Applied Causal Inference Powered by ML and AI

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

February 28, 2024

Publisher: Online
Version 0.1.1

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

Colophon
This document was typeset with the help of KOMA-Script and LaTeX using the kaobook class.
The source code of this book is available at:
https://github.com/fmarotta/kaobook

Publisher
First printed in September 2022 by the Authors for Online Distribution

Contents

Notation

$:=$	assignment or definition
$\equiv$	equivalence
$\mapsto$	"maps to" in the definition of a function
$X, Y, Z, ...$	random variables (includes vectors);
	for noise vectors, we use ϵ, ϵ_X, ϵ_j, ...
X'	transpose of a vector;
x	value of a random variable X
P	probability measure
P_X	probability distribution of X
$P_{Y\|X}$	conditional law of Y given X
p	density (either probability mass function
	or probability density function)
p_X	density of P_X
$\mathsf{p}(x)$	density of P_X evaluated at the point x
$\int \mathsf{p}(x)dx$	integral with respect to the base measure
	(Lebesgue for probability density and counting measure for pmf)
$\mathsf{p}(y\|x)$	(conditional) density of $P_{Y\|X=x}$ evaluated at y
z_{a}	the $\mathsf{a}-$quantile of the standard normal distribution
$\{X_i\}_{i=1}^n$	$= (X_1, X_2, ..., X_n)$
	typically an iid. sample of size n with distribution P_X
	$X_{j1}, ..., X_{jn}$ when referring to j^{th} component of X
$\mathrm{E}[X]$	expectation of X
$\mathrm{E}[Y\|X]$	conditional expectation of Y given X
$\mathbb{E}_n[f(Y,X)]$	empirical expectation (e.g. $\mathbb{E}_n[f(Y,X)] := \frac{1}{n}\sum_{i=1}^n f(Y_i, X_i)$
$\mathrm{Var}(X)$	variance of X
$\mathbb{V}_n[g(W)]$	empirical variance
	(e.g. $\mathbb{V}_n[g(W)] = \mathbb{E}_n[g(W)g(W)'] - \mathbb{E}_n[g(W)]\mathbb{E}_n[g(W)]'$)
$\mathrm{Cov}(X,Y)$	covariance of X, Y
$X \perp Y$	orthogonality of X, Y, i.e. $\mathrm{E}(XY') = 0$
$X \perp\!\!\!\perp Y$	independence of X, Y
$X \perp\!\!\!\perp Y \mid Z$	conditional independence of X, Y given Z
$P_{Y(x)} = P_{Y:do(X=x)}$	intervention distribution (can be indexed by M)
$P_{Y(x)\|X} = P_{Y\|X:fix_Y(X=x)}$	counterfactual distribution
G	directed graph
$\mathrm{pa}_{\mathsf{G}}(X)$, $\mathrm{de}_{\mathsf{G}}(X)$, $\mathrm{an}_{\mathsf{G}}(X)$	parents, descendants, and ancestors of node X
	in graph G
$\mathbb{R}^p$	the p-dimensional euclidean space
$\|x\|_1 := \sum_{j=1}^p \|x_j\|$	the ℓ_1-norm in $\mathbb{R}^p$
$\|x\| \equiv \|x\|_2 := \sqrt{\sum_{j=1}^p x_j^2}$	the ℓ_2-norm in $\mathbb{R}^p$
$\|x\|_\infty := \max_{j=1}^p \|x_j\|$	the ℓ_∞-norm in $\mathbb{R}^p$
$\|A\| := \sup_{x\in\mathbb{R}^p\setminus 0} \frac{x'Ax}{x'x}$	the operator norm (maximum eigenvalue) of a matrix A

Preface

This book aims to provide a working introduction to the emerging fusion of modern statistical inference – aka machine learning (ML) or artificial intelligence (AI) – and causal inference methods. The book is aimed at upper level undergraduates and master's-level students as well as doctoral students focusing on applied empirical research. A sufficient background for the core material is one semester of introductory econometrics and one semester of machine learning. We hope the book is also useful to empirical researchers looking to apply modern methods in their work.

The book provides an overview of key ideas in both predictive inference and causal inference and shows how predictive tools are key ingredients to answering many causal questions. We use the term predictive inference to refer to settings where prediction or description is the main goal such that models and estimates do not need a causal interpretation. ML/AI tools are largely designed to answer predictive inference questions, and we provide a high-level overview of popular ML/AI methods (such as Lasso, random forests, and deep neural networks, among others) to provide background for readers less familiar with these methods.

On the causal inference side, we introduce foundational ideas that provide the underpinning to attaching causal interpretations to statistical estimates. We discuss these ideas using the language of potential outcomes, directed acyclical graphs (DAGs), and structural causal models (SCMs). We view the language of potential outcomes, DAGs, and SCMs as complementary. We recognize that readers coming from different backgrounds may be more familiar or disposed to one of potential outcomes, DAGs, or SCMs, but we strongly believe that individuals interested in causal inference should be familiar with each of these frameworks. We find that they all offer useful insights and being able to communicate using each framework allows one to communicate with audiences interested in understanding causality coming from many different backgrounds.

The book has two main sections: Core Material and Advanced Topics. The Core Material provides the main content of the book. After concluding the Core Material, a reader should have an idea of the key ideas underlying both predictive and causal

inference and how to wed these ideas to learn canonical objects in causal inference settings. The Core Material is made up of chapters that move between predictive inference and causal inference, typically by first introducing tools developed for predictive inference and then showing how these tools can be used as inputs to answering causal inference questions. The Advanced Topics then provide extensions of the Core Material to settings with more complicated causal structures, such as instrumental variables models, to settings where understanding heterogeneity in causal effects is the goal, and to specific popular settings in empirical work such as Difference-in-Differences.

Within sections, blocks marked with ⋆ require more substantial preparation in mathematical statistics. We recommend that the reader looking to apply machine learning methods in their work skim or pass them on their first reading and return to them at their leisure.

Short lists of references and study problems are included after each chapter to offer the reader opportunities to investigate further and consolidate their knowledge.

We would like to also acknowledge the tremendous and exceptional help and expertise provided by Philipp Bach, Wenxuan Guo, Andy Haupt, Shunzhuang Huang, David Hughes, Jannis Kück, Malte Kurz, Sven Klassen, Oliver Schacht, Sophie Sun, Vira Semenova, Gulin Tuzcuoglu, Suhas Vijaykumar, John Walker, Thomas Wiemann, Justin Young, and Dake Zhang with both writing and developing supporting Notebooks in R and Python. We are also grateful to Alexander Quispe and Anzony Quispe for developing a Bookdown version of the notebooks and providing other complementary topics and great examples.

Chernozhukov, Hansen, Kallus, Spindler & Syrgkanis

0 Sneak Peek: Powering Causal Inference with ML and AI

A primary question we will be concerned with in this book is: What is the *causal effect* of an action on an outcome? For example, we may want to know what the effect of setting a product's price is on the volume of its sales.[1] To consider this question we scraped data on 9,212 toy cars from Amazon.com. Figure 0.1 shows a log-log-scale scatter plot of the 30-day average price at which each was offered and the reciprocal of its sales rank, a publicly available surrogate for sales volume.[2] We let D denote the log of the price and Y the negative log of the sales rank of a toy car randomly drawn from the population of toy cars sold on Amazon.com. We will use this example to preview the book's chapters and how they come together to enable the reader to power applied causal inference on modern datasets using ML and AI.

1: This effect may be referred to as the price *elasticity* of demand for the product.

2: Were the reader to do such an analysis using internal company data they would use actual sales volumes.

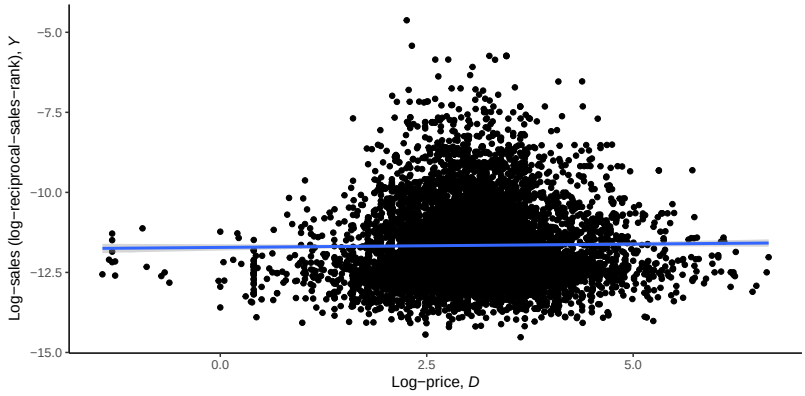


Figure 0.1: Log-prices and log-reciprocal-sales-rank of 9,212 toy cars on Amazon.com along with a linear fit.

In Chapter 1, we present linear regression by ordinary least squares (OLS), which can help us understand the relationship between these two variables. Here it suggests that a unit increase in D is associated with anything between a -0.008 and a 0.050 unit change in Y on average over toy cars; that is, $(-0.008, 0.050)$ is the 95% confidence interval on the slope of the best linear predictor. In words, it suggests one cannot rule out small negative or even slightly positive association between price and sales. It would be incorrect, however, to infer that arbitrarily increasing the price on any one toy car would cause almost no effect on its sales volume, or even increase it.

Instead, economic theory would suggest that the unobserved *potential* log-sales, $Y(d)$, of any *one* toy car should in fact *decrease* as the log-price that one sets, d, increases. In Chapter 2, we

present this notion of potential outcomes and study inference on their averages when actions are *randomized* (or, *exogenous*). For example, we may be interested in the average sales if price were set to a certain level. Unlike the randomized controlled trial (RCT) setting discussed in that chapter, here prices are *not* actually set at random; that is, prices are *endogenous*. Thus, the reason we may see no or a slightly positive association is *confounding* factors that affect both the potential sales at any one price and the particular price that is set. For example, whether a toy car is produced by a brand name or incorporates characters from a popular TV show might increase sales at any one price as well as lead the seller to choose a higher price, whether in anticipation of higher demand or because of higher production or licensing costs.

We formalize this notion of confounding in Chapter 5 and consider causal inference on averages of potential outcomes when one observes *all* confounding variables, W. In Chapter 6, we go on to consider a *linear* structural equation,

$$Y(d) = \alpha d + U, \tag{0.0.1}$$

which posits that, on average, log-sales at any one log-price is a linear function of the log-price, aside from the idiosyncrasies U of each one toy car. Within this structural equation, we interpret α as the causal effect of d on Y; that is, the effect of a change in d on Y produced by intervening in the system to change d while holding all other determinants of sales constant. This causal effect is generally not recovered from regression of observed Y on observed price, D, as observed price is set in the market and plausibly related to unobserved factors U.

In our simple linear structural equation, the assumption that W accounts for all confounding leads us to conclude that we have

$$Y = \alpha D + g(W) + \varepsilon, \quad \mathrm{E}[\varepsilon \mid D, W] = 0 \tag{0.0.2}$$

for some function $g(W)$. Thus, after all of our causal modeling and assumptions, what remains is inference on a coefficient in a possibly complex regression model of Y on D and W, all of them *observed* variables. That is, under our causal modeling and assumptions, making statistical inferences (such as constructing estimates and confidence intervals) on α in Eq. (0.0.2) from data on (Y, D, W) would be *causal inference*. (0.0.1) is the simplest of structural equations – to understand more complex structures we consider *systems* of equations, and in Chapter 7 even *nonlinear* structural equations.

To explain how we power such causal inference with ML and AI, let us now return to the question of *what is W in the first place*? There are many features we can observe about each toy car on Amazon.com in addition to its price and sales: all the text on the product page such as name and description, the product subcategory (beyond being a toy), the brand, the color, and the dimensions and weight of both the item and its packaging. What features can we make use of, and how?

Classical methods, like OLS, (Chapter 1) allow us to conduct inference on α when Eq. (0.0.2) is a linear regression with moderately high dimensions, that is, when W is a p-dimensional random vector, $g(W) = \beta_1 + \beta_2' W$, and p is much smaller than the number of observations we have (here, 9,212). Letting $g(W) = \beta_1 + \beta_2' W$ in Eq. (0.0.2) we obtain a *linear model*. There are 243 product subcategories for our toy cars. Consider identifying each with a number in $1, \ldots, 243$ and letting W be a 243-dimensional vector with a 1 in the index corresponding to the product's subcategory and 0 elsewhere. OLS regression of Y on D and this particular W explains 7.5% of Y's variance (as measured by adjusted R^2) and gives a 95% confidence interval on α of $(-0.026, 0.036)$. These results are not very different from what we inferred in the observed association between Y and D without adjusting for any confounding effects, but at least the upper bound is smaller – we indeed do not believe a positive effect is realistic.

Perhaps we need to control for more confounding effects than just subcategory membership. However, even without departing from linearity, OLS no longer provides reliable inference if we include too many features in W. Letting $g(W) = \beta_1 + \beta_2' W$ in Eq. (0.0.2) with a high-dimensional W, that is, where d is comparable to or bigger than the number of observations, we obtain a *linear model with high-dimensional controls*. In Chapter 3, we present more advanced ML methods than OLS: predictive inference in high dimensions using regularized linear regression. The use of regularized linear regression may improve prediction relative to OLS but introduces biases that imperil inference on coefficients. In Chapter 4, we show how to remedy this bias when making inferences on any one coefficient. In the context of causal inference, this setup allows us to potentially handle very many confounders, and the hope is that we can then more reliably justify having accounted for *all* confounders. In a nutshell, in the setting of Eq. (0.0.2), if we take $\tilde{Y}$ and $\tilde{D}$ to be the *residuals* from a modern high-dimensional linear regression of Y on $(1, W)$ and of D on $(1, W)$, respectively, then OLS regression of $\tilde{Y}$ on $\tilde{D}$ yields valid inference on α even when

W is high-dimensional.

Consider letting W be a 11546 dimensional vector including not only the indicator of subcategory but also the item's physical dimensions, transformed by log and expanded up to third power of the logarithms, missingness indicators, the interaction of these dimension features with subcategory, the indicator of brand (among 1827 brands). In this case, p is greater than the number of observations n. Using the methods we present in Chapter 4 to leverage this high-dimensional W in this particular set up, we obtain a 95% confidence interval on α of $(-0.10, -0.029)$. The confidence interval including only negative values is in concordance with the intuition that intervening to increase price would decrease demand. At the same time, we may still worry that a linear model is too restrictive, in essence allowing us only to control linearly for pre-specified confounders.[3]

In Chapter 9, we present nonlinear ML methods for regression: trees, ensembles, and neural nets. Compared to predicting log-price and log-sales with LASSO, using these methods (with a 2083-dimensional feature vector omitting the expansions and interactions needed for linear models) increases the R^2 by 25-53% and 89-189% (evaluated using 5-fold cross-validated R^2). Clearly these methods offer significant predictive improvements in this dataset. However, such nonlinear methods have no clear parameter to extract, no coefficient to inspect. While making excellent predictions, it is not immediately clear how to use them to make valid statistical inferences on finite-dimensional parameters, like average effects. We tackle that question in Chapter 10. Letting $g(W)$ be an arbitrary nonlinear function in Eq. (0.0.2) gives rise to what is called the *partially linear model*, which strikes a nice balance between structure and flexibility: the causal-effect part of the model is simple and interpretable – for each unit increase in action we get α increase in outcome – while the confounding part, which we have no interest in interpreting, can be almost-arbitrarily complex.[4] In the setting of Eq. (0.0.2), it turns out we can keep the method of residual-on-residual OLS inference, but using residuals from advanced *nonlinear* regressions, as long as we fit these regressions on parts of the data that exclude where we use them to make predictions and produce the residuals. This is *double machine learning* or *debiased machine learning* or *double/debiased machine learning*[5] for the partially linear model. Using DML together with gradient-boosted-tree regression to make inferences on the price elasticity α in this example yields a confidence interval of $(-0.139, -0.074)$, suggesting an effect whose direction agrees

3: One may include *pre-specified transformations* of confounders as well as discussed in Chapter 1.

4: Luckily, even if the partially linear assumption fails, estimates still reflect some average of the *causal* effects of increasing *all* prices by a small amount, provided we have accounted for all confounding effects in in W. See Remarks 10.2.2 and 10.3.3.

5: We will use these terms interchangeably and abbreviate them with *DML*.

even more strongly with our intuition, which can be attributed to these more powerful predictive methods being able to better account and correct for the confounding effects that pushed the apparent direction upward.

It is still unclear, however, whether the numeric features we observe can reliably capture all of the confounding effects – if they cannot, then no regression, no matter how flexible, can help. This problem – getting the right data to enable causal inference – is a common challenge when dealing with observational data. It is in using all the available data, where modern AI along with the tools we develop in this book come together to uniquely enable powerful causal inferences using modern observational data sets. Modern data sets are rich, containing far more than just numeric features. This data set, for example, contains text on each product – descriptions that capture many important features about each product that are not clearly tabulated but must be inferred by reading the text. Luckily, modern AI has made great inroads in recent years in machine cognition of text, images, videos, and other rich data.

In Chapter 11, we discuss how these powerful tools can be used in concert with DML. BERT is a large language model leveraging a deep learning architecture known as *transformers* and achieving impressive performance on natural-language-processing benchmarks. Using neural-net-based predictive models for log-price and log-sales built on top of BERT results in a 12-37% and 4-59% increase in cross-validated R^2, respectively, relative to the nonlinear models using only numeric features in the data. The non-numeric features in the data therefore seem to account for more than the baseline numeric factors of products in predicting price and sales. Using DML for the partially linear model together with these models that use the non-numeric features, we are able to make causal inferences that account for confounding factors reflected in the rich text on the product page for each toy car. Proceeding in this way, as we explain in greater detail in Chapter 11, we obtain a confidence interval on α of $(-0.21, -0.13)$. That we get a more negative estimate here again suggests that there were residual confounding effects inducing a spurious positive relationship between price and sales that we could only have controlled for and counteracted by using AI to account for the rich text data.

While it is relatively easy to validate predictive models' performance by using held-out test sets and cross-validation, it is difficult – impossible, even – to definitively validate a causal effect, as it will inevitably rest on fundamentally untestable assumptions. Nonetheless, we can have greater confidence in

estimates that correctly and fully leverage the available data and do not rely on unnecessary parametric assumptions. Estimates based on DML on top of AI allow us to do just that. We can use rich data without imposing strong functional form restrictions and importantly can do so without imperiling guarantees on valid statistical inference. The Core material outlines the basic ideas and provides fundamental results for using DML with AI learners to estimate and do inference for low-dimensional causal effects.

The Advanced Topics section includes chapters that expand upon the basic material from the Core chapters. In the Core material, we discuss more complex structures than the partially linear model introduced in this preview, but do inference essentially only when all relevant variables are observed. In Chapter 12, we present alternative ways to identify causal effects when we do not believe we observe all confounders – techniques such as sensitivity analysis, instrumental variables, and proxy controls, and we provide methods for causal inference in such settings in Chapter 13. These tools allow us to have confidence in causal estimates that leverage special structure like instruments or proxies without additionally making unnecessary parametric assumptions and with the ability to leverage rich data using powerful AI. In many examples, one may wish to understand heterogeneity in causal effects such as how causal effects differ across observed predictors. Chapter 14 covers DML inference on quantities that characterize this heterogeneity, and Chapter 15 goes beyond inference on low-dimensional causal parameters and discusses learning heterogeneous causal effects from rich individual-level data and even personalizing treatments based on such data. Finally, we consider application of DML in conjunction with two popular methods for identifying causal effects – difference-in-differences and regression discontinuity designs – in Chapter 16 and Chapter 17 respectively.

After studying the book, the reader should also be able to understand and employ DML in many other applications that are not explicitly covered. In the toy car example we focused on sales, but sales may not reflect demand when we reach the limits of on-hand inventory, something known as right-censoring. Censoring is an example of data coarsening, and mathematically it is not too dissimilar from the missingness of potential outcomes for actions not taken. Similarly, we may want to look at distributional effects beyond averages, like effects on the quantiles of sales. DML can often be applied to these problems and there is active research on applying it to ever more intricate problems.

There are also topics beyond our scope. We started by saying we focus on the causal effect of an action on an outcome – a broader yet much more challenging question is, among multiple variables, discovering which have causal effects on which. While we *do* discuss the use of directed acyclic graphs in Chapter 7 and Chapter 8, we only use them to represent assumed structure and only briefly mention how one might try to learn causal structure directly from data, which is the subject of *causal discovery*.

Our aim is rather focused: present the building blocks of predictive inference and of causal inference and illustrate their effective and correct use in concert in a way that allows readers to employ them in real, practical settings. The book interweaves the two kinds of inference, with many real-data examples with code notebooks. We hope the outcome is that we reach an endpoint where the reader is ready to power causal inferences with ML and AI and be able to draw valid, reliable inferences in practice using rich modern data.

Core Material

Predictive Inference with Linear Regression in Moderately High Dimensions

1

"Infer: to form an opinion or guess that something is true because of the information that you have."

– Cambridge Dictionary [1].

Least squares, and particularly its application to linear regression, is one of the most widely used statistical methods. It is an intuitive tool for predictive inference and for establishing association. The method of least squares was introduced in the 1800s by L. Legendre and C.F. Gauss. Here we review properties of least squares estimation of linear models in moderately high-dimensional problems, focusing on its use in predictive inference and for establishing association. This treatment provides a starting point for our subsequent review of modern statistical (machine) learning methods, which will relax our assumption on dimensionality as well as consider nonlinear models.

1.1 Foundation of Linear Regression

Regression and the Best Linear Prediction Problem

We consider a scalar random variable Y, an outcome of interest, and a p-vector of covariates

$$X = (X_1, \ldots, X_p)'.$$

We assume that a constant of 1 is included as the first component in X; that is, $X_1 = 1$.

For theoretical purposes, we first consider linear regression in the population. Working in the population means that we have access to unlimited amounts of data to compute population moments – such as $\mathrm{E}[Y]$, $\mathrm{E}[YX]$, and $\mathrm{E}[XX']$ – and that we can define "ideal" quantities. After defining these ideal quantities, we then turn to estimation with real data, which we will take to be a sample of observations drawn from the population.

Our first goal is to construct the best linear prediction rule for Y using X. That is, the predicted value of Y given X will be of the linear form:

$$\sum_{j=1}^{p} \beta_j X_j = \beta' X, \text{ for } \beta = (\beta_1, \ldots, \beta_p)',$$

where β's are called the regression parameters or coefficients.

We define β as any solution to the *Best Linear Prediction (BLP) Problem*,

$$\min_{b \in \mathbb{R}^p} \mathrm{E}\left[(Y - b'X)^2\right],$$

where we minimize the Expected or Mean Squared Error (MSE) for predicting Y using the linear rule

$$b'X = \sum_{j=1}^{p} b_j X_j, \quad b = (b_1, \ldots, b_p)'.$$

The solution to this optimization problem, $\beta'X$, is called the *Best Linear Predictor* (BLP) of Y using X. This jargon refers to the fact that $\beta'X$ is the best, according to MSE, linear prediction rule for Y among all possible linear prediction rules.

We can compute β by solving the first order conditions for the BLP problem:

$$\mathrm{E}\left[(Y - \beta'X)X\right] = 0.$$

Figure 1.1: The only known portrait of Legendre (a friendly caricature) by Julien Léopold Boilly. Source: Wikipedia. The hairstyle is amazing.

These equations are also referred to as the Normal Equations and are obtained by setting the derivative of the objective function $b \mapsto \mathrm{E}\left[(Y - b'X)^2\right]$ with respect to b equal to zero. Thus, any solution to the BLP problems satisfies the Normal Equations.

Defining the regression error or residual as

$$\varepsilon := (Y - \beta'X),$$

we can write the Normal Equations as

$$\mathrm{E}[\varepsilon X] = 0, \quad \text{or equivalently} \quad \varepsilon \perp X.$$

Note that we use $\perp$ to denote orthogonality between random variables, and $\perp\!\!\!\perp$ to denote full statistical independence.. That is, for random variables U and V, $U \perp V$ means $\mathrm{E}[UV] = 0$. Further, if U is a *centered random variable*, then $U \perp\!\!\!\perp V$ implies $U \perp V$, but the reverse implication is not true in general. Indeed, let $U \sim N(0,1)$ and $V = U^2 - 1$, then $U \perp V$ but $U \not\perp\!\!\!\perp V$.

Therefore, the BLP problem provides a simple decomposition of Y:

$$Y = \beta'X + \varepsilon, \quad \varepsilon \perp X,$$

where $\beta'X$ is the part of Y that can be linearly predicted or explained with X, and ε is whatever remains – the so-called unexplained or residual part of Y.

Best Linear Approximation Property

The normal equation $\mathrm{E}[(Y - \beta'X)X] = 0$ implies by the law of iterated expectations that

$$\mathrm{E}[(\mathrm{E}[Y \mid X] - \beta'X)X] = 0.$$

Therefore, the BLP of Y is also the BLP for the conditional expectation of Y given X. This observation is important and motivates the use of various transformations of regressors to form X.

From Best Linear Predictor to Best Predictor

Here we explain the use of constructed features or regressors. If W are "raw" regressors/features, *technical (constructed) regressors* are of the form

$$X = T(W) = (T_1(W), ..., T_p(W))',$$

where the set of transformations $T(W)$ is sometimes called the *dictionary* of transformations. Example transformations include polynomials, interactions between variables, and applying functions such as the logarithm or exponential. In the wage analysis

reported below, for example, we use quadratic and cubic transformations of experience, as well as interactions (products) of these regressors with education and geographic indicators.

The main motivation for the use of constructed regressors is to build *more flexible and potentially better* prediction rules. The potential for improved prediction arises because we are using prediction rules $\beta' X = \beta' T(W)$ that are *nonlinear* in the original raw regressors W and may thus capture more complex patterns that exist in the data. Conveniently, the prediction rule $\beta' X$ is still linear with respect to the parameters, β, and with respect to the constructed regressors $X = T(W)$.

In the population, the *best predictor* of Y given W is

$$g(W) = \mathrm{E}[Y \mid W],$$

the *conditional expectation* of Y given W. The *conditional expectation function* $g(W)$ is also called the *regression function* of Y on W. Specifically, the conditional expectation function $g(W)$ solves the best prediction problem[1]

$$\min_{m(W)} \mathrm{E}\left[(Y - m(W))^2\right].$$

Here we minimize the MSE among all prediction rules $m(W)$ (linear or nonlinear in W).

