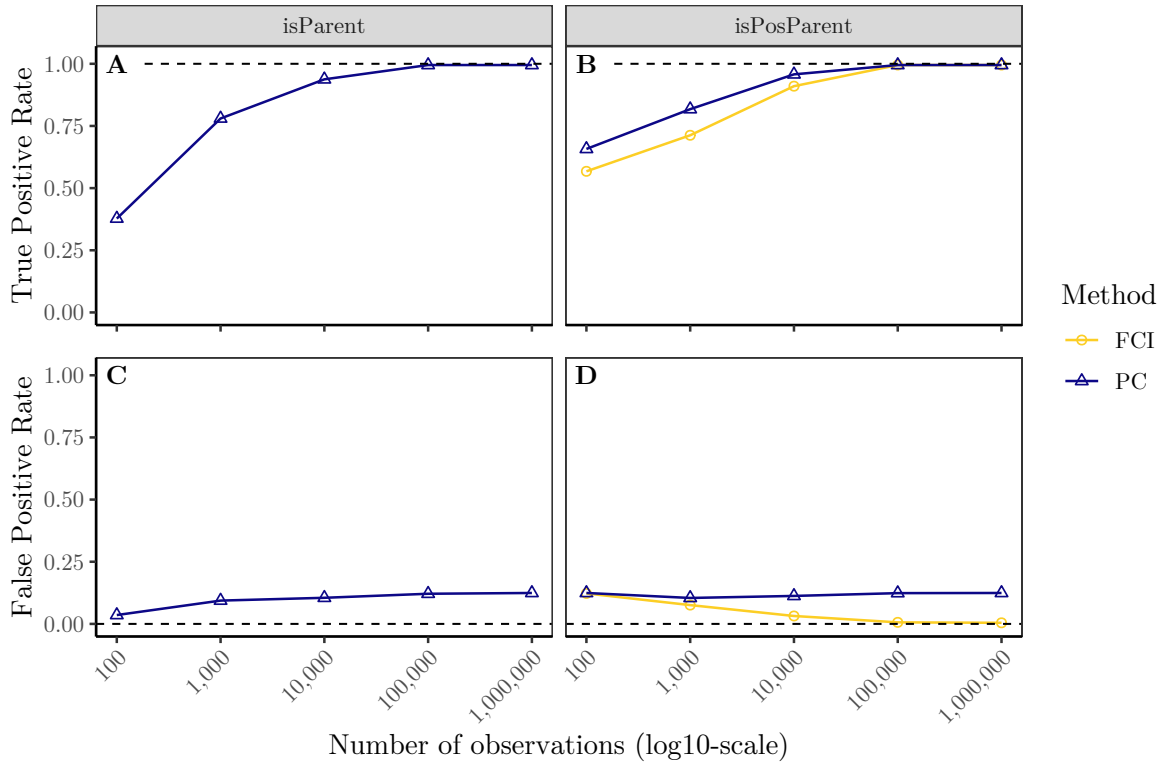


Figure C.6: Results for the discovery of parents (query *isParent*, directed edge in CPDAG) and possible parents (query *isPosParent*, undirected (directed or circle) edge in CPDAG (PAG)) for different sample sizes. The dashed lines indicate the optimal values: a true positive rate of 1 and a false negative rate of 0. The true causal structure is Figure 4.8A. Each point is the average rate across 100 simulated datasets. PAGs only imply ancestral relationships, hence FCI cannot answer the *isParent* query.

C.5 Notes on the implementation of LV-IDA

For LV-IDA, I use the implementation from `https://github.com/dmalinsk/lv-ida`. In rare cases where FCI is only able to direct very few edges, the enumeration of all possible MAGs in LV-IDA can take a very long time. The multiset of these cases typically contains many NA values, but not necessarily exclusively. Nevertheless, if the LV-IDA computation has not finished after 60 minutes, I stop the function execution and interpret the effect as unidentifiable by setting the result to "NA". Also, in even rarer cases, this implementation of LV-IDA gets stuck in one of its recursive functions. My observations suggest that this is the result of FCI returning a cyclic graph, for which LV-IDA does not work (see also the Github documentation). In the main simulation, this occurs in less than 1% of the simulations. Hence, in my final results, I remove these cases and report the first 100 simulations of each setting that successfully computed.