As the conditional expectation solves the same problem as the best linear prediction rule among a larger class of candidate rules, the conditional expectation generally provides better predictions than the best linear prediction rule.[2]

By using $\beta' T(W)$, we are implicitly approximating the best predictor $g(W) = \mathrm{E}[Y|W]$. Indeed, for any parameter b,

$$\mathrm{E}\left[(Y - b'T(W))^2\right] = \mathrm{E}\left[(g(W) - b'T(W))^2\right] + \mathrm{E}\left[(Y - g(W))^2\right],$$

That is, the mean squared prediction error is equal to the mean squared approximation error of $b'T(W)$ to $g(W)$ plus a constant that does not depend on b. Therefore, minimizing the mean squared prediction error is the same as minimizing the mean squared approximation error. Thus, the BLP $\beta' T(W)$ is the *Best Linear Approximation* (BLA) to the best predictor, which is the regression function $g(W)$. Finally, as the dictionary of transformations $T(W)$ becomes richer, the quality of the approximation of the BLA $\beta' T(W)$ to the best predictor $g(W)$ improves.

1: This result follows by rewriting the objective function as

$$\min_{m(W)} \mathrm{E}[\mathrm{E}[(Y - m(W))^2 \mid W]],$$

noting that it is equivalent to

$$\mathrm{E}[\min_{\mu \in \mathbb{R}} \mathrm{E}[(Y - \mu)^2 \mid W]],$$

and deriving the first order conditions for the inner minimization: $E(Y \mid W) - \mu = 0$.

2: Unless the conditional expectation function turns out to be linear, in which case the conditional expectation and best linear prediction rule coincide.

Example 1.1.1 (Approximating a Smooth Function with a Polynomial Dictionary) Suppose $W \sim U(0,1)$ where U denotes the continuous uniform distribution, and

$$g(W) = \exp(4 \cdot W).$$

We use

$$T(W) = \underbrace{(1, W, W^2, \ldots, W^{p-1})'}_{p \text{ terms}}$$

to form the BLA/BLP, $\beta' T(W)$. Figure 1.2 provides a sequence of panels that illustrate the approximation properties of the BLA/BLP corresponding to p =2, 3, and 4:

- With $p = 2$ we get a linear in W approximation to $g(W)$. As the figure shows, the quality of this approximation is poor.
- With $p = 3$ we get a quadratic-in-W approximation to $g(W)$. Here, the approximation quality is markedly improved relative to $p = 2$ though approximation errors are still clearly visible.
- With $p = 4$ we get a cubic-in-W approximation to $g(W)$, and the quality of approximation appears to be excellent.

This simple example highlights the motivation for using nonlinear transformations of raw regressors in linear regression analysis. What this example does not yet reveal are the *statistical* challenges of dealing with higher and higher dimension p when learning from a finite sample.

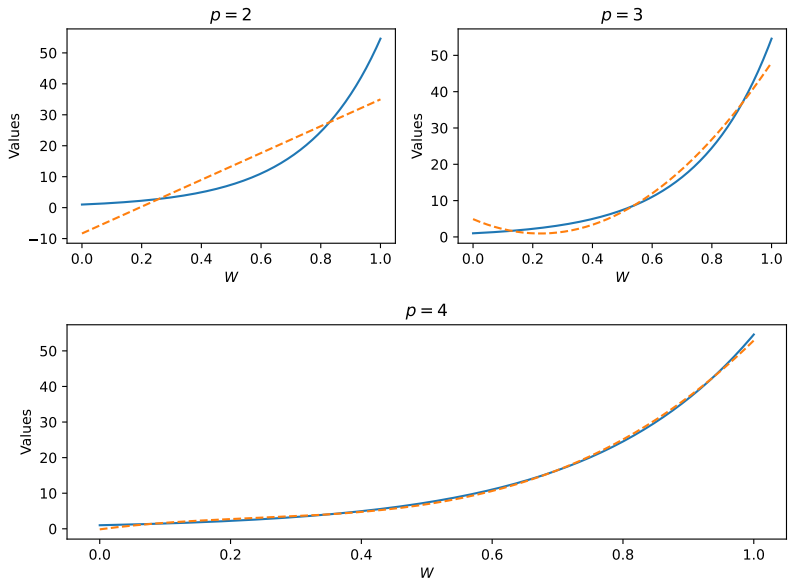


Figure 1.2: Refinements of Approximation to Regression Function $g(W)$ by using polynomials of W.

There are many ways of generating flexible approximations, which are studied by approximation theory and nonparametric statistical learning theory.[3]

3: See, e.g., Tsybakov [2]. We will also consider nonlinear approximations using trees and neural networks in Chapter 9.

When we have multiple variables, we may generate transformations of each of the variables and employ interactions – products involving these terms. As a simple concrete example, consider a case with two raw regressors, W_1 and W_2. We could build polynomials of second order in each of the raw regressors – $(1, W_1, W_1^2)$, $(1, W_2, W_2^2)$. We may then collect these variables along with the interaction in the raw regressors, $W_1 W_2$ in a vector

$$(1, W_1, W_2, W_1^2, W_2^2, W_1 W_2)$$

for use in a regression model. There are, of course, many other possibilities such as considering higher order polynomial terms, e.g. W_1^3; higher order interactions, e.g. $W_1^2 W_2$; and other nonlinear transformations, e.g. $\log(W_1)$.

1.2 Statistical Properties of Least Squares

The Best Linear Prediction Problem in Finite Samples

In practice, the researcher does not have access to the entire population, but observes only a sample

$$\{(Y_i, X_i)\}_{i=1}^n = ((Y_1, X_1), ..., (Y_n, X_n)).$$

We assume that this sample is a random sample from the distribution of (Y, X). Formally, this condition means that the observations were obtained as realizations of independently and identically distributed (iid) copies of the random variable (Y, X). By treating the observations as iid, we are modeling the data as independent random draws with replacement from a population. Other possible models include sampling without replacement from a finite population, stratified sampling, observations of a process over time, and other schemes or scenarios that induce dependence between the data points. For the most part, we focus on the iid model throughout this book.

We construct the best in-sample linear prediction rule for Y using X analogously to the population case by replacing theoretical expected values, E, with empirical averages, $\mathbb{E}_n$. Specifically,

$\mathbb{E}_n$ abbreviates the notation $\frac{1}{n}\sum_{i=1}^n$. For example,

$$\mathbb{E}_n[f(Y, X)] := \frac{1}{n}\sum_{i=1}^n f(Y_i, X_i).$$

given X, our predicted value of Y will be

$$\sum_{j=1}^{p} \hat{\beta}_j X_j = \hat{\beta}' X, \text{ for } \hat{\beta} = (\hat{\beta}_1, ..., \hat{\beta}_p)',$$

where $\hat{\beta}$ is any solution to the *Best Linear Prediction Problem in the Sample*, also known as Ordinary Least Squares (OLS):

$$\min_{b \in \mathbb{R}^p} \mathbb{E}_n[(Y - b'X)^2].$$

We often use the hat decoration $\hat{}$ for quantities that depend on the sample. For example, β denotes the BLP in the population, while $\hat{\beta}$ is the BLP in the sample.

That is, $\hat{\beta}$ minimizes the sample MSE for predicting Y using the linear rule $b'X$. The $\hat{\beta}$'s are called the sample regression coefficients.

We can compute $\hat{\beta}$ as any solution to the Sample Normal Equations,

$$\mathbb{E}_n[X(Y - X'\hat{\beta})] = 0,$$

which are obtained as the first order conditions to the Best Linear Prediction Problem in the Sample. Further, defining the residuals (or, in-sample regression errors) as

$$\hat{\varepsilon}_i := (Y_i - \hat{\beta}' X_i),$$

we obtain the decomposition

$$Y_i = X_i'\hat{\beta} + \hat{\varepsilon}_i, \quad \mathbb{E}_n[X\hat{\varepsilon}] = 0,$$

where $X_i'\hat{\beta}$ is the predicted or explained part of Y_i, and $\hat{\varepsilon}_i$ is the unexplained or residual part.

Properties of Sample Linear Regression

The best linear prediction rule in the population is $\beta' X$, and a key question is whether $\hat{\beta}' X$ estimates (that is, approximates using data) $\beta' X$ well.

The best linear prediction rule is also the best linear rule for predicting future values of Y given a new draw X, when new (Y, X) are sampled from the same population. Therefore, if we can approximate the best linear prediction rule in the population, we can also approximate the best linear prediction rule for predicting outcomes given future X's sampled from the population.

The fundamental statistical issue is that we are trying to estimate p parameters, $\beta_1, ..., \beta_p$, without imposing any assumptions on these parameters. Intuitively, to estimate each parameter

well, we need many observations per parameter. This intuition suggests that n/p should be large, or, equivalently that p/n should be small, in order for estimation error to be small. The following result captures this intuition more formally.

Theorem 1.2.1 (Approximation of BLP by OLS) *Under regularity conditions,*[a]

$$\sqrt{\mathrm{E}_X[(\beta' X - \hat{\beta}' X)^2]} = \sqrt{(\hat{\beta} - \beta)' \mathrm{E}_X[XX'](\hat{\beta} - \beta)}$$

$$\leq const_{\mathrm{P}} \cdot \sqrt{\mathrm{E}\varepsilon^2} \sqrt{\frac{p}{n}},$$

where E_X *is the expectation with respect to* X *alone, the inequality holds with probability approaching 1 as* $n \to \infty$*, and* $const_{\mathrm{P}}$ *is a constant that depends on the distribution of* (Y, X)*.*

[a] See Notes (Section 1.5) for references.

Theorem 1.2.1 says that, for nearly all realizations of data, the sample linear regression is close to the population linear regression if n is large and p is much smaller than n:

$$\sqrt{\mathrm{E}_X[(\beta' X - \hat{\beta}' X)^2]} \approx 0.$$

In other words, under our requirement of p/n small, the sample BLP approximates the population BLP well.

Given indexed random variables (vectors, elements) A_n and B_n in a metric space equipped with metric d, the notation $A_n \approx B_n$ means that the distance between A_n and B_n concentrates around 0 – formally, that $\lim_{n\to\infty} \mathrm{P}(d(A_n, B_n) \leq \varepsilon) = 1$ for each $\varepsilon > 0$.

Analysis of Variance

Analysis of variance involves the decomposition of the variation of Y into explained and unexplained parts. Explained variation is a measure of the predictive performance of a model. This decomposition can be conducted both in the population and in the sample.

The main idea is to use the previous decomposition of Y,

$$Y = \beta' X + \varepsilon, \quad \mathrm{E}[\varepsilon X] = 0,$$

to decompose the variation in Y into the sum of *explained variation* and *residual variation*:

$$\mathrm{E}[Y^2] = \mathrm{E}[(\beta' X)^2] + \mathrm{E}[\varepsilon^2].$$

The quantity

$$\mathrm{MSE}_{pop} = \mathrm{E}[\varepsilon^2]$$

is the population MSE. The ratio of the explained variation to the total variation is the population R^2:

$$R^2_{pop} := \frac{\mathrm{E}[(\beta' X)^2]}{\mathrm{E}[Y^2]} = 1 - \frac{\mathrm{E}[\varepsilon^2]}{\mathrm{E}[Y^2]} \in [0, 1].$$

That is, R^2_{pop} is the proportion of variation of Y explained by the BLP.

Remark 1.2.1 The "standard" definition of R^2 assumes that either we work with a centered Y, that is, we recenter Y such that $\mathrm{E}[Y] = 0$. (However, our definition above does not require this property).*centered random variable*

The decomposition of the variance in the sample proceeds analogously. Using the representation

$$Y_i = \hat{\beta}' X_i + \hat{\varepsilon}_i$$

and the orthogonality condition $\mathbb{E}_n[X\hat{\varepsilon}] = 0$ provided by the sample Normal Equations, we obtain the decomposition

$$\mathbb{E}_n[Y^2] = \mathbb{E}_n[(\hat{\beta}' X)^2] + \mathbb{E}_n[\hat{\varepsilon}^2].$$

Thus, we can define the sample MSE,

$$\mathrm{MSE}_{sample} = \mathbb{E}_n[\hat{\varepsilon}^2],$$

and the sample R^2,

$$R^2_{sample} := \frac{\mathbb{E}_n[(\hat{\beta}' X)^2]}{\mathbb{E}_n[Y^2]} = 1 - \frac{\mathbb{E}_n[\hat{\varepsilon}^2]}{\mathbb{E}_n[Y^2]} \in [0, 1].$$

By the law of large numbers and Theorem 1.2.1, when p/n is small, we have the following approximations:

$$\mathbb{E}_n[Y^2] \approx \mathrm{E}[Y^2], \quad \mathbb{E}_n[(\hat{\beta}' X)^2] \approx \mathrm{E}[(\beta' X)^2], \quad \mathbb{E}_n[\hat{\varepsilon}^2] \approx \mathrm{E}[\varepsilon^2].$$

Thus, when p/n is small and n is large, the sample fit measures are good approximations to population fit measures:

$$\mathrm{MSE}_{sample} \approx \mathrm{MSE}_{pop} \text{ and } R^2_{sample} \approx R^2_{pop}.$$

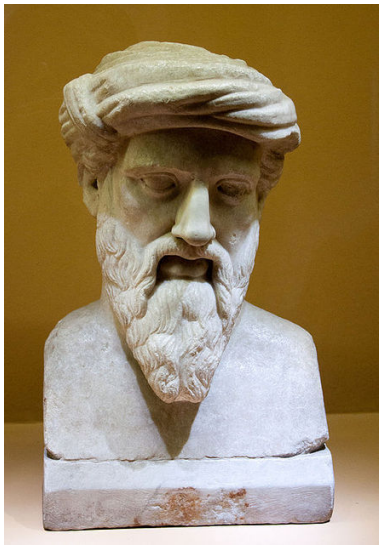


Figure 1.3: Pythagoras of Samos invented least squares and analysis of variance for the case of $n = 2$ and $p \leq 2$ around 570 BC. He was therefore the first known machine learner.

Overfitting: What Happens When p/n Is Not Small

When p/n is not small, the picture about predictive performance of the in-sample BLP becomes inaccurate and possibly misleading. In this setting, the in-sample linear predictor can be substantially different from the population BLP.

Consider an extreme example where $p = n$ and all variables in X are linearly independent. In this case, we have

$$\text{MSE}_{sample} = 0 \text{ and } R^2_{sample} = 1$$

no matter what MSE_{pop} and R^2_{pop} are. E.g. we could have $R^2_{sample} = 1$ even if $R^2_{pop} = 0$. Therefore, here we have an extreme example of *overfitting*, where the in-sample predictive performance overstates the out-of-sample predictive performance of the linear model. The following example illustrates less extreme cases.

Example 1.2.1 (Overfitting Example) Suppose $X \sim N(0, I_p)$ and $Y \sim N(0, 1)$ are statistically independent. It follows that the best linear predictor of Y is $\beta' X = 0$ and that $R^2_{pop} = 0$.

- If $p = n$, then the typical R^2_{sample} is $1 \gg 0$.
- If $p = n/2$, then the typical R^2_{sample} is about $.5 \gg 0$.
- If $p = n/20$, then the typical R^2_{sample} is about $.05 > 0$.

These results can be deduced by simulation or analytically.

The Linear Model Overfitting R Notebook and the Linear Model Overfitting Python Notebook contain code for the numerical experiment.

Provided $p < n$, better measures of out-of-sample predictive ability are the "adjusted" R^2 and MSE:[4]

$$\text{MSE}_{adjusted} = \frac{n}{n-p}\mathbb{E}_n[\hat{\varepsilon}^2], \quad R^2_{adjusted} := 1 - \frac{n}{n-p}\frac{\mathbb{E}_n[\hat{\varepsilon}^2]}{\mathbb{E}_n[Y^2]}.$$

4: The adjustment factor $\frac{n}{n-p}$ is derived in a homogeneous model, so that $\text{E}[\text{MSE}_{adjusted}] = \text{MSE}_{pop}$, see e.g., p. 8 in [3] for the derivation.

The adjustment by $\frac{n}{n-p}$ corrects for overfitting and provides a more accurate assessment of predictive ability of the linear model in Example 1.2.1 and more generally under the assumption of homogeneous ε. The intuition is that models with many parameters increase the in-sample fit and potentially cause overfitting. Hence, the number of parameters is incorporated in the definition of $\text{MSE}_{adjusted}$ and $R^2_{adjusted}$ in an attempt to account for this phenomenon.

Measuring Predictive Ability by Sample Splitting

How should we measure the predictive ability of the linear model (or other nonlinear models that we will discuss) more reliably, even in cases when p/n is not small?

A general way to measure predictive performance is to perform *data splitting*. The idea can be summarized in two parts:

1. Use a random part of a dataset, called the training sample, for estimating/training the prediction rule.
2. Use the other part, called the testing sample, to evaluate the quality of the prediction rule, recording out-of-sample mean squared error and R^2.

Generally, a predictive model is trained on a sample and the real test of its predictive ability happens when "new, unseen" observations arrive. With new observations in hand, we learn how far off our predictions are, when compared to the realized values. By partitioning the data set into two parts, we preserve an "unseen" set of observations on which to test our model, mimicking this process of ex-post performance assessment.[5]

5: If the "test set" is used many times to evaluate models, it becomes a "validation" set. The term "test set" is often reserved for the final evaluations of very few models.

The data splitting procedure can be described more formally as follows:

Generic Evaluation of Prediction Rules by Sample-Splitting

1. Randomly partition the data into training and testing samples. Suppose we use n observations for training and m for testing/validation.
2. Use the training sample to compute a prediction rule $\hat{f}(X)$. For example, $\hat{f}(X) = \hat{\beta}'X$ in the linear model.
3. Let $\mathcal{I}$ denote the indexes of the observations in the test sample. Then the out-of-sample/test mean squared error is

$$\text{MSE}_{test} = \frac{1}{m}\sum_{k\in\mathcal{I}}(Y_k - \hat{f}(X_k))^2,$$

and the out-of-sample/test R^2 is

$$R^2_{test} = 1 - \frac{\text{MSE}_{test}}{\frac{1}{m}\sum_{k\in\mathcal{I}} Y_k^2}.$$

In Section 3.B, we consider a more data-efficient evaluation procedure called cross-validation where test data are reused for training. In brief, we split the data into even parts, for each part we repeat the evaluation procedure taking that part to be the "test" sample, and finally we average the values of MSE_{test} that we computed in each round.

There is an important variation on the sample splitting procedure, called *stratified splitting* that provides guarantees that the training and test samples are similar.[6] In large samples, training and test samples will be similar by virtue of the laws of large numbers, but similarity is not guaranteed in moderate-sized samples. For more discussion, please see this blog on Data Splitting [4].

6: For example, we can make sure that the proportions of college-graduates and non-college-graduates are the same in both training and test samples. These issues are important in moderate-sized samples.

1.3 Inference about Predictive Effects or Association

Here we examine inference on *predictive effects*, which describe how our (population best linear) predictions change if the value of a regressor changes by a unit, while the other regressors remain unchanged.

Specifically, we partition the vector of regressors X into two components:

$$X = (D, W')',$$

where D represents the "target" regressor of interest, and W represents the other regressors, sometimes called the controls. We can therefore write

$$Y = \underbrace{\beta_1 D + \beta_2' W}_{\text{predicted value}} + \underbrace{\varepsilon}_{\text{error}}, \tag{1.3.1}$$

and ask the question:

> How does the predicted value of Y change if D increases by a unit while W remains unchanged?

Note that this question is purely about the properties of the prediction rule and generally has nothing to do with causality.

The answer is the predicted value of Y changes by

$$\beta_1.$$

Example 1.3.1 (Wage Differences) In the analysis of wages, which we will discuss later in more detail, an interesting

question can be formulated as:

- "What is the difference in predicted wages between female and non-female workers with the same job-relevant characteristics?"

Let D represent the female indicator and W represent experience, educational, occupational, and geographic characteristics. The answer to the question is then the population regression coefficient

$$\beta_1$$

corresponding to D.

Understanding β_1 via "Partialling-Out"

"Partialling-out" is an important tool that provides conceptual understanding of the regression coefficient β_1.

In the *population*, we define the partialling-out operation as a procedure that takes a random variable V and creates the "residualized" error variable $\tilde{V}$ by subtracting the part of V that is linearly predicted by W:

$$\tilde{V} = V - \gamma_{VW}' W, \quad \gamma_{VW} \in \arg\min_{\gamma} \mathrm{E}\left[(V - \gamma' W)^2\right].$$

When V is a vector, we apply the operation to each component. It can be shown that the partialling-out operation is linear in the sense that[7]

$$Y = \nu V + \mu U \implies \tilde{Y} = \nu \tilde{V} + \mu \tilde{U}.$$

Formally, this operation is well defined on the space of random variables with finite second moments.

We apply the partialling-out operation to both sides of our regression equation $Y = \beta_1 D + \beta_2' W + \varepsilon$ to get

$$\tilde{Y} = \beta_1 \tilde{D} + \beta_2' \tilde{W} + \tilde{\varepsilon},$$

which simplifies to the decomposition:

$$\tilde{Y} = \beta_1 \tilde{D} + \varepsilon, \quad \mathrm{E}\left[\varepsilon \tilde{D}\right] = 0. \tag{1.3.2}$$

Decomposition (1.3.2) follows because partialling-out eliminates $\beta_2' W$, since $\tilde{W} = 0$, and leaves ε untouched, $\tilde{\varepsilon} = \varepsilon$, since ε is linearly unpredictable by X and therefore by W. Moreover, $\mathrm{E}[\varepsilon \tilde{D}] = 0$ since $\tilde{D}$ is a linear function of $X = (D, W')'$ and ε is orthogonal to X and therefore to any linear function of X.

7: Verify this as a reading exercise. Use the definition of the BLP decompositions of U and V with respect to regressors W, to derive a BLP decomposition of Y with respect to W.

The decomposition (1.3.2) implies that $\mathrm{E}\varepsilon\tilde{D} = 0$ are the Normal Equations for the population regression of $\tilde{Y}$ on $\tilde{D}$. Therefore, we just rediscovered the following result.

Theorem 1.3.1 (Frisch-Waugh-Lovell, FWL [5],[6],[7]) *The population linear regression coefficient β_1 can be recovered from the population linear regression of $\tilde{Y}$ on $\tilde{D}$:*

$$\beta_1 = \arg\min_{b_1} \mathrm{E}[(\tilde{Y} - b_1\tilde{D})^2] = (\mathrm{E}[\tilde{D}^2])^{-1}\mathrm{E}[\tilde{D}\tilde{Y}],$$

where we assume D cannot be perfectly predicted by W, i.e., $\mathrm{E}[\tilde{D}^2] > 0$, so β_1 is uniquely defined.

In other words, β_1 can be interpreted as a (univariate) linear regression coefficient in the linear regression of *residualized* Y on *residualized* D, where the residuals are defined by partialling-out the linear effect of W from Y and D.

Technically, these are regression *errors*, not residuals, as we are here working with the population, whereas residuals refer to errors to the sample regression fit. However, we will not adhere strictly to this distinction as it will be convenient to apply analogous logic to partialling-out in the population and the sample.

When we work with the *sample*, we simply mimic the partialling-out operation in the population in the sample. In what follows, we assume p/n is small, so sample linear regression provides high-quality partialling-out. By the FWL Theorem applied to the sample instead of in the population, the sample linear regression of Y on D and W gives us the estimator $\hat{\beta}_1$ which is identical to the estimator obtained via sample partialling-out.

It is useful to give the formula for $\hat{\beta}_1$ in terms of sample partialling-out:

$$\hat{\beta}_1 = \arg\min_{b} \mathbb{E}_n[(\check{Y} - b\check{D})^2] = (\mathbb{E}_n[\check{D}^2])^{-1}\mathbb{E}_n[\check{D}\check{Y}], \quad (1.3.3)$$

where $\check{V}_i$ is the residual left after predicting V_i with controls W_i in the sample and we assume $\mathbb{E}_n[\check{D}^2] > 0$. That is,

$$\check{V}_i = V_i - \hat{\gamma}_{VW}'W_i, \quad \hat{\gamma}_{VW} \in \arg\min_{\gamma} \mathbb{E}_n[(V - \gamma'W)^2].$$

From Theorem 1.2.1, we know that using sample linear regression for partialling-out will provide high-quality estimates of the residuals when p/n is small. When p/n is not small, using sample linear regression for partialling-out won't be such a good idea and an alternative is to use penalized regression or dimension reduction. We will cover this in Chapter 3, but we can definitely try it out in the empirical example that concludes this chapter before we even attempt to understand it.

Why not?

Adaptive Inference

We next consider the large sample properties of the estimator $\hat{\beta}_1$.

Theorem 1.3.2 (Adaptive Inference) *Under regularity conditions and if $p/n \approx 0$, the estimation error in $\check{D}_i$ and $\check{Y}_i$ has no first order effect on the stochastic behavior of $\hat{\beta}_1$. Namely,*

$$\sqrt{n}(\hat{\beta}_1 - \beta_1) \approx \sqrt{n}\mathbb{E}_n[\tilde{D}\varepsilon]/\mathbb{E}_n[\tilde{D}^2] \tag{1.3.4}$$

and consequently,

$$\sqrt{n}(\hat{\beta}_1 - \beta_1) \overset{a}{\sim} N(0, \mathsf{V})$$

where

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

We can equivalently write

$$\hat{\beta}_1 \overset{a}{\sim} N(\beta_1, \mathsf{V}/n).$$

That is, $\hat{\beta}_1$ is approximately normally distributed with mean β_1 and variance V/n. Thus, $\hat{\beta}_1$ concentrates in a $\sqrt{\mathsf{V}/n}$- neighborhood of β_1 with deviations controlled by the normal law.

The first result in Theorem 1.3.2, equation (1.3.4), states the estimator minus the estimand is an approximate centered average. The remaining properties stated in the theorem then follow from the central limit theorem.

The *adaptivity* refers to the fact that estimation of residuals $\check{D}$ has a negligible impact on the large sample behavior of the OLS estimator – the approximate behavior is the same as if we had used true residuals $\tilde{D}$ instead. This adaptivity property will be derived later as a consequence of a more general phenomenon which we shall call *Neyman orthogonality*.[8]

The estimated standard error of $\hat{\beta}_1$ is $\sqrt{\hat{\mathsf{V}}/n}$, where $\hat{\mathsf{V}}$ is any estimator of V based on the plug-in principle such that $\hat{\mathsf{V}} \approx V$. The standard estimator for independent data is called the Eicker-Huber-White robust variance estimator ([8], [9],[10], [11]):

$$\hat{\mathsf{V}} = (\mathbb{E}_n[\check{D}^2])^{-1}\mathbb{E}_n[\check{D}^2\hat{\varepsilon}^2](\mathbb{E}_n[\check{D}^2])^{-1}.$$

This standard error estimator formally works when $p/n \approx 0$, but fails in settings where p/n is not small; see, e.g., [12].

The notation $A_n \overset{a}{\sim} N(0, \mathsf{V})$ reads as A_n is approximately distributed as $N(0, \mathsf{V})$. Approximate distribution formally means that $\sup_{R \in \mathscr{R}} |\mathrm{P}(A_n \in R) - \mathrm{P}(N(0, \mathsf{V}) \in R)| \approx 0$, where $\mathscr{R}$ is the collection of rectangular sets (intervals for the case of A_n being a scalar random variable).

8: We'll defer the formal defintion of Neyman orthogonality for a bit. See Section 4.3.

Consider the set, called the $(1-\mathsf{a})\%$ *confidence interval*,

$$[\hat{l}, \hat{u}] := \left[\hat{\beta}_1 - z_{1-\mathsf{a}/2}\sqrt{\hat{\mathsf{V}}/n}, \hat{\beta}_1 + z_{1-\mathsf{a}/2}\sqrt{\hat{\mathsf{V}}/n}\right],$$

where $z_{1-\mathsf{a}/2}$ denotes the $(1-\mathsf{a}/2)-$quantile of the standard normal distribution. For example, the 95% confidence interval is given by

$$\left[\hat{\beta}_1 - 1.96\sqrt{\hat{\mathsf{V}}/n}, \hat{\beta}_1 + 1.96\sqrt{\hat{\mathsf{V}}/n}\right].$$

If we were to envision drawing samples of size n repeatedly from the same population, a $(1-\mathsf{a})\times 100\%$ confidence interval would contain β_1 in approximately $(1-\mathsf{a})\times 100\%$ of the drawn samples:

$$\mathrm{P}(\beta_1 \in [\hat{l}, \hat{u}]) \approx 1-\mathsf{a}.$$

In other words, aside from "atypical" samples, which occur with probability smaller than $\approx \mathsf{a}$, the confidence interval contains the population value of the best linear predictor coefficient β_1. Note that what is random in the coverage event "$\beta_1 \in [\hat{l}, \hat{u}]$" is the confidence interval $[\hat{l}, \hat{u}]$, which depends on the specific sample. The population quantity β_1 is fixed over draws of samples since the population is unchanged.

An alternative is to use $\tilde{\mathsf{V}} = \frac{n}{n-p}(\mathbb{E}_n[\check{D}^2])^{-1}\mathbb{E}_n[\widehat{\varepsilon}^2]$ instead of $\hat{V}$ and the $(1-\mathsf{a}/2)-$quantile of the Student's t-distribution with $n-p$ degrees of freedom instead of $z_{1-\mathsf{a}/2}$. Under these choices, $\mathrm{P}(\beta_1 \in [\hat{l}, \hat{u}]) = 1-\mathsf{a}$ if $\varepsilon \perp\!\!\!\perp X$ and ε is normal. Normality here is only required for exact coverage. However, coverage may fail to be even approximately $1-\mathsf{a}$ when $\tilde{\mathsf{V}}$ is used and $\varepsilon \perp\!\!\!\perp X$ does not hold. We thus prefer the *robust* variance estimator $\hat{\mathsf{V}}$ because it ensures $\mathrm{P}(\beta_1 \in [\hat{l}, \hat{u}]) \approx 1-\mathsf{a}$ without relying on $\varepsilon \perp\!\!\!\perp X$, known as homoskedasticity. We do sometimes use Student's t quantiles because they converge to standard normal quantiles from above as $n-p$ grows and thus maintain approximate confidence interval coverage.

1.4 Application: Wage Prediction and Gaps

In labor economics, an important question is what determines the wage of workers. Interest in this question goes back to the work of Jacob Mincer (see [13]). While determining the factors that lead to a worker's wage is a causal question, we can begin to investigate it from a predictive perspective. We aim to answer two main questions:

- The Prediction Question: How can we use job-relevant characteristics, such as education and experience, to best predict wages?
- The Predictive Effect or Association Question: What is the difference in predicted wages between male and female workers with the same job-relevant characteristics?

We illustrate using data from the 2015 March Supplement of the U.S. Current Population Survey (CPS 2015). As outcome, Y, we use the log hourly wage, and we let X denote various

We here use the binary distinction between "male" and "female" workers only because it is thus recorded as a binary variable in the CPS 2015 data, taking only these two values and denoted by "sex." It is, nonetheless, self-reported. We will investigate differences in wages in the two groups defined by this variable. This may be thought to correspond to what is often referred to as a "gender wage gap."

This example also serves as an important reminder that data is simply speech organized into tables and, as such, can encode specific worldviews, subjective definitions, and biases, even when reflecting external observations. Data, including variable names and variable values, should therefore not be taken as objective truth simply because of its dry, tidy form and should instead be understood critically within the context of its collection.

characteristics of workers. We focus on a (sub) sample of single (never married) workers, which is of size $n = 5,150$. Table 1.1 provides mean characteristics of some key variables.

Table 1.1: Descriptive statistics for sample of never married workers.

	Sample Mean
Log Wage	2.97
Female	0.44
Some High School	0.02
High School Graduate	0.24
Some College	0.28
College Graduate	0.32
Advanced Degree	0.14
Experience	13.76

We will estimate a linear predictive (regression) model for log hourly wage using job-relevant characteristics

$$Y = \beta' X + \varepsilon, \quad \varepsilon \perp X,$$

assess the quality of the empirical prediction rule $\hat{\beta}' X$ using out-of-sample prediction performance, and analyze if there is a gap (i.e., difference) in pay for male and female workers (i.e. analyze the so-called "*gender wage gap*"). Any such gap may partly reflect discrimination in the labor market. We will discuss the potential to learn about discrimination in more detail in Chapter 6.

Prediction of Wages

Predicting Wages R Notebook and Predicting Wages Python Notebook contain the predictive exercise for wages.

Our goal here is to predict (log) wages using various characteristics of workers, and assess the predictive performance of two linear models using adjusted MSE and R^2 and out-of-sample MSE and R^2.

We employ three different specifications for prediction:

- In the **Basic Model** X consists of a set of raw regressors (e.g. sex, experience, education indicators, occupation and industry indicators, and regional indicators), for a total of $p = 51$ regressors. Our basic specification is inspired by the famous Mincer equation from labor economics; see, e.g., [13] for a review.

- In the **Flexible Model**, X consists of all raw regressors from the basic model as well as *technical regressors*, which

are transformations of the raw regressors, namely, polynomials in experience (experience^2, experience^3, and experience^4) and additional two-way interactions of the polynomials in experience with all other raw regressors except for sex. An example of a regressor created through a two-way interaction is *experience* times the indicator of having a *college degree*. In total, we have $p = 246$ regressors.

	p	R^2_{sample}	MSE_{sample}	R^2_{adj}	MSE_{adj}
basic	51	0.30	0.23	0.30	0.23
flexible	246	0.35	0.22	0.31	0.23
flexible Lasso	246	0.32	0.23	0.31	0.23

Table 1.2: Assessment of predictive performance with in-sample R^2 and MSE.

To enable both in- and out-of-sample performance evaluation. We start by randomly selecting 80% of the observations as the training sample and keep the other 20% for use as a test sample.

Table 1.2 shows measures of predictive performance in the training data. That is, the table reports predictive performance on the same data that were used to estimate the model parameters. The flexible regression model performs slightly better than the basic model (higher R^2_{adj} and lower MSE_{adj}). Note also that the discrepancy between the unadjusted and adjusted measures is not large, which is expected given that

$$p/n \text{ is small.}$$

We report results for evaluating the prediction rules in the test data in Table 1.3. That is, the table reports predictive performance on *new* data that were not used to estimate the models.

	MSE_{test}	R^2_{test}
basic	0.197	0.328
flexible	0.206	0.296
flexible Lasso	0.200	0.317

Table 1.3: Assessment of predictive performance on a 20% validation sample.

Based on this exercise, it appears that the basic regression model works slightly better than the flexible regression at predicting log wages for new observations. That is, we see that the test (out-of-sample) MSE and R^2 for the basic regression model are respectively slightly lower and higher than those of the

flexible regression model, indicating slightly superior out-of-sample predictive performance. This behavior is different from that obtained when looking at the within sample fit statistics reported in Table 1.2.

Tables 1.2 and 1.3 also provides the test MSE of the flexible model that has been estimated via Lasso regression. Lasso (*least absolute shrinkage and selection operator*) is a penalized regression method that can be used to reduce the complexity of a regression model when the ratio p/n is not small. We introduce this method in Chapter 3, but this does not prevent us from trying it here even though it may appear as a black box at this point. The out-of-sample MSE can be computed for any other black-box prediction method as well. In this example, this method performs similarly to the basic and flexible regression models estimated using OLS. This finding is not surprising given the modest dimensionality and similarity between the performance of the two OLS-estimated models.

Finally, to highlight the potential of estimating the linear model via OLS to overfit, we consider one more model.

- In the **Extra Flexible Model**, X consists sex and all two way interactions between experience, experience^2, experience^3, experience^4, and all other raw regressors except for sex. In total, we have $p = 979$ regressors in this specification.

	OLS	Lasso
MSE_{sample}	0.178	0.210
MSE_{adj}	0.235	0.223
MSE_{test}	0.250	0.199
R^2_{sample}	0.467	0.368
R^2_{adj}	0.345	0.331
R^2_{test}	0.148	0.322

Table 1.4: Assessment of predictive performance in the extra flexible model with $p = 979$ regressors.

We report measures of predictive performance in the training and test data from OLS and Lasso estimates of our "extra flexible" model in Table 1.4. Here, we see that the model estimated by OLS appears to be overfitting. The in-sample statistics substantially overstate predictive performance relative to the performance we see in the test data. For example, the R^2 and adjusted R^2 in the training data are 0.467 and 0.345, both of which substantially overstate the R^2 obtained in the test data, 0.148. We also see that the performance on the test data for the extra flexible model is substantially worse than the performance of the much simpler

basic and flexible models. That is, it looks like the OLS estimates of the extra flexible model have specialized to fitting aspects of the training data that do not generalize to the test data and lead to a deterioration in predictive performance relative to the simpler models.

The performance of the Lasso contrasts sharply with this behavior. We see that the in-sample and out-of-sample predictive performance measures for the Lasso based estimates of the extra flexible model are similar to each other. They are also similar to the performance of the simpler models. It seems that Lasso is finding a competitive predictive model without overfitting even in the extra flexible model. We will return to this behavior in Chapter 3 where we will show that Lasso and related methods are able to find good prediction rules in even extremely high-dimensional settings, where for example $p \gg n$, where OLS breaks down theoretically and in practice.

Wage Gap

Wage Gaps R Notebook and Wage Gaps Python Notebook contain the code for this section.

An important question is whether there is a gap (i.e., difference) in predicted wages between male and female workers with the same job-relevant characteristics. To answer this question, we estimate the log-linear regression model:

$$Y = \beta_1 D + \beta_2' W + \varepsilon, \tag{1.4.1}$$

where Y is log-wage, D is the indicator of being female (1 if female and 0 otherwise) and the W's are other determinants of wages. W includes education, polynomials in experience, region, and occupation and industry indicators plus all two-way interactions of polynomial in experience with region, occupation, and industry indicators.

Table 1.5: Empirical means for the groups defined by the sex variable for never-married workers.

	All	Male	Female
Log Wage	2.9708	2.9878	2.9495
Less then High School	0.0233	0.0318	0.0127
High School Graduate	0.2439	0.2943	0.1809
Some College	0.2781	0.2733	0.2840
College Graduate	0.3177	0.2940	0.3473
Advanced Degree	0.1371	0.1066	0.1752
Experience	13.7606	13.7840	13.7313

As we have log-transformed wages, we are analyzing the relative difference in pay for male and female workers. Table 1.5 tabulates

mean characteristics given sex. It shows that the difference in average log-wage between never married male and never married female workers is equal to 0.038 with male workers earning more. Thus, in this group, male average wage is about 3.8% higher than female average wage.[9] We also observe that never married female workers are relatively more educated than never married male workers.

9: This interpretation relies on the approximation $log(a) - log(b) \approx (a-b)/b$, which is accurate whenever $(a-b)/b$ is small and $b > 0$.

Table 1.6 summarizes the regression results. Overall, we see that the unconditional wage gap of size 3.8% for female workers increases to about 7% after controlling for worker characteristics. This means we would predict a female worker's wage to be about 7% less per hour on average than the wage of a male worker who had the same experience, education, geographical region and occupation.

The partialling-out approach provides a numerically identical estimate for the coefficient β_1 ($\beta_1 \approx 7\%$), numerically confirming the FWL theorem. Using Lasso for partialling-out (*p-out w/Lasso*) gives similar results to using OLS. This similarity is expected here, since

$$p/n \text{ is small,}$$

and partialling out by least squares should work well.

Table 1.6: Estimated conditional wage gaps for never married workers.

	Estimate	Std. Error
reg without controls	−0.038	0.016
reg with controls	−0.070	0.015
partial out reg w/ controls	−0.070	0.015
Double Lasso (p-out w/ Lasso)	−0.072	0.015

To sum up, our estimate of the conditional wage gap for never-married workers using OLS is about −7% and the 95% confidence interval is about [−10%, −4%].

One way to understand the estimate with controls (−0.070) is as the part of the total gap (−0.038) that cannot be explained by differences in group characteristics. Namely, take Eq. (1.4.1) and average it in the male and female groups to obtain:

$\mathbb{E}_n[\cdot \mid D = d]$ abbreviates $\mathbb{E}_n$ for the subsample of the data where $D = d$, for $d = 0, 1$.

$$\underbrace{\mathbb{E}_n[Y \mid D=1] - \mathbb{E}_n[Y \mid D=0]}_{-0.038} = \underbrace{\hat{\beta}_1}_{-0.070} + \underbrace{\hat{\beta}_2'(\mathbb{E}_n[W \mid D=1] - \mathbb{E}_n[W \mid D=0])}_{0.031}.$$

Here, 0.031 is the difference in log wages we predict based on differences in characteristics. That is, based on observed characteristics W and slopes $\hat{\beta}_2$, we would predict a higher average log wage for female workers than for male workers. This positive difference based on characteristics is counteracted by a negative difference of -0.070 that is *unexplained* by the characteristics and attributable to the difference in the sex variable alone, holding characteristics fixed.

Note these numerical values are rounded so the numbers under the braces across the equation above do not exactly add up.

One missing part in this interpretation is that the model Eq. (1.4.1) does not consider the possible interaction of sex and the characteristics in the prediction of log wage. We can augment Eq. (1.4.1) to account for this, resulting in the interactive log-linear regression model:

$$Y = \beta_1 D + \beta_2' W + \beta_3' W D + \varepsilon.$$

Fitting this new model provides an alternative decomposition:

Such a decomposition that accounts for an *interaction* term is known as a *Oaxaca-Blinder decomposition* introduced in [14] and [15].

$$\underbrace{\mathbb{E}_n[Y \mid D=1] - \mathbb{E}_n[Y \mid D=0]}_{-0.038} = \underbrace{\hat{\beta}_1}_{-2.320} + \underbrace{\hat{\beta}_2'(\mathbb{E}_n[W \mid D=1] - \mathbb{E}_n[W \mid D=0])}_{0.002} + \underbrace{\hat{\beta}_3' \mathbb{E}_n[W \mid D=1]}_{2.280}.$$

Here, 0.002 is the difference attributed to differences in group characteristics. Next, 2.280 is the difference attributed to the different predictive effect of the characteristics in the two groups captured by the coefficients on the interaction terms, β_3. The difference in predictive effects were previously not considered in the model without interactions. Finally, -2.320 is the remaining difference that remains unexplained by either difference in characteristics *or* their different predictive effect in the two groups.

In order to wrap up and provide a stylized illustration of the impact of dimensionality p on inference, we revisit the extra-flexible model from the previous section which used $p = 979$ controls within a subset of $n = 1000$ of the original observations. This setting gives us $p/n \approx 1$, so the usual theory for estimating linear model coefficients by OLS no longer applies. [16] provide more refined results for OLS estimates of regression coefficients in the case $p/n \to C < 1$. They find that OLS estimates of single coefficients can be consistent in this regime and provide an estimator of the asymptotic variance that is consistent when $p/n < 1/2$ as long as additional regularity conditions hold. They also find that the usual Eicker-Huber-White robust variance estimator is not consistent in this regime but that the jackknife

variance estimator, while not consistent, is conservative.

We report estimates of the conditional wage gap within this setup in Table 1.7. We report point estimates from OLS applied to the full set of variables and provide both the Eicker-Huber-White standard error (HC0) and the jackknife standard error (HC3).[10] These are provided for illustration, but we note that HC0 is known to be inconsistent and to behave very poorly, in the sense of generally being far too small, in the high-dimensional setting. HC3 is more reliable, but one should also be skeptical given that $p/n \approx 1$ in this example. Finally, we also report point estimate and standard error for the Double Lasso procedure which is consistent, asymptotically, normal and has estimable standard errors under structure outlined in Chapter 4 even when $p \gg n$. For now, we can think of it is a point of comparison.

10: The Eicker-Huber-White variance estimator is often referred to as "HC0" and the jackknife as "HC3."

	Estimate	HC0	HC3
regression	-0.067	0.039	0.073
Double Lasso (p-out w/ Lasso)	-0.054	0.034	0.034

Table 1.7: The estimated conditional wage gaps for never married workers with approximately 1000 controls in a sample of 1000 observations.

Comparing to the case with the full data set, we see that point estimates are not wildly different but that standard errors are larger. Part of the standard error difference is predicted simply by the difference in sample sizes. Specifically, $\sqrt{5150/1000} \approx 2.27$, so we would expect standard errors to be 2.27 times bigger with $n = 1000$ observations than with $n = 5150$. This inflation holds almost exactly for the Double Lasso.

More interestingly, now that $p/n \not\approx 0$, we start seeing substantial differences in standard errors between unregularized partialling out (regression) and partialling out with Lasso (aka Double Lasso). While we don't want to take the OLS standard errors too seriously – we know the Huber-Eicker-White standard error does not work in this setting and are also suspect of the jackknife here – the comparison between the OLS and Double Lasso standard errors and comparison to the full sample results are revealing. Compared to the full sample results, the jackknife standard error (HC3) is much larger than would be expected simply due to the decrease in the sample size in this example. The difference from this expectation (partially) reflects the impact of dimensionality on the OLS estimate of the regression coefficient. The Double Lasso seems to be roughly insensitive to the dimensionality of the control variables and scales exactly as one would expect given the difference in sample size.

The punchline of this final example is that OLS is no longer adaptive in the "p/n not small" regime. The lack of adaptivity

means that conventional properties of OLS may not hold and that other procedures may be highly preferable to OLS.

Notebooks

- Predicting Wages R Notebook and Predicting Wages Python Notebook contain a simple predictive exercise for wages. We will return to this dataset and prediction problem repeatedly in future chapters, re-estimating it using a broad range of ML estimators and providing a means of comparing their performance.

- Wage Gaps R Notebook and Wage Gaps Python Notebook contain a simple analysis of wage gaps.

- The Linear Model Overfitting R Notebook and the Linear Model Overfitting Python Notebook contain a set of simple simulations that show how measures of fit perform in a high p/n setting.

1.5 Notes

Least squares were invented by Legendre ([17]) and Gauss ([18]) around 1800. Frisch, Waugh, and Lovell ([5],[6],[7]) discovered the partialling-out interpretation of the least squares coefficients in the 1930s. The asymptotic theory mentioned in the note is more recent and has been developed since early work of Huber in the 70s on m-estimators (estimators that minimize objective functions that correspond empirical averages of losses) under moderately high dimensions; see e.g. [19] and the textbook [20].

For a good, concise treatment of classical least squares, see for example, Chapter 1 in Amemiya's classical graduate econometrics text [3]; Bruce Hansen's new textbook [21] is an excellent up-to-date reference.

Regularity conditions under which Theorem 1.2.1 and Theorem 1.3.2 hold under $p \to \infty$ and $p/n \to 0$ asymptotics can be found in [22] and [16]. The results of the latter reference allow for $p/n \to c < 1$, which introduces an additional asymptotic variance term when $c > 0$; the case with $c = 0$ recovers Theorem 1.3.2. See also review [23] for some recent understanding of properties of least squares estimators.

Study Questions

Modern notebooks, including Jupyter Notebooks, R Markdown, and Quarto offer a simple way to integrate code cells and explanations (text and formulas) in a single notebook. This allows the user to execute code in discretized chunks for clarity and ease of debugging as well as to better provide commentary on what the code is doing. See the Notebooks section above for examples.

1. Write a notebook (R, Python, etc.) where you briefly explain the idea of sample splitting to evaluate the performance of prediction rules to a fellow student, and show how to use it on the wage data. The explanation should be clear and concise (one paragraph suffices) so that a fellow student can understand. You can take our notebooks as a starting point, but provide a bit more explanation and modify them by exploring different specifications of the models (or looking at an interesting subset of the data or even other data – for example, the data you use for your research or thesis work).

2. Write a notebook (R, Python, etc), where you carry out a wage gap analysis, focusing on the subset of college-educated workers. The analysis should be analogous to what we've presented – explaining "partialling out," generating point estimates and standard errors – but don't hesitate to experiment and explain more. Exploring other data-sets or similar questions, e.g. wage gaps by race, is always welcome.

3. The half-serious link to Pythagoras was serious in its half. Consider sample linear regression with $n = 2$ and just one regressor, so that $Y_i = \hat{\beta} X_i + \hat{\varepsilon}_i$ for $i = 1, 2$, where $\hat{\beta}$ is the ordinary least squares estimator, a scalar quantity in this case. Let $\mathsf{Y} = (Y_1, Y_2)'$, $\mathsf{X} = (X_1, X_2)'$, $\hat{\varepsilon} = (\hat{\varepsilon}_1, \hat{\varepsilon}_2)'$, and let $\hat{\mathsf{Y}} = \hat{\beta}\mathsf{X}$. Find the connection between the decomposition $\mathsf{Y}'\mathsf{Y}/n = \hat{\mathsf{Y}}'\hat{\mathsf{Y}}/n + \hat{\varepsilon}'\hat{\varepsilon}/n$ and the Pythagorean theorem. Find the geometric interpretation for $\hat{Y}$, and write the explicit formula for $\hat{\beta}$ in this case. If you get stuck, google the "geometric interpretation of least squares."

1.A Central Limit Theorem

Univariate

Consider the scaled sum $W = \sum_{i=1}^{n} X_i/\sqrt{n}$ of independent and identically distributed variables X_i such that $\mathrm{E}[X] = 0$ and $\mathrm{Var}(X) = 1$. The classical CLT states that W is approximately Gaussian provided that none of the summands are too large,

namely

$$\sup_{x\in\mathbb{R}} |\mathrm{P}(W \leq x) - \mathrm{P}(N(0,1) \leq x)| \approx 0.$$

This result is reassuring, but the theorem does not inform us how small the error is in a given setting.

The Berry-Esseen theorem provides a quantitative characterization of the error.

Theorem 1.A.1 (Berry-Esseen's Central Limit Theorem)

$$\sup_{x\in\mathbb{R}} |\mathrm{P}(W \leq x) - \mathrm{P}(N(0,1) \leq x)| \leq K\mathrm{E}[|X|^3]/\sqrt{n},$$

for a numerical constant $K < .5$.

The result asserts that the Gaussian approximation error rate declines like $1/\sqrt{n}$. It also states that given n, the approximation quality improves as the third absolute moment $\mathrm{E}[|X|^3]$ decreases. This results gives a good guide regarding when the Gaussian approximation gives accurate results.[11] Of course, one can also check the approximation quality via simulation experiments that mimic the practical situation.

11: Consider, for instance, the case when X_i are centered and standardized Bernoulli random variables with success probability p, i.e., $X_i = \frac{Z_i - p}{\sqrt{p(1-p)}}$ and Z_i is Bernoulli with success probability p. The error in the Berry-Esseen theorem, in this case, becomes $\approx 1/\sqrt{p(1-p)n}$. Thus, the error in the Gaussian approximation is guaranteed to be small by the Berry-Esseen theorem only if $p(1-p)n$ is large. Thus, for extreme probabilities, where either success or failure events are extremely rare for the given sample size, i.e., when $p \cdot n$ or $(1-p) \cdot n$ is small, the use of the Gaussian approximation is not advisable.

Multivariate

Later in the book, we will use multivariate central limit theorems as well. To this end, we are going to state the following more general result due to [24], which refines earlier results by [25] and [26].

Let $\mathcal{I}$ be a countable set (either finite or infinite) and let $X_i, i \in \mathcal{I}$, be independent $\mathbb{R}^d$-valued random vectors. Assume that $\mathrm{E}[X_i] = 0$ for all i and that $\sum_{i\in\mathcal{I}} \mathrm{Var}(X_i) = I_d$. It is well known that in this case, the sum $W := \sum_{i\in\mathcal{I}} X_i$ exists almost surely and that $\mathrm{E}W = 0$ and $\mathrm{Var}(W) = I_d$.

Theorem 1.A.2 (Multivariate CLT; [24]) *For* X_i *and* W *as above and all measurable convex sets* $A \subseteq \mathbb{R}^d$*, we have*

$$|\mathrm{P}(W \in A) - \mathrm{P}(N(0, I_d) \in A)| \leq \left(42d^{1/4} + 16\right) \sum_{i\in I} \mathrm{E}\left[\|X_i\|^3\right].$$

Bibliography

[1] Cambridge Dictionary, *Infer* (cited on page 12).

[2] Alexandre B. Tsybakov. *Introduction to nonparametric estimation*. Springer, 2009 (cited on page 17).

[3] Takeshi Amemiya. *Advanced Econometrics*. Cambridge, MA: Harvard University Press, 1985 (cited on pages 21, 35).

[4] *Data Splitting | R-bloggers*. https://www.r-bloggers.com/2016/08/data-splitting/, Accessed: 2022-25-02 (cited on page 23).