C.6 Application results under different parameters

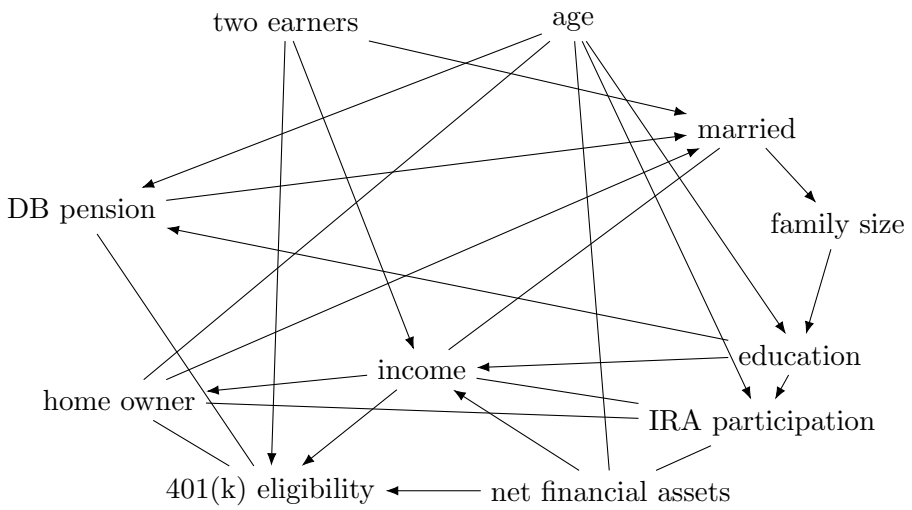


Figure C.7: CPDAG discovered by order-independent PC algorithm with $\alpha = .01$. The undirected edges indicate uncertainty about the edge orientation. The focal edge direction *401(k) eligibility* ← *net financial assets* seems implausible.

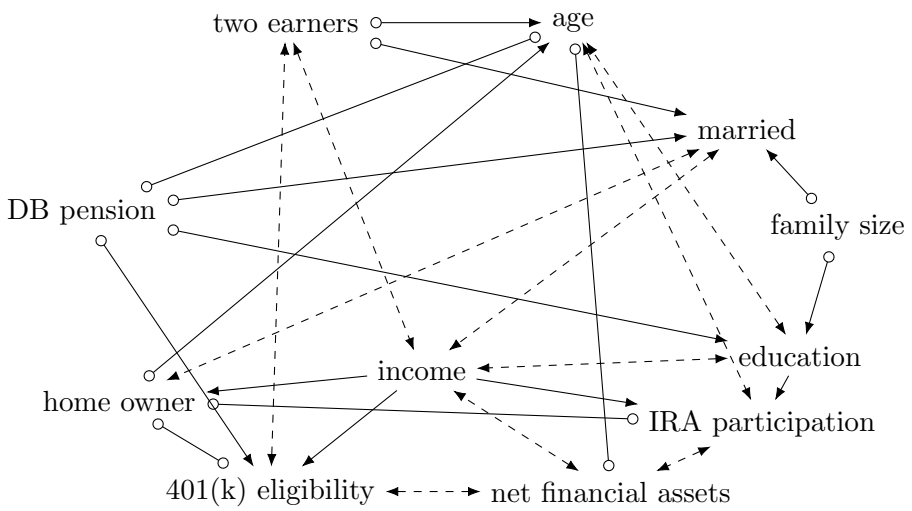


Figure C.8: PAG discovered by order-independent FCI algorithm with $\alpha = .05$. The circles indicate uncertainty about the edge orientation. The bidirected and dashed edges posit the existence of hidden variables. The focal edge between 401(k) eligibility and net financial assets is bidirected, implying a fully confounded relationship and an effect of 0 in LV-IDA.