[5] Ragnar Frisch and Frederick V Waugh. 'Partial time regressions as compared with individual trends'. In: *Econometrica* (1933), pp. 387–401 (cited on pages 25, 35).

[6] Michael C Lovell. 'Seasonal adjustment of economic time series and multiple regression analysis'. In: *Journal of the American Statistical Association* 58.304 (1963), pp. 993–1010 (cited on pages 25, 35).

[7] Michael C Lovell. 'A simple proof of the FWL theorem'. In: *Journal of Economic Education* 39.1 (2008), pp. 88–91 (cited on pages 25, 35).

[8] Friedhelm Eicker. 'Limit theorems for regressions with unequal and dependent errors'. In: ed. by Lucien M. Le Cam and Jerzy Neyman. 1967 (cited on page 26).

[9] Peter J. Huber. *The behavior of maximum likelihood estimates under nonstandard conditions*. English. Proc. 5th Berkeley Symp. Math. Stat. Probab., Univ. Calif. 1965/66, 1, 221-233 (1967). 1967 (cited on page 26).

[10] Halbert White. 'A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity'. In: *Econometrica* (1980), pp. 817–838 (cited on page 26).

[11] Friedhelm Eicker. 'Asymptotic normality and consistency of the least squares estimators for families of linear regressions'. In: *Annals of Mathematical Statistics* 34.2 (1963), pp. 447–456 (cited on page 26).

[12] Matias Cattaneo, Michael Jansson, and Whitney Newey. 'Alternative Asymptotics and the Partially Linear Model with Many Regressors'. In: *Working Paper, http://econ-www.mit.edu/files/6204* (2010) (cited on page 26).

[13] Thomas Lemieux. 'The "Mincer equation" thirty years after schooling, experience, and earnings'. In: *Jacob Mincer a Pioneer of Modern Labor Economics*. Springer, 2006, pp. 127–145 (cited on pages 27, 28).

[14] Ronald Oaxaca. 'Male-Female Wage Differentials in Urban Labor Markets'. In: *International Economic Review* 14.3 (1973), pp. 693–709. (Visited on 10/10/2023) (cited on page 33).

[15] Alan S. Blinder. 'Wage Discrimination: Reduced Form and Structural Estimates'. In: *Journal of Human Resources* 8.4 (1973), pp. 436–455. (Visited on 10/10/2023) (cited on page 33).

[16] Matias D. Cattaneo, Michael Jansson, and Whitney K. Newey. 'Inference in linear regression models with many covariates and heteroscedasticity'. In: *Journal of the American Statistical Association* 113.523 (2018), pp. 1350–1361 (cited on pages 33, 35).

[17] Adrien Marie Legendre. *Nouvelles méthodes pour la détermination des orbites des comètes: avec un supplément contenant divers perfectionnemens de ces méthodes et leur application aux deux comètes de 1805*. Courcier, 1806 (cited on page 35).

[18] Carl-Friedrich Gauss. *Theoria combinationis observationum erroribus minimis obnoxiae*. Henricus Dieterich, 1823 (cited on page 35).

[19] Peter J Huber. 'Robust regression: asymptotics, conjectures and Monte Carlo'. In: *Annals of Statistics* (1973), pp. 799–821 (cited on page 35).

[20] Elvezio M Ronchetti and Peter J Huber. *Robust Statistics*. John Wiley & Sons Hoboken, NJ, USA, 2009 (cited on page 35).

[21] Bruce E. Hansen. *Econometrics*. Princeton University Press, 2022 (cited on page 35).

[22] Alexandre Belloni, Victor Chernozhukov, Denis Chetverikov, and Kengo Kato. 'Some New Asymptotic Theory for Least Squares Series: Pointwise and Uniform Results'. In: *Journal of Econometrics* 186.2 (2015), pp. 345–366 (cited on page 35).

[23] Arun K. Kuchibhotla, Lawrence D. Brown, and Andreas Buja. 'Model-free study of ordinary least squares linear regression'. In: *arXiv preprint arXiv:1809.10538* (2018) (cited on page 35).

[24] Martin Raič. 'A multivariate Berry–Esseen theorem with explicit constants'. In: *Bernoulli* 25.4A (2019), pp. 2824–2853 (cited on page 37).

[25] Vidmantas Bentkus. 'On the dependence of the Berry–Esseen bound on dimension'. In: *Journal of Statistical Planning and Inference* 113.2 (2003), pp. 385–402 (cited on page 37).

[26] F Goetze. 'On the rate of convergence in the multivariate CLT'. In: *Annals of Probability* (1991), pp. 724–739 (cited on page 37).

Causal Inference via Randomized Experiments

2

> "Let us divide them in halfes, let us cast lots, that one half of them may fall to my share, and the other to yours; I will cure them without bloodletting and sensible evacuation; but do you do as ye know [. . .] we shall see how many Funerals both of us shall have."
>
> – Jan Baptist van Helmont [1].

In this chapter we begin discussion of causal inference by focusing on Randomized Control Trials (RCTs). In a randomized control trial, units are randomly divided into those that receive a treatment and those that receive no treatment. Under randomization and other assumptions, the difference in average outcomes between the treated and untreated groups is an average treatment (causal) effect (ATE). By considering pre-treatment covariates, we can improve the precision of the ATE estimate, explore heterogeneity across subgroups, or both. We describe methods for doing so and apply them to several RCTs. We introduce causal diagrams as a means of visualizing RCTs and their underlying causal assumptions. We conclude by outlining some limitations of RCTs.

2.1 Potential Outcomes Framework and Average Treatment Effects

In this section, we discuss the potential outcomes framework for analyzing causality and treatment effects. It offers an elegant way to formalize counterfactuals as a mathematical concept.

We begin by introducing the two *latent* (unobserved) variables

$$Y(1) \text{ and } Y(0).$$

They represent the potential or counterfactual random outcomes for an observational unit when the unit is subject to treatment (treatment state $d = 1$) or no treatment (control or untreated state $d = 0$) [2]. In an economic context, the treatment might be a training program or a policy intervention, and the outcome might be an individual's wage or employment status. In what follows, it is also useful to introduce the potential response or structural function:

$$d \mapsto Y(d),$$

which maps the potential treatment state $d \in \{0, 1\}$ to the random potential outcome $Y(d)$.

For simplicity, we do not consider multivalued or continuous treatments.

In this formulation, we have dependence of the potential outcome $Y(d)(\omega)$ on the underlying state of the world ω. In our formalization, ω will represent randomness across observational units and from any other sources.[1]

1: Recall that a random variable V is a mapping $\omega \mapsto V(\omega)$ from the underlying state of the world $\omega \in \Omega$ to the real line (or other metric space) such that we can assign a probability law to it.

The quantities $Y(1)$ and $Y(0)$ are "counterfactual" because they can't be simultaneously observed. That is, we generally do not have identical replicas of the observational units that are simultaneously subject to both treatment and control. [3] calls the inability to observe an individual simultaneously under treatment and control "the fundamental problem of causal inference". The inability to observed each individual's treatment and control outcome means that causal inference shares many features with "missing data"problems, see, e.g. [4].

The individual treatment effect is

$$Y(1) - Y(0).$$

2: As an example, we could uncover individual treatment effects if we had identical twins that could be put in treatment and control groups, and we believed that the only difference in outcomes between these twins is induced by treatment – that is, ω only depends on genetic makeup. Such an example seems unrealistic at best.

This effect will vary across individuals as well as with other sources of randomness encoded in ω. As mentioned above, only one of the two terms is actually observed, and hence it is generally infeasible to uncover the individual treatment effect.[2] However, we can hope to estimate averages and the distribution

of $Y(d)$ at the population level to compute quantities such as the average treatment effect (ATE):

$$\delta = \mathrm{E}[Y(1) - Y(0)] = \mathrm{E}[Y(1)] - \mathrm{E}[Y(0)].$$

Let D denote the actual *assigned treatment*, a random variable, which takes a value of 1 if the observational unit participated in the treatment and 0 otherwise.

Assumption 2.1.1 (Consistency) *We observe*

$$Y := Y(D).$$

For example, if treatment ($D = 1$) corresponded to completion of a job training program and control ($D = 0$) corresponded to not completing the program, Assumption 2.1.1 says that the observed wage outcome is equal to $Y(1)$ for a given person if she has completed the program (has $D = 1$) and is equal to $Y(0)$ if this person has not completed the training program (has $D = 0$). Assumption 2.1.1 seems almost tautological, but it importantly rules out hidden variation in treatment. That is, it requires that the treatment and control states are well-defined and clearly aligned with the observed treatment status, D.

Assumption 2.1.2 (No Interference) *Potential outcomes for any observational unit depend only on the treatment status of that unit and not on the treatment status of any other unit.*

Assumption 2.1.2 has implicitly been captured in our definition of potential outcomes, $Y(d)$, which give the outcome of each unit *when the unit* is subject to treatment state d. This formulation rules out scenarios where the treatment given to one unit may impact the outcome of a different unit. Such spillovers could occur, for example, on social networks where treating an individual could impact all of that individual's friends. Some forms of spillovers are readily accommodated by expanding the definition of treatment and correspondingly adjusting definition of potential outcomes,[3] but treating these extensions is beyond the scope of this book.[4]

Assumptions 2.1.1 and 2.1.2 encapsulate what is often referred to as the Stable Unit-Treatment Value Assumption (SUTVA); see, e.g. Imbens and Rubin [10].

The following analytical example may help gain better understanding of the potential outcomes framework.

3: For example, consider a case where each individual has two friends. We could define potential outcomes allowing for spillovers as $Y(d_0, d_1, d_2)$ where d_0 denotes the treatment state of an individual, d_1 denotes the treatment state of the individual's friend 1, and d_2 denotes the treatment state of the individual's friend 2.

4: For further reading we refer, among many others, to [5], [6], [7], [8] and [9].

Example 2.1.1 [Analytical Example] Consider the following model

$$\begin{aligned} Y(1) &:= \theta_1 + \epsilon_1 \\ Y(0) &:= \theta_0 + \epsilon_0 \\ D &:= 1(\nu > 0), \\ Y &:= Y(D), \end{aligned}$$

where θ_0 and θ_1 are constants, and $(\epsilon_0, \epsilon_1, \nu)$ are jointly normal random stochastic disturbances with mean 0 and covariance matrix Σ. Here, ν represents factors that influence selection into the treatment state. In this example $\mathrm{E}[Y(1)] = \theta_1$, $\mathrm{E}[Y(0)] = \theta_0$, and the ATE is $\delta = \theta_1 - \theta_0$. Importantly, only D and Y are observed.

Under Assumption 2.1.1, population data directly provide the conditional averages

$$\mathrm{E}[Y \mid D = d] = \mathrm{E}[Y(d) \mid D = d], \text{ for } d \in \{0, 1\}.$$

The difference of the two averages gives us the average predictive effect (APE) of treatment status on the outcome:

$$\pi = \mathrm{E}[Y \mid D = 1] - \mathrm{E}[Y \mid D = 0].$$

It measures the association of the treatment status with the outcome.

While the APE is identified – meaning computable from the population data – it may seem surprising (or not at all) that the APE in general does not agree with the ATE δ:

$$\delta \neq \pi. \tag{2.1.1}$$

The difference between the APE and ATE is generally said to be due to *selection bias*. The meaning of selection bias is clarified through the following example, and clarified theoretically below.

Example 2.1.2 (Selection Bias in Observational Data) Suppose we want to study the impact of smoking marijuana on life longevity. Suppose that smoking marijuana has no causal effect on life longevity:

$$Y = Y(0) = Y(1),$$

so that

$$\delta = \mathrm{E}[Y(1)] - \mathrm{E}[Y(0)] = 0.$$

However, the observed smoking behavior, D, is not assigned in an experimental study. Suppose that the behavior determining D is associated with poor health choices such as drinking alcohol, which are known to cause shorter life expectancy, so that $\mathrm{E}[Y \mid D = 1] < \mathrm{E}[Y \mid D = 0]$. In this case, we have negative a predictive effect:

$$\pi = \mathrm{E}[Y \mid D = 1] - \mathrm{E}[Y \mid D = 0] < 0 = \delta,$$

which differs from the true causal effect $\delta = 0$.

To sum up, in the smoking example, the chosen "treatment" variable D is potentially negatively associated with the potential health outcome, inducing the selection bias – the difference between the predictive effect and the causal effect.

Example 2.1.3 (Analytical Version of the Smoking Example) To capture dependence between $Y(d)$ and v in the smoking context analytically, we can go back to Example 2.1.1, and make variables ϵ_d and v be negatively associated:

$$\mathrm{E}[\epsilon_d v] < 0.$$

The negative association between the ϵ_d and v then results in the observed smoking status, D, being negatively associated with the potential outcomes $Y(d)$. Specifically, we have

$$\mathrm{E}[Y|D = 1] < \mathrm{E}[Y|D = 0],$$

which can be verified through additional analytical calculations or via simulation experiments (a homework).

It is useful to emphasize the main reason for having selection bias is that

$$\mathrm{E}[Y(d)|D = 1] \neq \mathrm{E}[Y(d)]$$

whenever D is not independent of $Y(d)$. If D and $Y(d)$ were independent,

$$\mathrm{E}[Y(d)|D = 1] = \mathrm{E}[Y(d)]$$

would hold since in this case D is uninformative about the potential outcome and drops out from the conditional expectation.

To sum up, the problem with observational studies like our contrived Example 2.1.2 is that the "treatment" variable D

is determined by individual behaviors which may be linked to potential outcomes. This linkage generates selection bias - the disagreement between APE and ATE. There are many ways of addressing selection bias, one of which is through an experiment, where we randomly assign the treatment to the units.

Random Assignment/Randomized Controlled Trials

A way to clearly remove selection bias is through random assignment of treatment.

Assumption 2.1.3 (Random Assignment/Exogeneity) *Suppose that treatment status is randomly assigned. Namely, D is statistically independent of each potential outcome $Y(d)$ for $d \in \{0,1\}$, which is denoted as*

$$D \perp\!\!\!\perp Y(d)$$

and $0 < \mathrm{P}(D = 1) < 1$.

This assumption states that the treatment assignment mechanism is purely random, and ensures that there are units in treatment and in control.

Example 2.1.4 (Analytical Example Continued) In the analytical example 2.1.1, Assumption 2.1.3 is satisfied if the stochastic shock ν determining D is independent of stochastic shocks ϵ_0 and ϵ_1 determining $Y(1)$ and $Y(0)$, i.e.

$$\nu \perp\!\!\!\perp (\epsilon_0, \epsilon_1).$$

A key result is that selection bias is removed under Assumption 2.1.3 which allows us to learn summaries of causal effects.

Theorem 2.1.1 (Randomization Removes Selection Bias) *Under Assumption 2.1.3, the average outcome in treatment group d recovers the average potential outcome under the treatment status d:*

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

for each $d \in \{0,1\}$. Hence the average predictive effect and average treatment effect coincide:

$$\begin{aligned} \pi &:= \mathrm{E}[Y \mid D = 1] - \mathrm{E}[Y \mid D = 0] \\ &= \mathrm{E}[Y(1)] - \mathrm{E}[Y(0)] =: \delta. \end{aligned}$$

Assumption 2.1.3 is often not plausible for observational data. In a *randomized controlled trial* (RCT)[5] , the aim is to ensure the plausibility of Assumption 2.1.3 by direct random assignment of treatment D. That is, subjects are randomly assigned a treatment state D by the experimenter without regard to any of their characteristics. Because the random assignment of the treatment is unrelated to all subject characteristics by construction, well-executed RCTs guarantee that Assumption 2.1.3 is satisfied. Because of this property, many consider RCTs as the gold standard in causal inference, and RCTs are routinely employed in a variety of important settings.[6] Examples include evaluating the efficacy of medical treatment, vaccinations, training programs, marketing campaigns, and other kinds of interventions.

5: Synonyms are experiments and A/B tests.

6: Of course, RCTs must be correctly done to guarantee Assumption 2.1.3. For example, RCTs where experimental protocols are not followed continue to suffer from selection bias. There are also examples, *quasi-experiments*, where we may believe that Assumption 2.1.3 is plausible that do not correspond to explicit designed experiments.

Example 2.1.5 (No Selection Bias in Experimental Data) Suppose that in the smoking example (Example 2.1.2), we worked with data where smoking or non-smoking was generated by perfectly enforced random assignment. In this case, we would have agreement between average predictive and treatment effects: $\pi = \delta$. While it is difficult to imagine a long-run RCT where study participants could be forced to smoke or not smoke marijuana (we discuss such limitations as well as ethical considerations in Section 2.4), RCTs are routinely employed in a variety of other important settings.

Statistical Inference with Two Sample Means

Inference is based on the independent sample $\{(Y_i, D_i)\}_{i=1}^n$ obtained from an RCT, where index i denotes the observational unit. We assume that each (Y_i, D_i) has the same distribution as (Y, D). Estimation of the two means $\theta_d = \mathrm{E}[Y \mid D = d]$ for $d = 0$ and $d = 1$ can be done by considering two group means

$$\hat{\theta}_d = \frac{\mathbb{E}_n[Y 1(D = d)]}{\mathbb{E}_n[1(D = d)]}.$$

The two means example can also be treated as a special case of linear regression,[7] but we find it instructive to work out the details directly for the two group means. We provide these details in Section 2.A.

7: Indeed, we can regress Y on D and $1 - D$; that is, estimate the model $Y = \theta_1 D + \theta_0(1-D) + U$. We can then apply the inferential machinery developed in the previous chapter.

Under mild regularity conditions, we have that

$$\sqrt{n}\begin{pmatrix} \hat{\theta}_0 - \theta_0 \\ \hat{\theta}_1 - \theta_1 \end{pmatrix} \overset{\text{a}}{\sim} N(0, \mathsf{V}),$$

where

$$\mathsf{V} = \begin{pmatrix} \frac{\text{Var}(Y|1(D=0))}{P(D=0)} & 0 \\ 0 & \frac{\text{Var}(Y|1(D=1))}{P(D=1)} \end{pmatrix}$$

so that $\hat{\delta} = \hat{\theta}_1 - \hat{\theta}_0$ obeys

$$\sqrt{n}(\hat{\delta} - \delta) \overset{\text{a}}{\sim} N(0, \mathsf{V}_{11} + \mathsf{V}_{22}).$$

To use this result in practice, variance components are usually estimated using the *plug-in principle*, which amounts to using the sample analogues of the expressions above.

Sometimes we are interested in relative effectiveness of treatment effects (for example, vaccine efficiency):

$$f(\theta) = (\theta_1 - \theta_0)/\theta_0 = \delta/\theta_0.$$

Relative effectiveness can be estimated by $\hat{\delta}/\hat{\theta}_0 = f(\hat{\theta})$, where $\hat{\theta} = \{\hat{\theta}_d\}_{d\in\{0,1\}}$ and $\theta = \{\theta_d\}_{d\in\{0,1\}}$, with approximate distribution obtained using the *delta method*:

$$\sqrt{n}(f(\hat{\theta}) - f(\theta)) \approx G'\sqrt{n}(\hat{\theta} - \theta) \overset{\text{a}}{\sim} N(0, G'\mathsf{V}G),$$

where $G = \nabla f(\theta)$, $\hat{\theta} = (\hat{\theta}_0, \hat{\theta}_1)'$, $\theta = (\theta_0, \theta_1)$.[8] .

8: The approximation follows from application of the first order Taylor expansion and continuity of the derivative ∇f at θ.

Pfizer/BioNTech Covid Vaccine RCT

Pfizer/BNTX was the first vaccine approved for emergency use in the EU and US to reduce the risk of Covid-19 disease. See the Food and Drug Administration (FDA) briefing for details about the RCT and the summary data. Volunteers were randomly assigned to receive either a treatment (2-dose vaccination) or a placebo, without knowing which they received, and the doctors making the diagnoses did not know whether a given volunteer received a vaccination or not. In other words, the trial was a double-blind randomized control trial. The results of the study are presented in the following table.

Figure 2.1: Tozinameran (Pfizer-BioNTech Covid-19 vaccine); Image Source: Wikipedia / Arne Müseler

Vaccination RCT R Notebook and Vaccination RCT Python Notebook contain the analysis of the Pfizer-BioNTech Covid-19 Vaccine RCTs.

Efficacy Endpoint Subgroup	BNT162b2 N[a]=19965 Cases n1[b] Surveillance Time[c] (n2[d])	Placebo N[a]=20172 Cases n1[b] Surveillance Time[c] (n2[d])	Vaccine Efficacy % (95% CI)[e]
Overall	9 2.332 (18559)	169 2.345 (18708)	94.6 (89.6, 97.6)
Age group (years)			
16 to 17	0 0.003 (58)	1 0.003 (61)	100.0 (-3969.9, 100.0)
18 to 64	8 1.799 (14443)	149 1.811 (14566)	94.6 (89.1, 97.7)
65 to 74	1 0.424 (3239)	14 0.423 (3255)	92.9 (53.2, 99.8)
≥75	0 0.106 (805)	5 0.109 (812)	100.0 (-12.1, 100.0)

Figure 2.2: The aggregate data from the Pfizer RCT; source: FDA briefing.

We see that the rate of Covid-19 infection was relatively low at the time. Specifically, the treatment group saw 9 Covid-19 cases per 19,965, while the control group saw 169 cases per 20,172.

The estimated average treatment effect is about

$$-792.7 \text{ cases per } 100{,}000,$$

and the 95% confidence band is[9]

$$[-922, -664].$$

Under Assumptions 2.1.3 and 2.2.1 the confidence band suggests that the Covid-19 vaccine caused a reduction in the risk of contracting Covid-19.

We also compute the Vaccine Efficacy metric, which according to [11], refers to the following measure:

$$\mathsf{VE} = \frac{\text{Risk for Unvaccinated - Risk for Vaccinated}}{\text{Risk for Unvaccinated}}.$$

It describes the relative reduction in risk caused by vaccination. Estimating the VE is simple as we can plug-in the estimated group means. We can compute standard errors using the delta method or by simulation. We obtain that the overall vaccine efficacy is 94.6%, replicating the results shown in Figure 2.2. Our 95% confidence interval for VE, based on the normal approximation, is

$$[90.9\%, 98.2\%],$$

which differs only slightly from the FDA briefing table.[10]

Remark 2.1.1 We notice that the confidence intervals for the VE for the two age groups of seniors are very wide, so to increase precision we pool them together and calculate the effectiveness of the vaccine for the two groups that are 65 or older. The resulting VE estimate is 95% and the two-sided

9: In this example, we don't need the underlying individual data to evaluate the effectiveness of the vaccine because the potential outcomes are Bernoulli random variables with mean $\mathrm{E}[Y(d)]$ and variance $\mathrm{Var}(Y(d)) = \mathrm{E}[Y(d)(1 - \mathrm{E}Y(d))]$.

10: The analysis in the FDA table is based on the inversion of exact binomial tests, the Cornfield procedure.

confidence interval based on the normal approximation is

$$[82\%, 106\%]$$

A more refined approach is possible, based on the inversion of exact binomial ratio Cornfield tests [12], which we report in Vaccination RCT R Notebook and Vaccination RCT Python Notebook. This approach, using Vaccination RCT R Notebook, yields a confidence interval of

$$[69\%, 99\%].$$

The reason is that the accumulated counts of binomials are too few for the Gaussian approximations to provide a high-quality approximation, so the exact binomial ratio test inversion delivers a more accurate confidence interval.

2.2 Pre-treatment Covariates and Heterogeneity

Sometimes we also have additional *pre-treatment* or *pre-determined* covariates W. We might be interested in either using these covariates to estimate average effects more precisely or to describe heterogeneity of the treatment effects. For example, we might be interested in the impact of a treatment across age or income groups.

For this purpose, we consider conditional average treatment effects (CATE):

$$\delta(W) = \mathrm{E}[Y(1) \mid W] - \mathrm{E}[Y(0) \mid W],$$

which compare the average potential outcomes conditional on a set of covariates W.

We can directly learn the conditional predictive effects (CAPE),

$$\pi(W) = \mathrm{E}[Y \mid D = 1, W] - \mathrm{E}[Y \mid D = 0, W],$$

from population data. However, these CAPE will generally not agree with the CATE. One assumption that will be sufficient for the CAPE and CATE to agree is having treatment assigned randomly and independently of covariates. As before, the use of RCTs help ensure the plausibility of this assumption.

Assumption 2.2.1 (Random Assignment Independent of Co-

variates) *Suppose that treatment status is randomly assigned. Namely, D is statistically independent of both the potential outcomes and a set of pre-determined covariates:*

$$D \perp\!\!\!\perp (Y(0), Y(1), W),$$

and $0 < \mathrm{P}(D = 1) < 1$.

This assumption spells out that, if we plan to use covariates in the analysis, randomization has to be made with respect to these covariates as well. In practice, it is often tempting to use post-treatment covariates, but the use of such variables runs the danger of violating Assumption 2.2.1. In the extreme case, conditioning on the post-treatment observed outcome Y, we find that $\pi(Y) = 0$, even when there is a treatment effect. In a less extreme case, conditioning on post-treatment variables related to the outcome can "control-away" part of the effect, diminishing estimates.

A common scenario where accidentally using a post-treatment covariate may occur is when researchers encounter missing data from imperfect data collection in following-up with control and treated units to collect demographic information. When we drop observations with missing data, we implicitly condition on a post-treatment variable (missingness) which can cause violations of Assumption 2.2.1.

The desire to assess randomization with respect to covariates motivates the following diagnostic procedure.

For random variables A and B, $A \sim B$ denotes that A and B have the same distribution.

Testing Covariance Balance. The random assignment assumption induces covariate balance. Namely, the distribution of covariates should be the same under both treatment and control:

$$W|D = 1 \sim W|D = 0,$$

and, equivalently,

$$D|W \sim D.$$

A useful implication is that D is not predictable by W:

$$\mathrm{E}[D \mid W] = \mathrm{E}[D].$$

This latter conditions is testable using regression tools. It amounts to saying that the R^2 of a regression of D on W is 0.

Under Assumption 2.2.1, Theorem 2.1.1 continues to hold, but we now have a stronger result.

Theorem 2.2.1 (Randomization with Covariates) *Under Assumption 2.2.1, the expected value of Y conditional on treatment status $D = d$ and covariates W coincides with the expected value of potential outcome $Y(d)$ conditional on covariates W:*

$$\mathrm{E}[Y \mid D = d, W] = \mathrm{E}[Y(d) \mid D = d, W] = \mathrm{E}[Y(d)|W],$$

for each d. Hence the conditional predictive and average treatment effects agree:

$$\pi(W) = \delta(W).$$

Regression and Statistical Inference for ATEs

Empirical researchers often base statistical inference on the ATE using the classical additive linear regression model, where covariates enter additively in the model. This approach has some good practical properties and often empirically leads to improvements in precision over the simple two-means approach, though this precision improvement is not guaranteed. Another approach that we will emphasize is the interactive regression approach, where de-meaned covariates are also interacted with the base treatment. Including interactions of de-meaned covariates with the treatment always improves precision, and it also allows us to discover treatment effect heterogeneity.

Classical Additive Approach: Improving Precision Under Linearity

We begin explaining the classical additive approach. Here, to simplify the exposition, we make the strong assumption that the conditional expectation function is exactly linear:

$$\mathrm{E}[Y \mid D, W] = D\alpha + \beta' X, \tag{2.2.1}$$

where $X = (1, W)$ contains an intercept and pre-treatment covariates W. This setup is clearly restrictive, but the statistical inference result will be valid without this assumption.[11] Later in the book, we will consider fully nonlinear models.

We assume that covariates are centered:[12]

$$\mathrm{E}[W] = 0.$$

By Assumption 2.2.1, there is covariate balance:

$$\mathrm{E}[W \mid D = 1] = \mathrm{E}[W \mid D = 0].$$

11: See Section 2.B for details.

12: Theoretically, this is implemented by redefining $W := W - \mathrm{E}[W]$. In estimation, this is implemented by redefining $W_i := W_i - \mathbb{E}_n[W]$.

Using centered covariates implies that

$$\mathrm{E}[Y(0)] = \mathrm{E}[\mathrm{E}[Y \mid D = 0, X]] = \beta_1$$

$$\mathrm{E}[Y(1)] = \mathrm{E}[\mathrm{E}[Y \mid D = 1, X]] = \beta_1 + \alpha.$$

That is, the average outcome in the untreated state is β_1, and the average treatment effect $\delta = \mathrm{E}[Y(1)] - \mathrm{E}[Y(0)]$ equals α.

Equation (2.2.1) implies that

$$Y = D\alpha + \beta' X + \epsilon, \quad \epsilon \perp (D, X), \tag{2.2.2}$$

implying that α coincides with the coefficient in the BLP of Y on D and X. In fact, even if we don't assume the model (2.2.1), we still have that $\alpha = \delta$. That is, the projection coefficient α recovers the ATE δ without the linearity assumption as we detail in Section 2.B. Furthermore the statistical inference result stated below will hold without requiring linear conditional expectation functions as it is simply a statement about inference on the BLP.

We are interested in statistical inference on the ATE and Relative ATE[13]

$$\alpha \quad \text{and} \quad \alpha/\beta_1.$$

13: Relative ATE is often called *lift* in business applications.

Under regularity conditions, application of the OLS theory from Chapter 1 gives us

$$\begin{pmatrix} \sqrt{n}(\hat{\alpha} - \alpha) \\ \sqrt{n}(\hat{\beta}_1 - \beta_1) \end{pmatrix} \overset{\text{a}}{\sim} N(0, \mathsf{V}),$$

where covariance matrix V has components:

$$\mathsf{V}_{11} = \frac{\mathrm{E}[\epsilon^2 \tilde{D}^2]}{(\mathrm{E}[\tilde{D}^2])^2}, \quad \mathsf{V}_{22} = \frac{\mathrm{E}[\epsilon^2 \tilde{1}^2]}{(\mathrm{E}[\tilde{1}^2])^2}, \quad \mathsf{V}_{12} = \mathsf{V}_{21} = \frac{\mathrm{E}[\epsilon^2 \tilde{D}\tilde{1}]}{\mathrm{E}[\tilde{1}^2]\mathrm{E}[\tilde{D}^2]},$$

where $\tilde{D} = D - \mathrm{E}[D]$ is the residual after partialling out X from D linearly and $\tilde{1} := (1 - D)$ is the residual after partialling out D and W from 1.

We also obtain the approximate normality for the Relative ATE using the delta method:

$$\sqrt{n}(\hat{\alpha}/\hat{\beta}_1 - \alpha/\beta_1) \overset{\text{a}}{\sim} N(0, G'\mathsf{V}G),$$

where

$$G = [1/\beta_1, -\alpha/\beta_1^2]'.$$

Improvement in Precision under Linearity

Now we explain the role of covariates in potentially delivering improvements in precision of estimating the ATE. The underlying idea is that of "denoising." This improvement, however, hinges on the linear model (2.2.1). In the next section, we will obtain improvement without linearity assumptions.

We consider what happens when we do not include covariates in the regression. In this case, the OLS estimator $\bar{\alpha}$ estimates the projection coefficient α in the BLP using $(1, D)$ alone:[14]

$$Y = \alpha D + \beta_1 + U, \quad \mathrm{E}[U] = \mathrm{E}[UD] = 0,$$

where the noise

$$U = \beta'(X - \mathrm{E}[X]) + \epsilon$$

contains the part of Y that is linearly predicted by X, $\beta'(X - \mathrm{E}[X]) = \beta' X - \beta_1$. We then have that $\bar{\alpha}$ obeys

$$\sqrt{n}(\bar{\alpha} - \alpha) \overset{a}{\sim} N(0, \bar{\mathsf{V}}_{11}), \;\; \bar{\mathsf{V}}_{11} = \frac{\mathrm{E}[U^2 \tilde{D}^2]}{(\mathrm{E}[\tilde{D}^2])^2}.$$

Under the linear model (2.2.1), it follows that

$$\mathsf{V}_{11} \leq \bar{\mathsf{V}}_{11},$$

with the inequality being strict ("<") if $\mathrm{Var}(\beta' X) > 0$.[15] That is, under (2.2.1), using pre-determined covariates improves the precision of estimating the ATE α.

However, this improvement theoretically hinges on the correctness of the additive linear model. Statistical inference on the ATE based on the the normal approximation provided above remains valid without this assumption as long as robust standard errors are used.[16] However, the precision can be either higher or lower than that of the classical two-sample approach without covariates. That is, without (2.2.1), V_{11} and $\bar{\mathsf{V}}_{11}$ are not generally comparable.

Remark 2.2.1 While the inferential result we derived is robust with respect to the linearity assumption on the CEF, the improvement in precision itself is **not** guaranteed in general and hinges on the validity of the linearity assumption. We provide simulation examples where controlling for predetermined covariates linearly lowers the precision (increases robust standard errors) in Covariates in RCT R Notebook and

14: Here $U = Y - \alpha D - \beta_1$ obeys

$$\begin{aligned}\mathrm{E}[U \mid D = d] &= \mathrm{E}[Y(d) - \alpha d - \beta_1 \mid D = d] \\ &= \mathrm{E}[Y(d) - \alpha d - \beta_1] = 0,\end{aligned}$$

invoking random assignment and the definition of α and β_1.

15: Verify this as a reading exercise.

16: We always use robust variance formulas throughout the book. However, the default inferential algorithms in R and Python often report the classical Student's formulas as variances, which critically rely on the linearity assumption.

Covariates in RCT Python Notebook.

The Interactive Approach: Always Improves Precision and Discovers Heterogeneity

Covariates in RCT R Notebook and Covariates in RCT Python Notebook explore the use of covariates to both improve precision and learn about heterogeneity via a simulation experiment.

We can also consider estimation of CATE through the lens of an interactive linear regression model, which interacts treatment indicator D with regressors X constructed from original raw regressors W. Including these interactions respects the logic of approximating the conditional expectation of Y given D and raw regressors using linear functional forms. To simplify exposition, we first assume that the interactive model is exactly correct for the CEF:

$$\mathrm{E}[Y \mid D, W] = \alpha' XD + \beta' X. \tag{2.2.3}$$

In Section 2.C, we explain how this approach works without this assumption.

As before, we assume

$$X = (1, W')', \quad \mathrm{E}[W] = 0,$$

which can be achieved in practice by recentering. Here, we recover CATE via

$$\begin{aligned}\delta(W) &= \mathrm{E}[Y(1) \mid W] - \mathrm{E}[Y(0) \mid W] \\ &= \mathrm{E}[Y \mid D = 1, W] - \mathrm{E}[Y \mid D = 0, W] = \alpha' X.\end{aligned}$$

Using that $\mathrm{E}W = 0$, the ATE is then

$$\delta = \mathrm{E}[\delta(W)] = \mathrm{E}[\alpha' X] = \alpha_1,$$

where α_1 is the first component of α. The function $\alpha_2' W$, where α_2 is the vector all elements of α excluding α_1, therefore describes the deviation of CATE away from the ATE.

We can verify that α is the coefficient of the linear projection equation:

$$Y = \alpha' DX + \beta' X + \epsilon, \;\; \epsilon \perp (X, DX).$$

Therefore, we can treat

$$\bar{D} := DX$$

as a vector of technical treatments[17] and invoke the "partialling

17: A technical treatment refers to any variable obtained as a transformation of the original treatment variable.

out" approach for inference on components of α. The variance formulas are given in Section 2.C.

Remark 2.2.2 (Improvement in Precision Guarantee) Unlike the previous approach, the "interactive" approach always delivers improvements in precision for estimating δ, even if the linearity in (2.2.3) does not hold; this was demonstrated by Lin [13]. Section 2.C explains this point in detail and provides a deeper dive into the properties of the interactive approach without assuming correct linear specification of the CEF.

Reemployment Bonus RCT

Reemployment Bonus RCT R Notebook and Reemployment Bonus RCT Python Notebook explore the use of covariates to improve precision and learn about heterogeneity in a Reemployment Bonus RCT.

Here we re-analyze the Pennsylvania re-employment bonus experiment [14], which was conducted in the 1980s by the U.S. Department of Labor to test the incentive effects of alternative compensation schemes for unemployment insurance (UI). In these experiments, UI claimants were randomly assigned either to a control group or one of five treatment groups. We focus our discussion on treatment group 4. In the control group the current rules of the UI applied. Individuals in the treatment groups were offered a cash bonus if they found a job within some pre-specified period of time (qualification period), provided that the job was retained for a specified duration; see the Penn Data Codebook for further details on the data.

We consider the

- classical 2-sample approach, no adjustment (CL)
- classical linear regression adjustment (CRA)
- interactive regression adjustment (IRA)
- interactive regression adjustment with double lasso (partialling out by lasso) (IRA-DL)

We use the last approach in the spirit of exploration and experimentation. We describe the last approach and establish its validity in Chapter 4.

Estimates of the ATE on (log) unemployment duration and corresponding estimated standard errors are given in Table 2.1.

Table 2.1: Estimates of the ATE of the reemployment bonus on log unemployment duration..

	CL	CRA	IRA	IRA-DL
Estimate	-0.0855	-0.0797	-0.0755	-0.0789
Std. Error	0.0359	0.0356	0.0356	0.0356

The different estimators deliver fairly similar point estimates suggesting that treatment group 4 experiences an average decrease in unemployment duration of around 8%. The three regression estimators deliver estimates that are slightly more precise (have lower standard errors) than the simple difference in means estimator.

We also see that the regression estimators offer slightly lower estimates of the ATE than the difference in means estimator. These differences likely occur due to minor imbalances in the treatment allocation: People older than 54 tended to receive the treatment more than other groups of qualified UI claimants during the later period of the experiment. Loosely speaking, the regression estimators try to correct for this imbalance by "partialling out" the effect of this oversampling and averaging over differences net of these "imbalancing" effects. We will explain how regression adjustment corrects for imbalances in Chapter 5.

See Reemployment Bonus RCT R Notebook and Reemployment Bonus RCT Python Notebook for the results from the balance check.

2.3 Drawing RCTs via Causal Diagrams

RCTs can be visualized using causal diagrams. These enable us to simply and clearly show the causal assumptions that underpin our model for retrieving treatment effects. Causal diagrams were introduced as early as 1920s by Sewall and Philip Wright ([15],[16]) and emerged as a fully formal tool due to the work of Judea Pearl and James H. Robins ([17], [18]).

In causal diagrams, random variables are denoted by nodes; and arrows between nodes represent causal effects. In our RCT set-up, we have that the assigned treatment variable causes outcome variable Y, and the pre-treatment variables W also cause the outcome variable Y, but they don't cause the treatment assignment D. This causal diagram is illustrated in Figure 2.3 below.

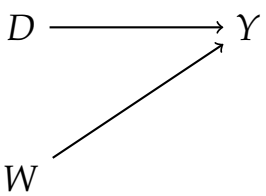


Figure 2.3: Causal Diagram for a RCT

Figure 2.4 depicts a version of the diagram that also includes potential outcomes as nodes.

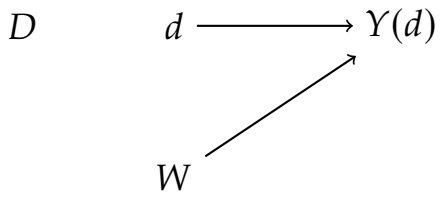


Figure 2.4: A Causal Diagram for the RCT Research Design

In Figure 2.4, we show the potential outcomes $Y(d)$ as a single node. The pre-treatment covariates affect this node, which is represented by the arrow from W to the $Y(d)$ node. The assigned treatment variable D is independent of the node $Y(d)$, which is shown by the absence of an arrow connecting the two nodes. The arrow from d to $Y(d)$ shows the causal dependency of $Y(d)$ on the deterministic node d. The assigned treatment D is also shown to be independent of the node W. The potential outcome process $d \mapsto Y(d)$ and treatment assignment jointly determine the realized outcome variable Y via the assignment $Y := Y(D)$.

We further develop the use of these concepts and the use of causal diagrams as a formal tool in Chapter 7 and Chapter 8.

2.4 The Limitations of RCTs

Here, we briefly outline some of the primary limitations of RCTs. We first consider threats to identification, outlining settings in which the stable unit treatment value assumption (SUTVA), an important assumption that underpins causal inference in an RCT setting, is unlikely to hold, and the implications for inference. We then address ethical and practical concerns in RCT implementation and generalizability.

Externalities, Stability, and Equilibrium Effects

The traditional formulation of Rubin's causal model relies on SUTVA as described in Section 2.1. Part of SUTVA is the requirement that the potential outcomes of one unit should be unaffected by the assignment of treatments to other units [19]. In the following, we consider some cases where this assumption might not hold.

In a vaccine example, this assumption holds if treatment and control populations are "small" (infinitesimal) subpopulations of the entire general population. Our methods measure the average vaccine effects in these settings. However, if we vaccinate a

sufficiently large percentage of people, reaching herd immunity, the outcomes for the control group would be essentially the same as outcomes for the treated. SUTVA therefore would not hold.[18]

18: Because SUTVA does not hold in the vaccination context, it is customary to use relative measures of impact like "vaccine efficiency" because they may be a somewhat more stable measure when generalizing from "small" treated subpopulations to a "large" treated population.

In economics, we refer to such spillover effects as externalities or, in some contexts, as general equilibrium effects. For example, there is a positive externality created by people who take the vaccine (and people that don't take vaccine "free ride," once the vaccination level is high enough). Consider another example. We might want to study the earning effect of getting a college degree versus not having a college degree. If treatment will target a relatively small subpopulation of people, there likely won't be any large general equilibrium wage effects. On the other hand, if the treatment will target a large subpopulation, the equilibrium wage will likely adjust (the college wage premium might decrease, for example). In another example, the outcomes for one individual in large-scale training programs may be affected by the number of people trained to perform the same job.

Ethical, Practical, and Generalizability Concerns

Many RCTs are infeasible because implementing them would be unethical. The general ethical principles and guidelines for research involving human subjects are set out in the 1978 Belmont report ([20]). The key ethical principles are "Respect for persons," "Beneficence," and "Justice." Human subject trials are subject to regulation by an institutional review board, which determines whether the trial is ethical with reference to these guiding principles, or whether it should be prevented from registering.

For example, we previously considered a hypothetical RCT where individuals are assigned to a smoking treatment group. The trial would violate the principle of "beneficence" as the researcher might be causing physical harm to study participants by assigning them to smoking. Thus, RCTs are rarely a feasible means of retrieving the causal effects of harmful interventions as they tend to be unethical.

RCTs may also face practical issues. They can be prohibitively expensive when the treatment is costly, data collection costs are high, or the sample size required for adequate power is high. These issues make it difficult to implement long-term RCTs and find evidence on the long-term effects of interventions, particularly because they are more likely to suffer from attrition.

It may also be politically infeasible for policymakers to enforce randomization of receipt of a desirable treatment.

Even in the best case, where an RCT is successfully implemented and we are confident in our retrieved average treatment effect, it may be difficult to generalize (or extrapolate) the result of an RCT in a specific context to a general finding. This difficulty might be because local conditions or implementation capacity materially differ between where interventions are staged or because the scale of the intervention is important.

Notebooks

- Vaccination RCT R Notebook and Vaccination RCT Python Notebook contain the analysis of vaccination examples.
- Covariates in RCT R Notebook and Covariates in RCT Python Notebook explore the use of covariates to improve precision and learn about heterogeneity via a simulation experiment.
- Reemployment Bonus RCT R Notebook and Reemployment Bonus RCT Python Notebook explore the use of covariates to improve precision and learn about heterogeneity in a Reemployment Bonus RCT.

Notes

RCTs have a profound influence on business, economics and science more generally. For example, RCTs are routinely used to study the efficacy of drugs and efficacy of various programs in labor and development economics, among other subfields of economics. The FDA moved to RCTs as the gold standard of proving that treatments work in 1970s-80s. In the tech industry and marketing, RCTs are also called "A/B Tests" and are now widely used. Many major tech companies have their own experimental platforms to carry out thousands of experiments.[19]

The expansion of the use of experimentation in economics is associated with the work of Richard Thaler, the recipient of the 2017 Alfred Nobel Memorial Prize in Economics;[20] Abhijit Banerjee, Esther Duflo, and Michael Kremer, the recipients of the 2019 Alfred Nobel Memorial Prize in Economics;[21] and John List, among many others.

19: See, for example, ExP platform at Microsoft and the WebLab platform at Amazon.

20: "for his contributions to behavioural economics." Source: NobelPrize.org

21: "for their experimental approach to alleviating global poverty." Source: NobelPrize.org

We touched upon very basic ideas here. The basic random design is just one of many possible randomized designs that allow us to uncover causal effects. For an in-depth analysis of design of experiments, please see lecture notes by Art Owen ([21]). For standard RCTs and causal analysis more generally, see the book by Imbens and Rubin [10]. Duflo et al. [22] is another good overview of the use of RCTs with a focus on development economics applications. For real examples of how RCTs are done and designed in practice, see, for example, the FDA registry of RCTs, the American Economic Association for a registry of RCTs in economics, or the The Poverty Action Lab.

Study Questions

1. Set-up a simulation experiment that illustrates the contrived smoking example, following the analytical example we've presented in the text. Illustrate the difference between estimates obtained via an RCT (smoking generated independently of potential outcomes) and an observational study (smoking choice is correlated with potential outcomes).

2. Sketch out the proof of the large sample properties of the two means estimator.

3. Study the notebook on vaccinations RCTs. Try to replicate the results in the FDA briefing table for each age 18-64 (exact replication is not required). Explain your calculations.

4. Study the notebook on the reemployment example. Experiment with putting even more flexible controls (e.g. use extra interactions of some controls). Report your findings.

5. Work and experiment with the Covariates in RCT notebook. Explain the main points being made.

6. Skim over the information on the Pfizer RCT design briefing. Write down one paragraph summarizing the study design.

7. Skim over one of the RCTs registered with AEA RCT Registry. Write down one paragraph summarizing the study design.

8. Think of some RCTs where stability (SUTVA) is likely to hold and some RCTs where it likely does not.

9. Explain why we can't learn individual treatment effects by first putting a unit in treatment and then putting the individual in control second (or the other way around). A hint is to think of all sources of randomness represented by ω. Would the situation be different if you had a time machine?

2.A Approximate Distribution of the Two Sample Means

To demonstrate the result in the text, we note that

$$\hat{\theta}_d - \theta_d = \frac{\mathbb{E}_n[(Y(d) - \mathrm{E}Y(d))1(D = d)]}{\mathbb{E}_n[1(D = d)]}$$

for $d \in \{0, 1\}$ because we can re-write the population group average as

$$\theta_d = \mathrm{E}[Y(d)] = \mathrm{E}[Y(d)]\frac{\mathbb{E}_n[1(D = d)]}{\mathbb{E}_n[1(D = d)]}.$$

Hence, for each $d \in \{0, 1\}$,

$$\sqrt{n}(\hat{\theta}_d - \theta_d) = \sqrt{n}\frac{\mathbb{E}_n[(Y(d) - \mathrm{E}Y(d))1(D = d)]}{\mathbb{E}_n[1(D = d)]}.$$

By the law of large numbers, $\mathbb{E}_n[1(D = d)] \approx \mathrm{P}(D = d)$; so we have the approximation

$$\sqrt{n}\{\hat{\theta}_d - \theta_d\}_{d\in\{0,1\}} \approx \sqrt{n}\frac{\mathbb{E}_n[(Y(d) - \mathrm{E}Y(d))1(D = d)]}{\mathrm{P}(D = d)}.$$

Note that the terms being averaged are

$$\frac{(Y_i(d) - \mathrm{E}[Y(d)])1(D_i = d)}{\mathrm{P}(D = d)}.$$

These terms have zero mean[22] and variance

$$\frac{\mathrm{E}[(Y(d) - \mathrm{E}[Y(d)])^2 1(D = d)^2]}{\mathrm{P}(D = d)^2} = \frac{\mathrm{Var}(Y \mid 1(D = d) = 1)}{\mathrm{P}(D = d)}.$$

22: Why? Hint: Use the law of iterated expectations.

Also note the zero covariance:

$$\mathrm{E}\left[\frac{(Y(1)-\mathrm{E}[Y(1)])1(D=1)}{\mathrm{P}(D=1)}\frac{(Y(0)-\mathrm{E}[Y(0)])1(D=0)}{\mathrm{P}(D=0)}\right]=0.$$

The application of the central limit theorem then yields the claimed result.

2.B Statistical Properties of the Classical Additive Approach*

Here we analyze statistical inference on ATE using OLS and adjusting for $X=(1,W)$, without making the linearity assumptions we made in Section 2.2.

We consider the linear projection equation in the population:

$$Y=D\alpha+X'\beta+\epsilon,\quad \epsilon\perp(D,X).$$

Here, we have that D and $X=(1,W)$ with $\mathrm{E}[W]=0$, so that $\beta'X=\beta_1+\beta_2'W$. Moreover, we have that $D\perp W$ in the RCT setting.

First, we'd like to verify that $\alpha=\mathrm{E}[Y(1)]-\mathrm{E}[Y(0)]$ and $\beta_1=\mathrm{E}[Y(0)]$. For $U:=\beta_2'W+\epsilon$, we can write

$$Y=D\alpha+\beta_1+U,\quad U\perp(1,D).$$

$U\perp(1,D)$ holds because $(1,D)\perp(W,\epsilon)$ using that $\mathrm{E}[W]=0$ and that $D\perp(W,\epsilon)$. Therefore, $D\alpha+\beta_1$ coincides with the population projection of Y onto $(1,D)$. Hence, the projection coefficients are the same as those obtained by the 2-sample approach in the population. Therefore, $\beta_1=\mathrm{E}[Y(0)]$ and $\alpha=\mathrm{E}[Y(1)]-\mathrm{E}[Y(0)]$.

Second, we'd like to explain the details of the approximate normality for the estimators of sample OLS coefficients $\hat\beta_1$. The OLS theory of the first chapter implies that the OLS estimator $\hat\alpha$ obeys

$$\sqrt{n}(\hat\alpha-\alpha)\approx\sqrt{n}\frac{\mathbb{E}_n[\epsilon\tilde D]}{\mathbb{E}_n[\tilde D^2]}\overset{\mathrm{a}}{\sim}N(0,\mathsf{V}_{11}),$$

where $\tilde D=D-\mathrm{E}[D]$ is the residual after partialling out X from D linearly,[23] and

$$\mathsf{V}_{11}=\frac{\mathrm{E}[\epsilon^2\tilde D^2]}{(\mathrm{E}[\tilde D^2])^2}.$$

23: Derive that $\tilde D=D-\mathrm{E}[D]$ from Assumption 2.2.1.

Applying the same theory for β_1 (the intercept coefficient), yields[24]

$$\sqrt{n}(\hat{\beta}_1 - \beta_1) \approx \sqrt{n}\frac{\mathbb{E}_n[\epsilon\tilde{1}]}{\mathbb{E}_n[\tilde{1}^2]} \overset{a}{\sim} N(0, \mathsf{V}_{22}),$$

where $\tilde{1} := (1 - D)$ is the residual after partialling out D and X from 1 and

$$\mathsf{V}_{22} = \frac{\mathrm{E}[\epsilon^2\tilde{1}^2]}{(\mathrm{E}[\tilde{1}^2])^2}.$$

We can also establish that the estimators are jointly approximately normal with covariance

$$\mathsf{V}_{12} = \frac{\mathrm{E}[\epsilon^2\tilde{D}\tilde{1}]}{\mathrm{E}[\tilde{1}^2]\mathrm{E}[\tilde{D}^2]}.$$

24: To explain the derivation, note that by partialling out D and W (recall that $X = (1, W)$) from 1 and Y, we obtain

$$\tilde{Y} = \beta_1\tilde{1} + \epsilon; \quad \tilde{1} := (1 - D).$$

The projection of 1 on D and W is given by D since D is binary and we've assumed $\mathrm{E}[W] = 0$.

2.C Statistical Properties of the Interactive Regression Approach*

Here we analyze the estimation of the ATE using OLS and adjusting for (W, DW) without making any linearity assumptions on the potential outcomes as we did in Section 2.2. We essentially show that the interactive model can be viewed as estimating the BLP of each of the two potential outcomes $Y(0)$ and $Y(1)$. Using this fact one can then easily argue that the variance of the OLS estimate of the effect using the interactive model can only be lower than the variance of the unadjusted OLS estimate.

Letting $X = (1, W)$ be an intercept and the pre-treatment covariates W, let us write the BLP of each of $Y(0)$ and $Y(1)$ using X as

$$Y(d) = \beta_d' X + \varepsilon_d, \quad \varepsilon_d \perp X, \quad d = 0, 1. \tag{2.C.1}$$

Under Assumption 2.2.1, (2.C.1) coincides with the BLP of Y using X in the $D = d$ population. Letting $\varepsilon = D\varepsilon_1 + (1 - D)\varepsilon_0$, we thus have

$$Y = \beta_d' X + \varepsilon, \quad \mathrm{E}[\varepsilon X \mid D = d] = 0, \quad d = 0, 1. \tag{2.C.2}$$

The BLPs in each of the two populations, $D = 0$ and $D = 1$, can be combined across the populations to state the BLP of Y using (X, DX) marginally:

$$Y = \beta_0' X + \beta_\delta' XD + \varepsilon, \quad \varepsilon \perp (X, DX), \tag{2.C.3}$$

where $\beta_\delta = \beta_1 - \beta_0$.[25] Such a linear rule is called *interactive* because it includes the interaction (meaning, product) of D and W as a regressor, in addition to D and W.

25: Note that (2.C.1) and (2.C.2) imply $\mathrm{E}[\varepsilon DX] = 0$ and $\mathrm{E}[\varepsilon X] = 0$ and thus that $\varepsilon \perp (X, DX)$.

We assume that covariates are centered:

$$\mathrm{E}[W] = 0.$$

Since X contains an intercept, $\varepsilon_d \perp X$ implies $\mathrm{E}[\varepsilon_d] = 0$. Together with centered covariates, we find that

$$\mathrm{E}[Y(d)] = \mathrm{E}[\beta_d' X + \varepsilon_d] = \beta_{d,1}.$$

This means that the ATE coincides with the coefficient on D in the BLP of Y using (X, DX). That is, $\beta_{\delta,1} = \delta$.

We are often interested in the ATE and Relative ATE

$$\delta \quad \text{and} \quad \delta/\mathrm{E}[Y(0)].$$

If we use OLS to estimate the BLP of Y using (X, DX), then an application of the OLS theory in the previous chapter gives us that, under regularity conditions,

$$\begin{pmatrix} \sqrt{n}(\hat{\beta}_{\delta,1} - \delta) \\ \sqrt{n}(\hat{\beta}_{0,1} - \mathrm{E}[Y(0)]) \end{pmatrix} \overset{\text{a}}{\sim} N(0, \mathsf{V}),$$

where covariance matrix V has components:

$$\mathsf{V}_{11} = \frac{\mathrm{E}[\epsilon^2 \tilde{D}^2]}{(\mathrm{E}[\tilde{D}^2])^2}, \quad \mathsf{V}_{22} = \frac{\mathrm{E}[\epsilon^2 \tilde{1}^2]}{(\mathrm{E}[\tilde{1}^2])^2}, \quad \mathsf{V}_{12} = \mathsf{V}_{21} = \frac{\mathrm{E}[\epsilon^2 \tilde{D}\tilde{1}]}{\mathrm{E}[\tilde{1}^2]\mathrm{E}[\tilde{D}^2]},$$

where $\tilde{D} = D - \mathrm{E}[D]$ is the residual after partialling out linearly $(1, W, DW)$ from D and $\tilde{1} := (1 - D)$ is the residual after partialling out (D, W, DW) from 1.[26]

26: The derivation follows identical steps as that in Section 2.B with the only exception that when defining $\tilde{D}$ we need to partial out $(1, W, DW)$ from D and when defining $\tilde{1}$ we need to partial out (D, W, DW) from 1. However, since $\mathrm{E}[W] = \mathrm{E}[DW] = 0$, the two residuals take the same form of $D - \mathrm{E}[D]$ and $1 - D$ correspondingly.

We can then obtain the approximate normality for the Relative ATE using the delta method:

$$\sqrt{n}(\hat{\beta}_{\delta,1}/\hat{\beta}_{0,1} - \delta/\mathrm{E}[Y(0)]) \overset{\text{a}}{\sim} N(0, G'\mathsf{V}G),$$

where

$$G = [1/\mathrm{E}[Y(0)], -\delta/(\mathrm{E}[Y(0)])^2]'.$$

We can rewrite (2.C.3) as

$$Y = \beta_{0,1} + D\beta_{\delta,1} + U, \quad U = \beta_{0,2}' W + \beta_{\delta,2}' WD + \varepsilon.$$

From $\varepsilon \perp (X, D, DX)$, $\mathrm{E}[W] = 0$, and Assumption 2.2.1, we obtain that $U \perp (1, D)$, meaning that $\beta_{0,1} + D\beta_{\delta,1}$ is the BLP

of Y using $(1, D)$. We can therefore estimate the ATE as the coefficient on D *either* in the OLS of Y on $(1, D)$ *or* in the OLS of Y on (X, DX). The former exactly coincides with the unadjusted estimator $\hat{\delta}$ from Section 2.1, which obeys

$$\sqrt{n}(\hat{\delta} - \delta) \overset{a}{\sim} N(0, \bar{\mathsf{V}}_{11}), \quad \bar{\mathsf{V}}_{11} = \frac{\mathrm{E}[U^2 \tilde{D}^2]}{(\mathrm{E}[\tilde{D}^2])^2}.$$

Since ϵ satisfies the BLP conditions for each of the treatment populations, i.e. $\mathrm{E}[\varepsilon W \mid D = d] = 0$, it then follows that

$$\mathsf{V}_{11} \leq \bar{\mathsf{V}}_{11}.$$

Moreover, the inequality is strict if $\mathrm{Var}(\beta_{0,2}' W) > 0$ or $\mathrm{Var}(\beta_{1,2}' W) > 0$.[27] That is, pre-determined covariates improve the precision of estimating the ATE δ, when using the interactive model, without any linearity assumptions on the CEF.

27: Verify this as a reading exercise.

Bibliography

[1] Jan Baptist van Helmont. *Oriatrike, Or, Physick Refined, the Common Errors Therein Refuted, and the Whole Art Reformed and Rectified*. Loyd, London, 1662 (cited on page 41).

[2] Donald B. Rubin. 'Estimating causal effects of treatments in randomized and nonrandomized studies.' In: *Journal of Educational Psychology* 66.5 (1974), pp. 688–701 (cited on page 42).

[3] P. Holland. 'Causal Inference, Path Analysis, and Recursive Structural Equations Models'. In: *Sociological Methodology*. Washington, DC: American Sociological Association, 1986, pp. 449–493 (cited on page 42).

[4] Peng Ding and Fan Li. 'Causal Inference: A Missing Data Perspective'. In: *Statistical Science* 33.2 (2018), pp. 214 –237. DOI: 10.1214/18-STS645 (cited on page 42).

[5] Tyler J. VanderWeele, Guanglei Hong, Stephanie M. Jones, and Joshua L. Brown. 'Mediation and Spillover Effects in Group-Randomized Trials: A Case Study of the 4Rs Educational Intervention'. In: *Journal of the American Statistical Association* 108.502 (2013), pp. 469–482. (Visited on 02/17/2024) (cited on page 43).

[6] Peter M. Aronow and Cyrus Samii. 'Estimating average causal effects under general interference, with application to a social network experiment'. In: *The Annals of Applied Statistics* 11.4 (2017), pp. 1912 –1947. DOI: 10.1214/16-AOAS1005 (cited on page 43).

[7] Michael P. Leung. 'Treatment and Spillover Effects Under Network Interference'. In: *The Review of Economics and Statistics* 102.2 (2020), pp. 368–380 (cited on page 43).

[8] Francis J. DiTraglia, Camilo García-Jimeno, Rossa O'Keeffe-O'Donovan, and Alejandro Sánchez-Becerra. 'Identifying causal effects in experiments with spillovers and non-compliance'. In: *Journal of Econometrics* 235.2 (2023), pp. 1589–1624. DOI: https://doi.org/10.1016/j.jeconom.2023.01.008 (cited on page 43).

[9] Gonzalo Vazquez-Bare. 'Identification and estimation of spillover effects in randomized experiments'. In: *Journal of Econometrics* 237.1 (2023), p. 105237. DOI: https://doi.org/10.1016/j.jeconom.2021.10.014 (cited on page 43).

[10] Guido W. Imbens and Donald B. Rubin. *Causal Inference in Statistics, Social, and Biomedical Sciences*. Cambridge University Press, 2015 (cited on pages 43, 61).

[11] Walter A Orenstein, Roger H Bernier, Timothy J Dondero, Alan R Hinman, James S Marks, Kenneth J Bart, and Barry Sirotkin. *Field evaluation of vaccine efficacy / Walter A. Orenstein ... [et al.]* 1984 (cited on page 49).

[12] Jerome Cornfield. 'A statistical problem arising from retrospective studies'. In: *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 4. University of California Press Berkeley, CA. 1956, pp. 135–148 (cited on page 50).

[13] Winston Lin. 'Agnostic notes on regression adjustments to experimental data: Reexamining Freedman's critique'. In: *Annals of Applied Statistics* 7.1 (2013), pp. 295–318 (cited on page 56).

[14] Yannis Bilias. 'Sequential testing of duration data: The case of the Pennsylvania 'reemployment bonus' experiment'. In: *Journal of Applied Econometrics* 15.6 (2000), pp. 575–594 (cited on page 56).

[15] Philip G. Wright. *The Tariff on Animal and Vegetable Oils*. New York: The Macmillan company, 1928 (cited on page 57).

[16] Sewall Wright. 'Correlation and Causation'. In: *Journal of Agricultural Research* 20.7 (Jan. 1921), pp. 557–585 (cited on page 57).

[17] Judea Pearl. 'Causal diagrams for empirical research'. In: *Biometrika* 82.4 (1995), pp. 669–688 (cited on page 57).

[18] Sander Greenland, Judea Pearl, and James M. Robins. 'Causal diagrams for epidemiologic research'. In: *Epidemiology* 10.1 (1999), pp. 37–48 (cited on page 57).

[19] David R. Cox. *Planning of experiments*. Wiley, 1958 (cited on page 58).

[20] *The Belmont report: Ethical principles and guidelines for the protection of human subjects of research*. Tech. rep. National Commission for the Protection of Human Subjects of Biomedical and Behavioral Research, 1978 (cited on page 59).

[21] Art Owen. 'A First Course in Experimental Design: Notes from Stat 263/363'. Lecture notes. Accessed 1/17/2024. 2020 (cited on page 61).

[22] Esther Duflo, Rachel Glennerster, and Michael Kremer. 'Using randomization in development economics research: A toolkit'. In: *Handbook of Development Economics* 4 (2007), pp. 3895–3962 (cited on page 61).

Predictive Inference via Modern High-Dimensional Linear Regression

3

"Il semble que la perfection soit atteinte non quand il n'y a plus rien à ajouter, mais quand il n'y a plus rien à retrancher."
(perfection is attained not when there is no longer anything to add, but when there is no longer anything to take away.)

– Antoine de Saint-Exupéry [1].

Here we discuss the use of penalized regressions for constructing predictions in high-dimensional settings, particularly when $p > n$. We first motivate the high-dimensional setting as arising both from having a high-dimensional regressor set and from constructing technical regressors from raw regressors. We then discuss Lasso, which penalizes the size of the model by the sum of the absolute value of its coefficients. We conclude with an overview of other penalized regression methods.

3.1 Linear Regression with High-Dimensional Covariates

The Framework

We consider a regression model

$$Y = \beta' X + \epsilon, \quad \epsilon \perp X,$$

where $\beta' X$ is the population best linear predictor of Y using X, or simply the population linear regression function. The vector $X = (X_j)_{j=1}^p$ is p-dimensional. That is, there are p regressors, and

$$p \text{ is large, possibly much larger than } n.$$

This case where p is large relative to the sample size is what we call a *high-dimensional* setting. High-dimensional settings arise when

- data have large dimensional features (i.e. many covariates are available for use as regressors),
- we construct many technical regressors[1] from raw regressors, or
- both.

1: Recall, a *technical regressor* is any variable obtained as a transformation of a basic regressor.

Examples of datasets where many covariates are available and potential corresponding exemplary applications include

- country characteristics in cross-country wealth analysis,
- housing characteristics in house pricing/appraisal analysis,
- individual health information in electronic health records and claims data, and
- product characteristics at the point of purchase in demand analysis.

Another source of high-dimensionality is the use of constructed features or regressors of the form

$$X = T(W) = (T_1(W), ..., T_p(W))',$$

where W denotes original raw regressors. As we discussed in Chapter 1, the set of transformations $T(W)$ is sometimes called the *dictionary* of transformations. Example transformations include polynomials, splines, interactions between variables, and applying functions such as the logarithm or exponential. In the wage analysis in Chapter 1, for example, we used quadratic and cubic transformations of experience, as well as interactions

(products) of these regressors with education and geographic indicators. Recall that the main motivation for the use of constructed regressors is to build *more flexible and potentially better* prediction rules.

The potential for improved prediction arises because we are using prediction rules $\beta'X = \beta'T(W)$ that are *nonlinear* in the original raw regressors W and may thus capture more complex patterns that exist in the data. Conveniently, the prediction rule $\beta'X$ is still linear with respect to the parameters, β, and with respect to the constructed regressors $X = T(W)$, so inherits much from the previous discussion of linear regression provided in Chapter 1.

In summary, we have provided two motivations for using high-dimensional regressors in prediction:

- The first motivation is that modern datasets have high-dimensional features that can be used as regressors.
- The second motivation is that we can use nonlinear transformations of features or raw regressors and their interactions to form constructed regressors. Using transformations allows us to better approximate the best prediction rule – the conditional expectation of the outcome given raw regressors.

Lasso

Recall that we are considering a regression model

$$Y = \beta'X + \epsilon = \sum_{j=1}^{p} \beta_j X_j + \epsilon, \quad \epsilon \perp X \tag{3.1.1}$$

where p is possibly much larger than n.

Classical linear regression or least squares fails in these high-dimensional settings because it *overfits* in finite samples. Intuitively, overfitting refers to using patterns that are idiosyncratic to a specific dataset and do not generalize out of sample. That is, it corresponds to using a prediction rule that is overly complex in that it uses patterns that help explain a given dataset, increasing in-sample measures of fit, but are not present in different data even if the data are drawn from the same population, potentially harming out-of-sample prediction performance.

The potential for classical linear regression estimated with least squares to overfit is especially apparent when $p \geq n$. In this case,

conventional least squares will perfectly fit the data regardless of the value of β as long as the covariate matrix is rank n.[2] We therefore make some assumptions and modify the regression method to deal with cases where p is large.

2: Recall that we illustrated the problem with overfitting in Section 1.2.

An intuitive starting point is the assumption of *approximate sparsity*. Under approximate sparsity, there is a small group of regressors with relatively large coefficients whose use alone suffices to approximate the BLP $\beta'X$ well. The rest of the regressors are assumed to have relatively small coefficients and contribute little to the approximation of the BLP.

An example of approximate sparsity is captured by regression coefficients of the form[3]

$$\beta_j \propto 1/j^2, \quad j = 1, ..., p.$$

3: The notation $\propto$ reads as "proportional to."

Here, the first few coefficients capture almost all the explanatory power of the full vector of coefficients as shown in Figure 3.1.

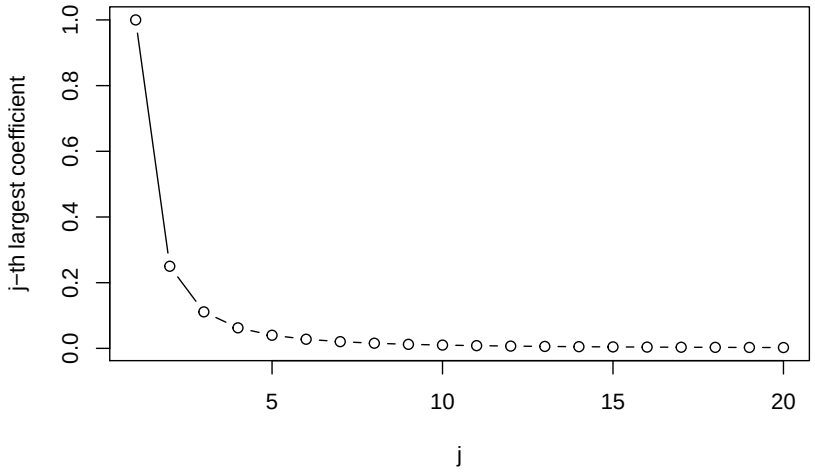


Figure 3.1: Example of regression coefficients, $\beta_j = 1/j^2$ that satisfy approximate sparsity.

Next, we define approximate sparsity formally.

Definition 3.1.1 ***Approximate sparsity:*** *The sorted absolute values of the coefficients decay quickly. Specifically, the j^{th} largest coefficient (in absolute value) denoted by $|\beta|_{(j)}$ obeys*

$$|\beta|_{(j)} \leq Aj^{-a}, \quad a > 1/2, \tag{3.1.2}$$

for each j, where the constants a and A do not depend on the sample size n.

For estimation purposes, we have a random sample $\{(Y_i, X_i)\}_{i=1}^n$. We seek to construct a good linear predictor $\hat{\beta}'X$, which works well when p/n is not small.

Before defining the Lasso problem, it is important to note that we are treating all variables as centered and thus do not include an intercept in the model. In practice, this construction means that, for raw variables Y^* and X^*, we start by defining demeaned versions of these variables $Y = Y^* - \mathbb{E}_n[Y^*]$ and $X = X^* - \mathbb{E}_n[X^*]$ for use in estimation of model parameters.[4] We note that the centered model (3.1.1) is equivalent to starting with the model

$$Y^* = \alpha + \beta' X^* + \epsilon \quad \epsilon \perp X^*$$

with intercept $\alpha = \mathrm{E}[Y^*] - \beta' \mathrm{E}[X^*]$. For estimates $\hat{\beta}$ obtained by estimating (3.1.1), we can thus recover an estimate of α as $\hat{\alpha} = \mathbb{E}_n[Y^*] - \hat{\beta}' \mathbb{E}_n[X^*]$.

A centered random variable U has $\mathrm{E}[U] = 0$, and a centered variable U in a sample has $\mathbb{E}_n[U] = 0$.

4: When performing validation exercises, demeaning and any other transformations that depend on features of the data, such as standardization, should be done in both training and test data using the features of the *training* data rather than of the full sample or the test data.

When discussing theoretical properties, we will further assume that regressors are normalized,

$$\mathrm{E}[X_j^2] = 1.$$

We do state the estimation algorithms without assuming this normalization. The combination of centering and normalization – *standardization* – is commonly employed in practice and is done by default in many software packages.

Lasso constructs $\hat{\beta}$ as the solution of the following penalized least squares problem:

$$\min_{b \in \mathbb{R}^p} \quad \sum_i (Y_i - b' X_i)^2 + \lambda \cdot \sum_{j=1}^{p} |b_j| \hat{\psi}_j, \tag{3.1.3}$$

which is called the Lasso regression problem. The first term is n times the sample mean squared error, and the second term is called a *penalty term*. The penalty term introduces a cost to the complexity of the prospective model where complexity is captured by the sum of the products of the absolute values of the coefficients b_j with the *penalty loadings* $\hat{\psi}_j$ all multiplied by the *penalty level* λ.

Rather than work with centered variables, we could equivalently define (3.1.3) with an intercept where the intercept *does not* enter the penalty function. The important thing to keep in mind is that it is rarely appropriate to penalize the intercept.

The penalty loadings are typically set as

$$\hat{\psi}_j = \sqrt{\mathbb{E}_n[X_j^2]}.$$

The use of this penalty ensures invariance of Lasso predictions to rescaling X_j'. Note that many software packages implement the Lasso with simple penalty loadings $\hat{\psi}_j = 1$. In such cases,

the use of standardized variables produces the same results as using these penalty loadings.

As long as $\lambda > 0$, the introduction of the penalty term in (3.1.3) leads to a prediction rule which is less complex than the rule that would be obtained via solving the unpenalized least squares problem. Specifically, the penalty term in the Lasso problem, $\sum_{j=1}^{p} |b_j| \hat{\psi}_j$, provides a measure of complexity of a regression model in terms of the overall magnitude of the coefficients. When λ is positive, minimizing the Lasso problem requires trading off in-sample fit with this measure of complexity. As a result, the overall magnitude of the estimated coefficients, as measured by the penalty term, will be smaller than the overall magnitude of the coefficients absent this penalty. That is, the Lasso solution will have coefficients that are "shrunk" towards 0 relative to the unpenalized least squares problem.[5]

5: This overall shrinkage towards zero relative to the unpenalized problem is sometimes referred to as *shrinkage bias* or *regularization bias*.

One important benefit of introducing the penalty term is that it helps guard against overfitting by introducing a cost to model complexity. Intuitively, overfitting occurs as a model is made increasingly complex in an effort to make improvements to in-sample fit that are small relative to sampling error and could thus correspond to idiosyncrasies of a specific finite sample. The penalty term imposes a cost to complexity which help keep increases to complexity that have small benefit in terms of improving fit from being made. Through careful choice of λ, we can theoretically guarantee that the Lasso predictor is similar to the optimal predictor, and thus generalizable, even in high-dimensional settings.

A second important feature of Lasso is that it imposes the approximate sparsity condition on the estimated coefficients $\hat{\beta}$. Approximate sparsity is produced because the penalty function in (3.1.3) has a kink at zero which results in the marginal cost of including regressor X_j ($\lambda \hat{\psi}_j > 0$) always being positive when $\lambda > 0$. Therefore, Lasso includes a regressor X_j with non-zero coefficient only if its marginal predictive ability is higher than this marginal cost threshold. That is, Lasso does *variable selection*: The Lasso solution drops any variable (equivalently sets the variable's coefficient to 0) whose marginal predictive benefit does not exceed the marginal cost of inclusion. We illustrate this variable selection property numerically in Example 3.1.1 below.

It is important to note that Lasso will not generally select the "right" set of variables. Lasso will tend to exclude variables with small, but non-zero population coefficients. Lasso will also tend to fail to select the right variables in settings where the

X variables are correlated.[6] That is, one should not conclude that Lasso has selected exactly the variables with non-zero coefficients in the population unless one can rule out variables with small, but non-zero coefficients and ensure that variables are all at most weakly correlated.[7] This failure does not mean that the Lasso predictions are poor quality, but does mean that care should be taken in interpreting the selected variables.

Example 3.1.1 (Simulation Example) Consider

$$Y = \beta' X + \epsilon, \quad X \sim N(0, I_p), \quad \epsilon \sim N(0,1),$$

with approximately sparse regression coefficients:

$$\beta_j = 1/j^2, \quad j = 1, ..., p$$

and

$$n = 300, \quad p = 1000.$$

Figure 3.2 shows that $\hat{\beta}$ is sparse and is close to β. We see that Lasso sets most of regression coefficients to zero. It figures out *approximately* the right set of regressors, including only those with the two largest coefficients. Note that Lasso does not, and in fact cannot, select the regressors with non-zero coefficients in this example as all variables have non-zero coefficients.

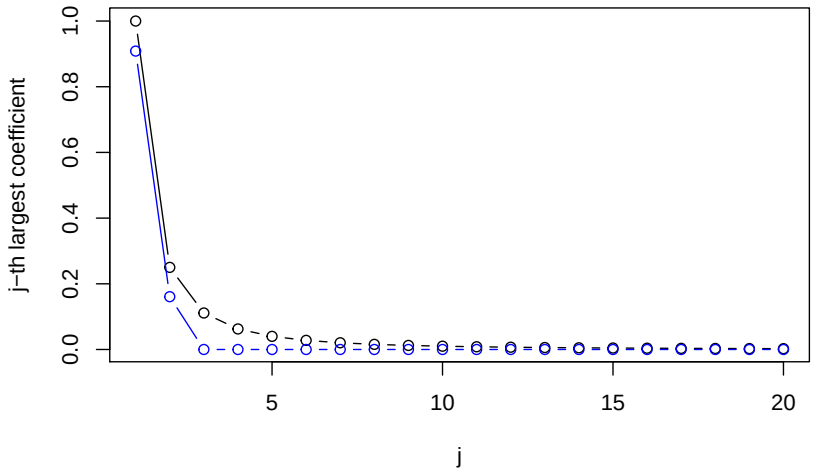


Figure 3.2: The true coefficients (black) vs. coefficients estimated by Lasso (blue) in Example 3.1.1.

A crucial point for the two Lasso properties that we have discussed is the choice of the penalization parameter λ. A theoretically valid choice is[8]

$$\lambda = 2 \cdot c\hat{\sigma}\sqrt{n} z_{1-\mathsf{a}/(2p)} \tag{3.1.4}$$

where $\hat{\sigma} \approx \sigma = \sqrt{\mathrm{E}[\epsilon^2]}$ is obtained via an iteration method defined in Appendix 3.A, $c > 1$, and $1 - \mathsf{a}$ is a confidence

6: For example, consider a scenario where variable X_1 has coefficient $\beta_1 = 0$ but is highly correlated to variables $X_2, ..., X_k$ that have non-zero coefficients. It is quite plausible that the marginal predictive benefit of including X_1 in the model is very high when $X_2, ..., X_k$ are not in the model while the marginal predictive benefit of any one of $X_2, ..., X_k$ is relatively low. In this case, X_1 may enter the Lasso solution with a non-zero coefficient while all of $X_2, .., X_k$ are excluded.

7: This inability to select *exactly* the right regressors is not special to Lasso but shared by all variable selection procedures.

8: Recall that z_t is such that $\mathrm{P}((N(0,1) \leq z_t) = t$.

level.[9] We can further simplify the choice using Feller's tail inequality:

$$z_{1-\mathsf{a}/(2p)} \leq \sqrt{2\log(2p/\mathsf{a})},$$

where the inequality becomes sharp as $p \to \infty$.

This penalty level ensures that the Lasso predictor $\hat{\beta}'X$ does not overfit the data and delivers good predictive performance under approximate sparsity ([2, 3]). Another good way to pick the penalty level when building a model for prediction is by cross-validation ([4]).[10]

9: Practical recommendations, based on theory and that seem to work well in practice, are to set $c = 1.1$ and $\mathsf{a} = .05$.

10: Cross-validation is a repeated data-splitting method for choosing penalty parameters for Lasso and for selecting among predictive models more generally. We outline the basic idea of cross-validation in Section 3.B.

Quick Heuristics for Lasso Properties and Penalty Choice*

Here, we provide a sketch of the mathematics of the Lasso estimator illustrating its variable selection properties and motivating the choice of λ in (3.1.4).

Assume $\hat{\psi}_j = 1$ for simplicity. The j-th component $\hat{\beta}_j$ of the Lasso estimator $\hat{\beta}$ is set to zero if the marginal predictive benefit of changing $\hat{\beta}_j$ away from zero is smaller than the marginal increase in penalty (see Figure 3.3):

$$\hat{\beta}_j = 0 \text{ if } \left| \frac{\partial}{\partial \hat{\beta}_j} \sum_i (Y_i - \hat{\beta}' X_i)^2 \right| < \lambda.$$

That is,

$$\hat{\beta}_j = 0 \text{ if } |-\hat{S}_j| < \lambda, \quad \hat{S}_j = 2\sum_i (Y_i - \hat{\beta}' X_i) X_{ji}.$$

We discuss more detailed heuristics for penalty level selection in the appendix, but the rough idea is that the penalty should dominate the noise $S_j = 2\sum_i (Y_i - \beta' X_i) X_{ji}$ in the measurement of the marginal predictive ability. By the high-dimensional central limit theorem ([5]), we have that

$$(S_j)_{j=1}^p \overset{\mathsf{a}}{\sim} 2\sqrt{n}\sigma (\mathcal{N}_j)_{j=1}^p, \quad \mathcal{N}_j \sim N(0,1).$$

Therefore, to guarantee that Lasso sets to zero the any coefficient whose actual value is zero, we would like to choose λ to dominate

$$2\sqrt{n}\sigma \max_{j=1,\ldots,p} |\mathcal{N}_j|$$

with high probability, say $1 - \mathsf{a}$. Then by the union bound and

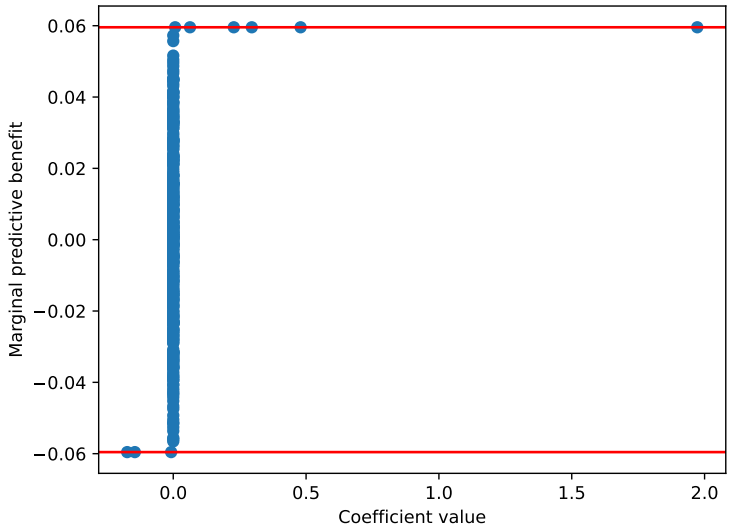


Figure 3.3: Example relationship between coefficient value and (signed) marginal predictive value $\hat{S}_j$ at the optimal solution to the Lasso objective. The red lines correspond to $\{-\lambda, \lambda\}$.

symmetry of centered normal variables,

$$\begin{aligned} &\mathrm{P}\Big(\max_{j=1,\ldots,p} |\mathcal{N}_j| > z_{1-\mathsf{a}/(2p)}\Big) \\ &\leq 2\sum_{j=1}^{p} \mathrm{P}\left(\mathcal{N}_j > z_{1-\mathsf{a}/(2p)}\right) \\ &= 2p\Big(1-(1-\mathsf{a}/(2p))\Big) = \mathsf{a}. \end{aligned}$$

The union bound here is crude, but the bound is not very loose. In particular, when the $\mathcal{N}_j$'s are independent, the bound becomes sharp as $p \to \infty$. Finally, setting

$$\lambda = 2\sigma\sqrt{n} z_{1-\mathsf{a}/(2p)}$$

we conclude that

$$\mathrm{P}(\max_j |S_j| \leq \lambda) \geq 1 - \mathsf{a},$$

up to a vanishing error. That is, this choice of λ guarantees that variables with $\beta_j = 0$ are excluded from the model (have $\hat{\beta}_j = 0$) with high probability.

OLS Post-Lasso

We can use the Lasso-selected set of regressors, those regressors whose Lasso coefficient estimates are non-zero, to refit the model by least squares. This method is called "least squares post Lasso" or simply *Post-Lasso* ([3]). Compared to Lasso,

Post-Lasso undoes the overall shrinkage toward zero relative to unconstrained least squares from the estimated non-zero coefficients, as we illustrate in Figure 3.1.5 below.[11] Removing this shrinkage towards zero from the non-zero coefficients sometimes delivers improvements in predictive performance.

11: Note that the estimates of the large coefficients are nearly perfect after OLS refitting of the model selected by Lasso in this example.

Post-Lasso. We define the Post-Lasso

$$\widetilde{\beta} \in \arg\min_{b \in \mathbb{R}^p} \sum_i (Y_i - b' X_i)^2 \text{ such that } b_j = 0 \text{ if } \hat{\beta}_j = 0 \text{ for each } j, \tag{3.1.5}$$

where $\hat{\beta}$ is the Lasso coefficient estimator. The formal properties of the Post-Lasso estimator $\widetilde{\beta}$ are similar to those of Lasso $\hat{\beta}$; see Section 3.2.

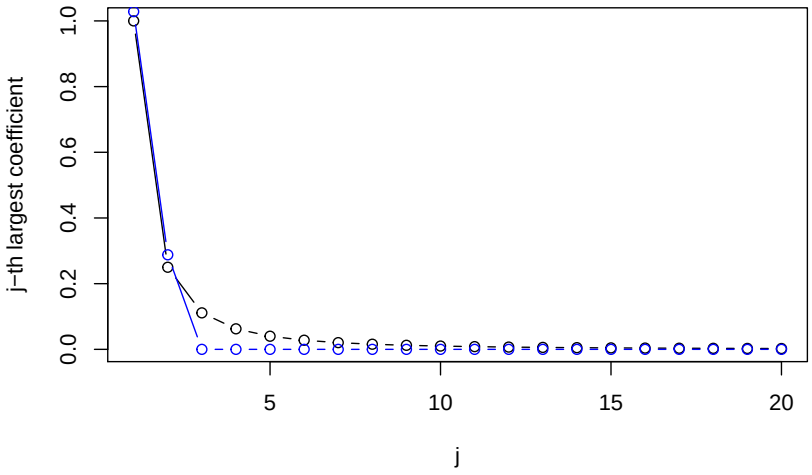


Figure 3.4: The true coefficients (black) vs. coefficients estimated by Post-Lasso (blue) in the Example 3.1.1. Post-Lasso tends to remove regularization bias from the estimated non-zero coefficients.

Remark 3.1.1 (Cross Validation and OLS Post-Lasso) Note that, when using Post-Lasso, one should either use the theoretically justified penalty parameter ([3]) as outlined above or cross-validation for the overall OLS Post-Lasso process. That is, one *should not* apply cross-validation to the Lasso to find a value for λ and then use this same value of λ with Post-Lasso. Unsurprisingly, using a penalty parameter chosen to optimize cross-validation performance for Lasso tends to lead to poor empirical performance when applied to an entirely different procedure, Post-Lasso.

3.2 Predictive Performance of Lasso and Post-Lasso

The best linear prediction rule (out-of-sample) is $\beta'X$. We want to understand the quality of the Lasso prediction rule, $\hat{\beta}'X$. That is,

- Does $\hat{\beta}'X$ provide a good approximation to $\beta'X$?

Recall that with Lasso, we are trying to estimate p parameters $\beta_1, ..., \beta_p$, imposing approximate sparsity via penalization. Under approximate sparsity, only a few, say s, parameters will be "important." We can call s the *effective dimension*. Lasso approximately figures out which parameters are important to keep. Further, intuitively, to estimate each of the "important" s parameters well, we need many observations for each such parameter. This means that n/s must be large, or, equivalently s/n must be small. Using previous reasoning from least squares theory, we might also conjecture that the key determinant of the rate at which Lasso approximates the best linear predictor is $\sqrt{s/n}$. This conjecture is almost correct.

Theorem 3.2.1 *Under approximate sparsity as defined in Definition 3.1.1, restricted isometry conditions stated below, choosing λ as in* (3.1.4)*, and other regularity conditions stated e.g. in [3, 6], with probability approaching $1-\alpha$ as $n \to \infty$, the following bound holds:*

$$\sqrt{\mathrm{E}_X\left[(\beta'X - \hat{\beta}'X)^2\right]} \leq \text{const} \cdot \sqrt{\mathrm{E}[\epsilon^2]}\sqrt{\frac{s\log(\max\{p,n\})}{n}},$$

where E_X denotes expectation with respect to X, and the effective dimension is

$$s = \text{const} \cdot A^{1/a} \cdot n^{\frac{1}{2a}},$$

where constant a is the speed of decay of the sorted coefficient values in the approximate sparsity definition, Definition 3.1.1. Moreover, the number of regressors selected by Lasso is bounded by

$$\text{const} \cdot s$$

with probability approaching $1-\mathsf{a}$ as $n \to \infty$. The constants const *are different in different places and may depend on the distribution of (Y, X) and on* a.

The definition of effective dimension stated in this theorem applies, for instance, under the regularity condition that $\max_{j=1}^{p} \|\mathrm{E}[X_j X]\|_1 \leq \text{const}$; i.e. the sum of the absolute values of every row of the covariance matrix $\mathrm{E}[XX']$ is at most a constant. One can also obtain appropriate notions of effective dimension under weaker assumptions on the covariance matrix. For example, one obtains $s \propto n^{1/(2a-1)}$ if $\max_{j=1}^{p} \|\mathrm{E}[X_j X]\|_2 \leq \text{const}$ or $s \propto n^{1/(2(a-1))}$ if $\max_{j=1}^{p} \|\mathrm{E}[X_j X]\|_\infty \leq \text{const}$ where $\propto$ means "is proportional to."

Therefore, if $s\log(\max\{p,n\})/n$ is small, Lasso and Post-Lasso regression come close to the population regression function/best linear predictor. Relative to our conjectured rate $\sqrt{s/n}$, there

is an additional factor $\sqrt{\log(\max\{p, n\})}$ in the bound. This factor captures the price of not knowing *a priori* which of the p regressors are the s important ones. Lasso approximately finds these important predictors, but correspondingly suffers a loss relative to a predictor estimated with knowledge of the best s-dimensional model ("oracle estimator"). A theoretical guarantee similar to Theorem 3.2.1 has been established for cross-validated Lasso [4], though with number of selected regressors diverging slowly relative to s rather than achieving $s = \text{const} \cdot s$.

Under approximate sparsity and with appropriate choice of penalty parameters, Lasso and Post-Lasso will approximate the best linear predictor well. Theoretically, they will not overfit the data, and we can thus use the sample and adjusted R^2 and MSE to assess out-of-sample predictive performance. Of course, it is always a good idea to verify the out-of-sample predictive performance by using sample splitting.

Remark 3.2.1 (Exact Sparsity) It is helpful to consider the exactly sparse case, in which there are only k non-zero coefficients bounded by some constant and the rest of the coefficients are exactly zero. In this case, the effective dimension is (up to constants) equal to the number of non-zero coefficients, i.e.

$$s = \text{const} \cdot k.$$

To see this, note that β satisfies the approximate sparsity condition with $A = \text{const} \cdot k^a$ for $a \geq 1$, since $\beta_j \leq \text{const} \leq \text{const} \cdot k^a / j^a$ for $j \leq k$ and $\beta_j = 0 \leq \text{const} \cdot k^a / j^a$ for $j > k$. Then $s \leq \text{const} \cdot k n^{1/2a}$, which yields the result as $a \to \infty$.

On regularity conditions*. A sufficient condition under which Theorem 3.2.1 can be established is the restricted isometry condition:

Definition 3.2.1 (Restricted Isometry) *The following conditions hold:*

$$\textit{Uniformly in } Z \subset X : \dim(Z) \leq L = s \log(n),$$

$$\sup_{\|a\|=1} |a'(\mathbb{E}_n[ZZ'] - \mathrm{E}[ZZ'])a| \approx 0,$$

$$0 < C_1 \leq \inf_{\|a\|=1} a'\mathrm{E}[ZZ']a \leq C_2 < \infty,$$

where C_1 and C_2 are constants.

This condition says that "small groups" of regressors are not

collinear and are well-behaved. I.e. we have that subvectors Z of X with dimension $L = s\log(n)$ have empirical Gram matrices $\mathbb{E}_n[ZZ']$ that are close to their population analogues $\mathrm{E}[ZZ']$ in the operator norm and have population covariance matrix $\mathrm{E}[ZZ']$ with eigenvalues bounded away from zero and from above. This condition is simple and intuitive but is stronger than necessary. Results similar to Theorem 3.2.1 have been shown to hold under considerably weaker conditions. The condition $\sup_{\|a\|=1} |a'(\mathbb{E}_n[ZZ'] - \mathrm{E}[ZZ'])a| \approx 0$ has been demonstrated to be valid under various more primitive conditions; see Appendix 3.C.

3.3 A Helicopter Tour of Other Penalized Regression Methods for Prediction

Instead of the Lasso penalty, other penalty schemes can be used, leading to different regression estimators with different properties. These estimators are motivated by different structures for the coefficients on the set of regressors in a high-dimensional model. We consider three important settings where coefficient are sparse, dense, or sparse+dense.

We have already seen that sparse coefficient vectors have a small number of relatively large, non-zero coefficients with the rest of the coefficients being close enough to zero to be ignorable. A dense coefficient vector has the vast majority or all coefficients non-zero and of comparable magnitude. A sparse+dense structure has the vast majority of coefficients being non-zero and of similar magnitude along with a small number of relatively large coefficients. Figure 3.5 illustrates each setting.

Throughout this section, we assume that regressors have been centered and normalized to have second empirical moment equal to 1. We thus ignore coefficient specific penalty parameters like the $\hat{\psi}_j$ in the Lasso problem (3.1.3).

We have already outlined Lasso regression, which performs best in an approximately sparse setting. We next consider the Ridge method, which performs best in the dense setting.

Ridge. The Ridge method estimates coefficients by penalized least squares, where we minimize the sum of squared prediction error plus the penalty term given by the sum of

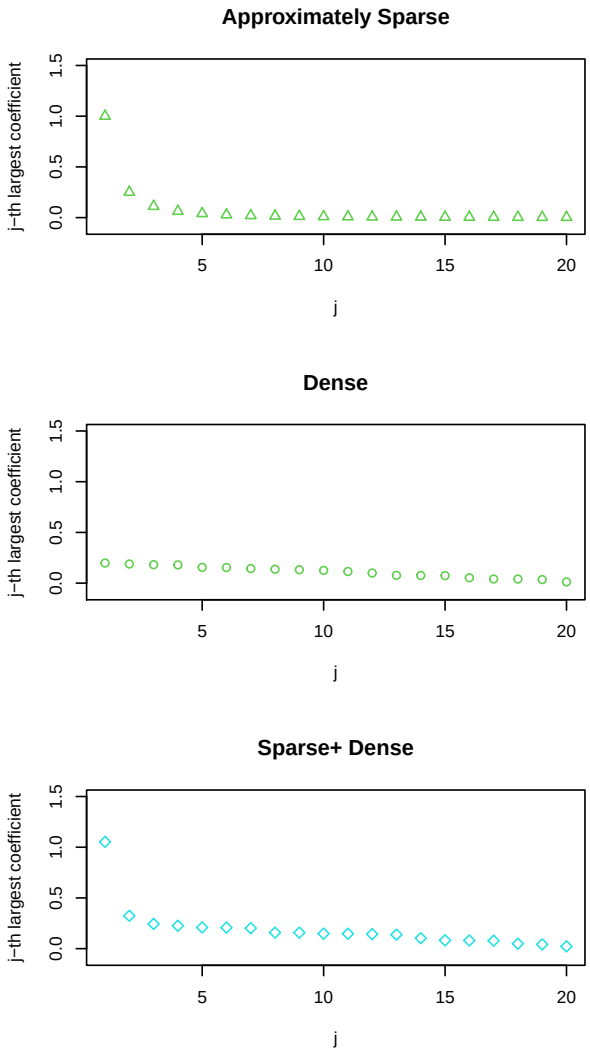


Figure 3.5: The Lasso penalty is best suited for approximately sparse models, and the Ridge penalty for models with small dense coefficients. The Elastic Net can be tuned to perform well with either sparse or dense coefficients. The Lava penalty is best suited for models with coefficients generated as the sum of approximately sparse coefficients and small dense coefficients.

the squared values of the coefficients times a penalty level λ:

$$\hat{\beta}(\lambda) = \arg\min_{b \in \mathbb{R}^p} \sum_{i=1}^{n} (Y_i - b'X_i)^2 + \lambda \sum_j b_j^2.$$

Ridge balances the complexity of the model measured by the sum of squared coefficients with the goodness of in-sample fit. In contrast to Lasso, Ridge penalizes the large values of coefficients much more aggressively and small values much less aggressively – indeed, squaring big values makes them even bigger and squaring small numbers makes them even smaller.

Because of the latter property,

- Ridge does not set estimated coefficients to zero and so it does not do variable selection.
- The Ridge predictor $\hat{\beta}'X$ is especially well suited for prediction in "dense" models, where the β_j's are all small without necessarily being approximately sparse.
- Ridge regression is also well suited when the matrix $\mathrm{E}[XX']$ is poorly behaved, as measured by the decay of its eigenvalues to zero.

In the dense case, the Ridge predictor can easily outperform the Lasso predictor.

Like Ridge, the Lasso predictor empirically seems to have reasonable prediction performance in the presence of ill-behaved design matrices, although we don't understand its theoretical properties well in this case.

Remark 3.3.1 (Theoretical Properties of the Ridge Procedure*) For excellent analysis of Ridge properties, see [7], who present the following bound for the fixed (conditional on) $X_1, \ldots X_n$ case holding with high probability:

$$\mathbb{E}_n\left[(\hat{\beta}'X - \beta'X)^2\right] \lesssim \sum_{j=1}^{p} \frac{\lambda^2 \zeta_j \gamma_j^2}{(\zeta_j^2 + \lambda)^2} + \frac{\mathrm{E}[\epsilon^2]}{n} \sum_{j=1}^{p} \left(\frac{\zeta_j^2}{(\zeta_j + \lambda)^2}\right),$$

where $(\zeta_j)_{j=1}^p$ are eigenvalues of $\mathbb{E}_n[XX']$ and γ_j are such that $\beta = \sum_{j=1}^p \gamma_j c_j$ with c_k being the eigenvectors of $\mathbb{E}_n[XX']$. The theoretically optimal penalty level can be chosen to minimize the right hand side, though doing so is infeasible as the right hand side depends on β. In practice, the penalty level is generally chosen by cross-validation. An analogous result holds for bounding $\mathrm{E}_X\left[(\hat{\beta}'X - \beta'X)^2\right]$ in the case of random $X_1, \ldots X_n$; see [7] for the statement.

The first component on the right hand side can be thought of as squared bias, and the second component is mean squared estimation error. Observe that when $\zeta_j = 1$ and λ is bounded, the second term is of order p/n, which translates to the rate of $\sqrt{p/n}$ after taking the square root. Having the second term go to 0 thus requires $\sqrt{p/n} \to 0$. In contrast, p can be larger than n and the second term can still vanish when eigenvalues

decay to zero. In this case, the effective dimension for a given λ is

$$d(\lambda) = \sum_{j=1}^{p} \frac{\zeta_j^2}{(\zeta_j + \lambda)^2},$$

and the second term is of order $d(\lambda)/n$. The ratio $d(\lambda)/n$ then determines the rate at which the Ridge predictor approximates the optimal predictor if the square bias term is of smaller order. Of course, it is hard to know that the square bias term is of smaller order than the second term in practice. The squared bias term will also not be of small order when there is a large γ_j associated with a large eigenvalue ζ_j.

Remark 3.3.2 (Connection to Principal Components*) Ridge regression is closely related to *principal components regression* which regresses an outcome of the first K principal components of the predictor variables X_i. Principal components provide mutually orthogonal rotations of the original X_i's that maximize fit to the overall design matrix. Here, we consider a case where we have $p < n$ centered predictor variables that are linearly independent. We let P_{ki} denote the i^{th} element of the k^{th} normalized principal component - the principal component divided by it's standard deviation which is given by the k^{th} largest eigenvalue of $\mathbb{E}_n[XX']$, ζ_k. Under these conditions, the ridge prediction can be expressed as

$$X_i'\hat{\beta} = \sum_{k=1}^{p} P_{ki} \frac{\zeta_k}{\zeta_k + \lambda} \mathbb{E}_n[P_k Y].$$

Note that principal components regression using the first K principal components would produce predictions

$$\hat{y}_i = \sum_{k=1}^{K} P_{ki} \mathbb{E}_n[P_k Y].$$

That is, Ridge and principal components regression are tightly connected. Unlike principal components regression, Ridge regression does not pre-select which principal components to use but instead places less weight on low variance principal components according to $\frac{\zeta_k}{\zeta_k+\lambda}$. We find the implicit use of principal components in ridge to be interesting, but note that we can explicitly use principal components as input variables in all penalized methods and in the more advanced methods that we discuss in Chapter 9. We visit using Principal Component Analysis for feature extraction when we outline feature engineering in Chapter 11. For further discussion, see

[8] p. 64-67 or the blog post Ridge vs PCA.

Ridge and Lasso have other useful modifications or hybrids that can perform well in the sparse, dense or sparse + dense settings. One popular modification is the Elastic Net [9] that can perform well in either the sparse or the dense scenario with appropriate tuning.

Elastic Net. The Elastic Net method estimates coefficients by penalized least squares with the penalty given by a linear combination of the Lasso and Ridge penalties:

$$\hat{\beta}(\lambda_1, \lambda_2) = \arg\min_{b \in \mathbb{R}^p} \sum_i (Y_i - b'X_i)^2 + \lambda_1 \sum_j b_j^2 + \lambda_2 \sum_j |b_j|.$$

We see that the penalty function has two penalty levels λ_1 and λ_2, which are chosen by cross-validation in practice.

- By selecting different values of penalty levels λ_1 and λ_2, we have more flexibility with Elastic Net for building a good prediction rule than with just Ridge or Lasso.
- The Elastic Net performs variable selection unless we completely shut down the Lasso penalty by setting $\lambda_2 = 0$.
- With proper tuning, Elastic Net works well in regression models where regression coefficients are either approximately sparse or dense.

See [10] for some theoretical results on Elastic Net.

Another way to combine the Lasso and Ridge penalties is the Lava method, which is intended to work well in sparse+dense settings.

Lava. The Lava method ([11], [12]) estimates coefficients by solving the penalized least squares problem:

$$\hat{\beta}(\lambda_1, \lambda_2) = \arg \min_{b: b = \delta + \xi \in \mathbb{R}^p} \sum_i (Y_i - b'X_i)^2$$

$$+\lambda_1 \sum_j \delta_j^2 + \lambda_2 \sum_j |\xi_j|.$$

Here components of the parameter vector are split into a "dense part" δ_j and "sparse part" ξ_j, where the δ_j's are penalized like in Ridge, and the ξ_j's are penalized like in Lasso. The minimization program automatically determines the best split into the dense and sparse parts. There are two corresponding penalty levels λ_1 and λ_2, which can be chosen by cross-validation in practice.

- Compared to the Elastic Net, the Lava method penalizes large and small coefficients much less aggressively – large coefficients are penalized like Lasso and small coefficients like Ridge. Like Ridge, Lava does not do variable selection.
- Lava is designed to work well in

  "sparse + dense"

  regression models where there are several large coefficients and many small coefficients that do not vanish quickly enough to satisfy approximately sparsity.
- With proper tuning that allows either λ_1 or λ_2 to be set to large values, Lava can also work in either "sparse" or "dense" models.

Theoretical guarantees for these methods are given in [11] and [12]. Theoretically and practically, Lava can significantly outperform Lasso, Ridge and Elastic Net in "sparse+dense" models, and, with appropriate tuning, has comparable performance to Lasso in "sparse" models and to Ridge in "dense" models.

3.4 Choice of Regression Methods in Practice

How should we select the appropriate penalized regression method? The answer is simple if we are interested in building the best prediction. We can split the data into training and testing sets and simply choose the method that performs the best on the test set. Rigorous theoretical guarantees for this approach have been provided by [13].

We show an example of this approach in R Notebook on ML for Prediction of Wages and Python Notebook on ML for Prediction of Wages which illustrate the use of penalized regression methods for predicting log-wages using CPS 2015 data. We can

also use ensemble methods to aggregate prediction methods to get boosts in predictive performance – we describe these aggregation methods in Chapter 9.

Notebooks

- R Notebook on Penalized Regressions and Python Notebook on Penalized Regressions provide details of implementation of different penalized regression methods and examine their performance for approximating regression functions in a simulation experiment. The simulation experiment includes one case with approximate sparsity, one case with dense coefficients, and another case with both approximately sparse and dense components.

- R Notebook on ML for Prediction of Wages and Python Notebook on ML for Prediction of Wages provide details of implementation of different penalized regression methods and examine their performance for predicting log-wages using CPS 2015 data.

Notes

Lasso was introduced by Frank and Friedman [14], and its geometric and computational properties were elaborated on by Tibshirani [15], who also gave it its name. The first general theoretical analysis of Lasso was done by Bickel, Ritov, and Tsybakov [2]. Hastie, Tibshirani, and Wainwright [16] provides a good textbook introduction.

There are many variations on the basic Lasso theme, only some of which we mentioned in this chapter. The properties of the Post-Lasso estimator in approximately sparse models (without assuming that Lasso perfectly selects the "right model") were first established in [3]. The properties of Lasso and Post-Lasso don't hinge on the assumption of Gaussian or sub-Gaussian errors, as proven in [6], though such assumptions are often imposed. Fundamentally, the properties of these procedures rely on a high-dimensional central limit theorem ([5]) that allows Gaussian approximations to key average-like quantities.While cross-validation has been frequently used to select the penalty level, validity of this approach for Lasso was only proven recently – [4]. The Lasso has been extended to clustered dependence by [17] and to time series and many time series by [18], with the corresponding package available at this Link.

There is a large literature on Ridge estimation, with the reference [7] providing what seems to be the state of the art. The Lava approach has been proposed and analyzed in [11] and [12]. [12] also discusses applications to problems with latent confounding and, for this reason, refers to Lava as the spectral deconfounder. We discuss other approaches to dealing with latent confounding in Chapter 12 and Chapter 13.

Study Problems

1. Solve the Lasso optimization problem analytically with only one regressor and interpret the solution.

2. Experiment with the R Notebook on Penalized Regressions, trying out modifications of the Monte-Carlo experiments. As examples, you might change parameters that govern the speed of decay of coefficients to zero, change the error distribution, or alter the structure of dependence among the design variables. Try to explain the results to a fellow student, linking explanations to the theoretical properties of these methods.

3. Experiment with the R Notebook on ML Prediction of Wages. Try to explain the results to a fellow student, linking explanations to the theoretical properties of these methods.

3.A Additional Discussion and Results

Iterative Estimation of σ

The plug-in choice of λ given in equation (3.1.4) requires an estimate of σ. We can estimate σ using the following iterative method. Let X^0 be a small set of regressors (a trivial choice is just the intercept, but we may include, for example, the five regressors that are most strongly correlated with the Y_i's). Let $\widehat{\beta}_0$ be the least squares estimator of the coefficients on the covariates associated with X^0, and define

$$\hat{\sigma}_0 := \sqrt{\mathbb{E}_n[(Y_i - \hat{\beta}_0' X_i^0)^2]}.$$

Set $k = 0$, and specify a small constant $\nu \geq 0$ as a tolerance level and a constant $K > 1$ as an upper bound on the number of iterations:

We find that $K = 1$ works well in practice.

1. Compute the Lasso estimator $\widehat{\beta}$ based on the penalty level λ given in equation (3.1.4) using $\hat{\sigma}_k$.
2. Set $\hat{\sigma}_{k+1} = \sqrt{\mathbb{E}_n[(Y_i - \hat{\beta}' X_i)^2]}$.
3. If $|\hat{\sigma}_{k+1} - \hat{\sigma}_k| \leqslant \nu$ or $k > K$, stop; otherwise set $k \leftarrow k + 1$ and go to (1).

We note that the plug-in choice of λ given in equation (3.1.4) relies on assuming homoskedasticity of the BLP residuals, i.e. $\epsilon \perp\!\!\!\perp X$. This independence implies that $\mathrm{E}[\epsilon^2 X_j^2] = \mathrm{E}[\epsilon^2]\mathrm{E}[X_j^2]$. With independent observations where we do not have $\epsilon \perp\!\!\!\perp X$, we should use penalty loadings $\hat{\psi}_j = \sqrt{\mathbb{E}_n[\hat{\epsilon}^2 X_j^2]}$, where $\hat{\epsilon}_i \approx \epsilon_i$ can be estimated in a similar iterative manner as described above. In this case, we would then take $\hat{\sigma} = 1$ in formula (3.1.4) for λ (see [6] for more details).

We expect the homoskedastic formula for the penalty provided in (3.1.4) will work well in many cases, especially when random variables ϵ, X_j are expected to have fast decaying tail probabilities. For example, when fourth moments of ϵ, X_j are bounded by some constant factor of their second moments, an application of the Cauchy-Schwarz inequality implies that $\mathrm{E}[\epsilon^2 X_j^2] \leq \text{const} \cdot \mathrm{E}[\epsilon^2]\mathrm{E}[X_j^2]$, which is, up to a constant, the simplifying condition implied by homoskedasticity.

Some Lasso Heuristics via Convex Geometry*

Assume $\hat{\psi}_j = 1$ for each j for simplicity, which amounts to normalizing regressors to have the second empirical moment equal to 1. Consider

$$\widehat{\beta} \in \arg\min_{b \in \mathbb{R}^p} \widehat{Q}(b) + \frac{\lambda}{n}\|b\|_1, \tag{3.A.1}$$

where

$$\widehat{Q}(b) = \mathbb{E}_n[(Y_i - b' X_i)^2].$$

The key quantity in the analysis of (3.A.1) is the score – the gradient of $\widehat{Q}$ at the true value:

$$S = -\nabla \widehat{Q}(\beta_0) = 2\mathbb{E}_n[X\epsilon].$$

The score S is the effective "noise" in the problem that should be dominated by the regularization. However, we would like to

make the regularization bias as small as possible. This reasoning suggests choosing the smallest penalty level λ that is just large enough to dominate the noise with high probability, say $1 - \mathsf{a}$, which yields

$$\lambda > c\Lambda, \text{ for } \Lambda := n\|S\|_\infty. \tag{3.A.2}$$

Here, Λ is the maximal score scaled by n, and $c > 1$ is a theoretical constant that guarantees that the score is dominated.

It is useful to mention some simple heuristics for the principle (3.A.2) which arise from considering the simplest case where all of the regressors are irrelevant so that $\beta = 0$. We want our estimator to perform at a near-oracle level in all cases, including this case, but here the oracle estimator β^* sets $\beta^* = \beta = 0$. We thus also want $\widehat{\beta} = \beta = 0$ in this case, at least with a high probability, say $1 - \mathsf{a}$. From the subgradient optimality conditions for (3.A.1), we must have

$$-S_j + \lambda/n > 0 \text{ and } S_j + \lambda/n > 0 \text{ for all } 1 \leq j \leq p \tag{3.A.3}$$

for the Lasso estimator for each coefficient to be exactly 0. We can guarantee (3.A.3) holds by setting the penalty level λ/n such that $\lambda > n \max_{1\leq j\leq p} |S_j| = n\|S\|_\infty$ with probability at least $1 - \mathsf{a}$, which is precisely what the rule (3.A.2) does.

Gaussian approximations to this score motivate the following X-dependent penalty implementation.

Remark 3.A.1 (Refining Penalty Levels) An X-dependent penalty level can be specified as follows:

$$\lambda = c \cdot 2\hat{\sigma}\Lambda(1 - \mathsf{a}|\{X_i\}_{i=1}^n), \tag{3.A.4}$$

where

$$\Lambda(1 - \mathsf{a}|\{X_i\}_{i=1}^n) \\ = (1 - \mathsf{a}) - \text{quantile of } \; n\|\mathbb{E}_n[Xg/\Psi]\|_\infty \mid \{X_i\}_{i=1}^n,$$

g_i are i.i.d. $N(0, 1)$, and $\Psi = \text{diag}(\hat{\psi}_j)_{j=1}^p$. $\Lambda(1 - \mathsf{a}|\{X_i\}_{i=1}^n)$ can be thus be easily approximated by simulation. The use of normal errors g_i could be motivated by assuming the Gaussian errors ϵ_i in the model or by appealing to a high-dimensional central limit theorem. We note that by the union

bound and Feller's tail inequality,

$$\begin{aligned}\Lambda(1-\mathsf{a}|\{X_i\}_{i=1}^n) &\leq \sqrt{n}z_{1-\mathsf{a}/(2p)} \\ &\leq \sqrt{2n\log(2p/\mathsf{a})}.\end{aligned} \tag{3.A.5}$$

Thus, $\sqrt{2n\log(2p/\mathsf{a})}$ provides a simple upper bound on the penalty level.

Refined penalty levels are important when components of X_i are highly correlated, in which case the X-dependent penalty will be much lower that the bounds given in 3.A.5. Using the lower penalty level can offer both practical and theoretical boosts in performance in such cases.

Other Variations on Lasso

Here and below we assume that

$$\hat{\psi}_j = 1, \quad j = 1, \ldots, p$$

to simplify notation. A variant of Lasso, called the *Square-root Lasso* estimator ([19],[20]), is defined as a solution to the following program:

$$\min_{b\in\mathbb{R}^p} \sqrt{\mathbb{E}_n[(Y-b'X)^2]} + \frac{\lambda}{n}\|b\|_1. \tag{3.A.6}$$

Analogously to Lasso, we may set the penalty level as

$$\lambda = c\cdot\widetilde{\Lambda}(1-\mathsf{a}|\{X_i\}_{i=1}^n), \tag{3.A.7}$$

where $c > 1$ and

$$\begin{aligned}&\widetilde{\Lambda}(1-\mathsf{a}|\{X_i\}_{i=1}^n) \\ &\quad = (1-\mathsf{a}) - \text{quantile of } \; n\|\mathbb{E}_n[Xg]\|_\infty/\sqrt{\mathbb{E}_n[g^2]} \mid \{X_i\}_{i=1}^n,\end{aligned}$$

with $g_i \sim N(0,1)$ independent for $i = 1, \ldots, n$. As with Lasso, there is also a simple asymptotic option for setting the penalty level:

$$\lambda = c\cdot 2\sqrt{n}z_{1-\mathsf{a}/(2p)}. \tag{3.A.8}$$

The main attractive feature of (3.A.6) is that the penalty level λ specified above is independent of the value σ. This estimator has statistical performance that is as good as the iterative or cross-validated Lasso. Moreover, the estimator is a solution to a

highly tractable conic programming problem:

$$\min_{t\geq 0, b\in\mathbb{R}^p} t + \frac{\lambda}{n}\|b\|_1 : \sqrt{\mathbb{E}_n[(Y - b'X)^2]} \leq t, \tag{3.A.9}$$

where the criterion function is linear in parameters t and positive and negative components of b, while the constraint can be formulated with a second-order cone, informally known as the "ice-cream cone."

There are several other estimators that make use of penalization by the ℓ_1-norm. A final important case is the *Dantzig selector* estimator [21]. It also relies on ℓ_1-regularization but exploits the notion that the residuals should be nearly uncorrelated with the covariates. The estimator is defined as a solution to

$$\min_{b\in\mathbb{R}^p} \|b\|_1 \ : \ \|\mathbb{E}_n[X(Y - b'X)]\|_\infty \leq \lambda/n. \tag{3.A.10}$$

Again, one may set $\lambda = \sigma\Lambda(1-\mathsf{a}|\{X_i\}_{i=1}^n)$. Here, we focused our discussion on Lasso but virtually all theoretical results carry over to other ℓ_1-regularized estimators including (3.A.6) and (3.A.10). We also refer to [22] for a feasible Dantzig estimator that combines the square-root Lasso method (3.A.9) with the Dantzig method.

3.B Cross-Validation

Cross-validation is a common practical tool that provides a way to choose tuning parameters such as the penalty level in Lasso. The idea of cross-validation is to rely on repeated splitting of the training data to estimate the out-of-sample predictive performance.

Definition 3.B.1 (Cross-Validation in Words)

- *We partition the data into K blocks called "folds." For example, with $K = 5$, we split the data into 5 non-overlapping blocks.*
- *Leave one block out. Fit a prediction rule on all the other blocks. Predict the outcome observations in the left out block, and record the empirical Mean Squared Prediction Error. Repeat this for each block.*
- *Average the empirical Mean Squared Prediction Errors over blocks.*
- *We do these steps for several or many values of the tuning*

parameters and choose the value of the tuning parameter that minimizes the Averaged Mean Squared Prediction Error.

We can also consider many different methods for constructing prediction rules as well. For example, we could try Lasso with many different values of the penalty parameter and Ridge with many different values of the penalty parameter and choose the tuning parameter and method (Lasso or Ridge) that minimizes the cross-validated Mean Squared Prediction Error.

Definition 3.B.2 (Cross-Validation: Formal Description)

- *Randomly partition the observation indices* $1, ..., n$ *into* K *folds* $B_1, ..., B_K$.
- *For each* $k = 1, ..., K$, *fit a prediction rule denoted by* $\hat{f}^{[k]}(\cdot; \theta)$, *where* θ *denotes the tuning parameters such as penalty levels and* $\hat{f}^{[k]}$ *depends only on observations with indices not in the fold* B_k.
- *For each* $k = 1, ..., K$, *the empirical out-of-sample MSE for the block* B_k *is*

$$MSE_k(\theta) = \frac{1}{m_k} \sum_{i \in B_k} (Y_i - \hat{f}^{[k]}(X_i; \theta))^2,$$

  where m_k *is the size of the block* B_k.
- *Compute the cross-validated MSE as*

$$CV\text{-}MSE(\theta) = \frac{1}{K} \sum_{k=1}^{K} MSE_k(\theta).$$

- *Choose the tuning parameter* $\hat{\theta}$ *as a minimizer of* $CV\text{-}MSE(\theta)$.

Remark 3.B.1 (On Guarantees of Cross-Validated Predictors) A common step people do in practice is to retrain the predictor $\hat{f}(X)$ on the entire data with the best tuning parameter $\hat{\theta}$ found by cross-validation. Theoretical properties of the resulting cross-validated predictor $\hat{f}(X)$ are only well understood for some high-dimensional problems. E.g., see [4] for results on Lasso with cross-validation.

Remark 3.B.2 (Guarantees for Pooled Cross-Validated Estimator) On the other hand, there are rigorous theoretical guarantees for the pooled cross-validated predictor:

$$\hat{f}(X) = \frac{1}{K} \sum_{k=1}^{K} \hat{f}^{[k]}(X; \hat{\theta}),$$

which are provided by [23] and [13] who establish that the resulting prediction rule has optimal or near-optimal rates for approximating the best predictor in a given class.

Note that the pooled procedure is different from the default CV procedure implemented in many software packages and used in many applications.

3.C Laws of Large Numbers for Large Matrices*

The following results are useful for justifying the restricted isometry condition for empirical Gram matrices $\mathbb{E}_n[XX']$.

Let s_n, p_n, k_n be sequences of positive constants, $\ell_n = \log(n)$, and C a fixed positive constant. Let $(X_i)_{i=1}^n$ be iid. vectors. Denote by $(Z_i)_{i=1}^n$ corresponding subvectors.

Suppose that $\max_{\|a\|=1} \mathrm{E}[(Z'a)^2] \leq C$ for all $Z \subset X$ such that $\dim(Z) \leq s_n \ell_n$ and that one of the following holds:

(a) X_i is a sub-Gaussian, namely

$$\sup_{\|u\|\leq 1} \mathrm{P}(|X_i'u| > t) \leq 2\exp(-t^2/c_2^2)$$

for all $t \geq 0$, and $s_n(\log n)\left(\log\left(\max\{p_n, n\}\right)\right)/n \to 0$;

(b) X_i has bounded components, namely

$$\max_j |X_{ij}| \leq k_n$$

and $k_n^2 s_n \log^2 n \log(s_n \log n) \log\left(\max\{p_n, n\}\right)/n \to 0$.

Then with probability $1 - \delta_n$

$$\max_{Z\subset X:\dim(Z)\leq s_n\ell_n} \max_{\|a\|=1} |a'\left(\mathbb{E}_n\left[ZZ'\right] - \mathrm{E}\left[ZZ'\right)\right]a| \leq \Delta_n,$$

where (δ_n, Δ_n) are decreasing sequences and $(\delta_n, \Delta_n) \to 0$.

Under (a) the result follows from Theorem 3.2 in [24] and under (b) the result follows from [25]. These references also imply finite-sample characterization of error bounds (δ_n, Δ_n).

3.D A Sketch of the Lasso Guarantee Under Exact Sparsity*

Let us assume that the population BLP β_0 satisfies exact sparsity, i.e. only s out of p coefficients are non-zero. Denote with A the set of non-zero coefficients and with A^c the complement of that set. Since the Lasso minimizes the objective $\hat{Q}(b) + \frac{\lambda}{n}\|b\|_1$ for $\widehat{Q}(b) = \mathbb{E}_n[(Y - b'X)^2]$, we have

$$\hat{Q}(\hat{\beta}) - \hat{Q}(\beta_0) \leq \frac{\lambda}{n}(\|\beta_0\|_1 - \|\hat{\beta}\|_1). \tag{3.D.1}$$

Let $\nu := \hat{\beta} - \beta_0$. Since the objective $\hat{Q}(\beta)$ is convex in β, we have by an application of the Cauchy-Schwarz inequality that

$$\hat{Q}(\hat{\beta}) - \hat{Q}(\beta_0) \geq \nabla\hat{Q}(\beta_0)'\nu = -S'\nu \geq -\|S\|_\infty\|\nu\|_1$$

for $S = -\nabla\widehat{Q}(\beta_0) = 2\mathbb{E}_n[X\epsilon]$.

We will assume that λ is chosen such that we have $\frac{\lambda}{n} \geq 2\|S\|_\infty$ with probability $1 - \mathsf{a}$.[12] We focus then on the good event where the above inequality is satisfied. Then we can combine the above two inequalities:

$$\frac{\lambda}{n}(\|\beta_0\|_1 - \|\hat{\beta}\|_1) \geq -\|S\|_\infty\|\nu\|_1 \geq -\frac{\lambda}{2n}\|\nu\|_1.$$

Hence with with high probability,

$$\hat{\beta} - \beta_0 \in RC = \{\nu : \|\beta_0 + \nu\|_1 \leq \|\beta_0\|_1 + \|\nu\|_1/2\}.$$

Note also that $\nu \in RC$ implies[13]

$$\|\nu_{A^c}\|_1 \leq 3\|\nu_A\|_1 \tag{3.D.2}$$

where ν_A denotes the entries from ν in A and ν_{A^c} denotes the entries of ν in A^c. This inequality roughly states that the error vector $\nu = \hat{\beta} - \beta_0$ is primarily supported on A.

We impose the following regularity condition:

$$0 < C_1 \leq \min_{\nu \in RC \setminus 0} \frac{\nu'\mathrm{E}[XX']\nu}{\|\nu\|^2} \leq C_2 < \infty. \tag{3.D.3}$$

The restricted isometry conditions we impose in the text are known to imply this condition.[14]

Suppose that we can argue that we have, for any vector $\nu \in$

12: The High-Dimensional CLT bounds tell us that if we set $\lambda \approx \sqrt{n\log(\max\{p/\mathsf{a}, n\})}$, then this inequality holds with probability $1 - \mathsf{a}$.

13: Verify this as a reading exercise.

14: See, e.g. Lemma 10 in [26] for an argument based on [2].

RC,

$$\nu' \mathbb{E}_n[XX']\nu \geq \hat{C}_1 \|\nu\|_2^2 \tag{3.D.4}$$

for some $\hat{C}_1 > 0$ that will be generally be related to C_1 and features of the population. (3.D.4) is oftentimes referred to as the empirical Restricted Strong Convexity (RSC) property. We provide an example and the corresponding $\hat{C}_1$ below.

Then, using the fact that $\hat{Q}(\beta)$ is quadratic in β, we can invoke the exact second order Taylor expansion:

$$\hat{Q}(\hat{\beta}) - \hat{Q}(\beta_0) = S'\nu + \nu' \mathbb{E}_n[XX']\nu \geq -\|S\|_\infty \|\nu\|_1 + \hat{C}_1 \|\nu\|_2^2.$$

When combined with the upper bound from the optimality of $\hat{\beta}$ for the penalized empirical loss and the fact that $\frac{\lambda}{n} \geq 2\|S\|_\infty$, this expansion yields

$$\frac{\lambda}{n}\|\nu\|_1 \geq \hat{Q}(\hat{\beta}) - \hat{Q}(\beta_0) \geq -\frac{\lambda}{2n}\|\nu\|_1 + \hat{C}_1 \|\nu\|_2^2.$$

The second crucial inequality that

$$\|\nu\|_2^2 \leq \frac{3\lambda}{2\hat{C}_1 n}\|\nu\|_1 \tag{3.D.5}$$

then follows.

Finally, note that for any vector ν that is primarily supported on A, the ℓ_2 and ℓ_1 norms are within a factor $\approx \sqrt{s}$ of each other:

$$\|\nu\|_1 = \|\nu_A\|_1 + \|\nu_{A^c}\|_1 \leq 4\|\nu_A\|_1 \leq 4\sqrt{s}\|\nu_A\|_2 \leq 4\sqrt{s}\|\nu\|_2$$

where we used the norm inequality, that for an s-dimensional vector v, we have $\|v\|_1 \leq \sqrt{s}\|v\|_2$. Thus, we can conclude

$$\|\nu\|_2 \leq \frac{6\lambda}{\hat{C}_1 n}\sqrt{s}. \tag{3.D.6}$$

Using the assumption that $\nu' \mathrm{E}[XX']\nu \leq C_2\|\nu\|_2$ for $\nu \in RC$, we get the final bound:

$$\sqrt{\mathrm{E}_X[(X'\hat{\beta} - X'\beta_0)^2]} = \sqrt{\nu' \mathrm{E}[XX']\nu} \leq C_2\|\nu\|_2 \leq \frac{6\lambda\, C_2}{\hat{C}_1 n}\sqrt{s}.$$

It remains to argue the empirical RSC property. Note that if

$$\|\mathbb{E}_n[XX'] - \mathrm{E}[XX']\|_\infty \leq \mu_n$$

with probability approaching 1,[15] then we have

15: Application of the high-dimensional CLT implies that we can take $\mu_n \propto \sqrt{\frac{\log(\max\{p,n\})}{n}}$.

$$\begin{aligned} v'\mathbb{E}_n[XX']v &\geq v'\mathrm{E}[XX']v - \|v\|_1^2 \|\mathbb{E}_n[XX'] - \mathrm{E}[XX']\|_\infty \\ &\geq (C_1 - 16s\mu_n)\|v\|_2^2 \end{aligned}$$

by Condition (3.D.3) and an application of the Hölder inequality. Thus, if n is large enough such that $16s\mu_n \leq \frac{C_1}{2}$, we conclude that the empirical RSC condition holds with $\hat{C}_1 = \frac{C_1}{2}$.

Bibliography

[1] Antoine de Saint-Exupéry. *Terre des hommes*. Gallimard, 1939 (cited on page 69).

[2] Peter J. Bickel, Ya'acov Ritov, and Alexandre B. Tsybakov. 'Simultaneous analysis of Lasso and Dantzig selector'. In: *Annals of Statistics* 37.4 (2009), pp. 1705–1732 (cited on pages 76, 87, 95).

[3] Alexandre Belloni and Victor Chernozhukov. 'Least Squares After Model Selection in High-dimensional Sparse Models'. In: *Bernoulli* 19.2 (2013). ArXiv, 2009, pp. 521–547 (cited on pages 76–79, 87).

[4] Denis Chetverikov, Zhipeng Liao, and Victor Chernozhukov. 'On cross-validated lasso in high dimensions'. In: *Annals of Statistics* 49.3 (2021), pp. 1300–1317 (cited on pages 76, 80, 87, 93).

[5] Victor Chernozhukov, Denis Chetverikov, and Kengo Kato. 'Central Limit Theorems and Bootstrap in High Dimensions'. In: *Annals of Probability* 45.4 (2017), pp. 2309–2352 (cited on pages 76, 87).

[6] Alexandre Belloni, Daniel L. Chen, Victor Chernozhukov, and Christian B. Hansen. 'Sparse Models and Methods for Optimal Instruments with an Application to Eminent Domain'. In: *Econometrica* 80.6 (2012). Arxiv, 2010, pp. 2369–2429 (cited on pages 79, 87, 89).

[7] Daniel Hsu, Sham M. Kakade, and Tong Zhang. 'Random design analysis of ridge regression'. In: *Conference on Learning Theory*. Vol. 23. JMLR Workshop and Conference Proceedings. 2012, pp. 9.1–9.24 (cited on pages 83, 88).

[8] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The Elements of Statistical Learning*. Vol. 1. Springer series in statistics New York, 2001 (cited on page 85).

[9] Hui Zou and Trevor Hastie. 'Regularization and variable selection via the elastic net'. In: *Journal of the Royal Statistical Society: Series B* 67.2 (2005), pp. 301–320 (cited on page 85).

[10] Christine De Mol, Ernesto De Vito, and Lorenzo Rosasco. 'Elastic-net regularization in learning theory'. In: *Journal of Complexity* 25.2 (2009), pp. 201–230. DOI: https://doi.org/10.1016/j.jco.2009.01.002 (cited on page 85).

[11] Victor Chernozhukov, Christian Hansen, and Yuan Liao. 'A lava attack on the recovery of sums of dense and sparse signals'. In: *Annals of Statistics* 45.1 (2017), pp. 39–76 (cited on pages 85, 86, 88).

[12] Domagoj Ćevid, Peter Bühlmann, and Nicolai Meinshausen. 'Spectral deconfounding via perturbed sparse linear models'. In: *Journal of Machine Learning Research* 21 (2020), pp. 1–41 (cited on pages 85, 86, 88).

[13] Marten Wegkamp. 'Model selection in nonparametric regression'. In: *Annals of Statistics* 31.1 (2003), pp. 252–273 (cited on pages 86, 94).

[14] lldiko E. Frank and Jerome H. Friedman. 'A statistical view of some chemometrics regression tools'. In: *Technometrics* 35.2 (1993), pp. 109–135 (cited on page 87).

[15] Robert Tibshirani. 'Regression shrinkage and selection via the Lasso'. In: *Journal of the Royal Statistical Society: Series B* 58.1 (1996), pp. 267–288 (cited on page 87).

[16] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman & Hall/CRC, 2015 (cited on page 87).

[17] Alexandre Belloni, Victor Chernozhukov, Christian Hansen, and Damian Kozbur. 'Inference in High-Dimensional Panel Models With an Application to Gun Control'. In: *Journal of Business & Economic Statistics* 34.4 (2016), pp. 590–605 (cited on page 87).

[18] Victor Chernozhukov, Wolfgang Karl Härdle, Chen Huang, and Weining Wang. 'Lasso-driven inference in time and space'. In: *Annals of Statistics* 49.3 (2021), pp. 1702–1735 (cited on page 87).

[19] Alexandre Belloni, Victor Chernozhukov, and Lie Wang. 'Square-root lasso: pivotal recovery of sparse signals via conic programming'. In: *Biometrika* 98.4 (2011). Arxiv, 2010, pp. 791–806 (cited on page 91).

[20] Alexandre Belloni, Victor Chernozhukov, and Lie Wang. 'Pivotal estimation via square-root lasso in nonparametric regression'. In: *Annals of Statistics* 42.2 (2014), pp. 757–788 (cited on page 91).

[21] Emmanuel Candès and Terence Tao. 'The Dantzig selector: statistical estimation when p is much larger than n'. In: *Annals of Statistics* 35.6 (2007), pp. 2313–2351 (cited on page 92).

[22] Eric Gautier and Alexander B. Tsybakov. 'High-Dimensional Instrumental Variables Regression and Confidence Sets'. In: *ArXiv working report* (2011) (cited on page 92).

[23] Guillaume Lecué and Charles Mitchell. 'Oracle inequalities for cross-validation type procedures'. In: *Electronic Journal of Statistics* 6 (2012), pp. 1803–1837 (cited on page 94).

[24] M. Rudelson and S. Zhou. 'Reconstruction from anisotropic random measurements'. In: *ArXiv:1106.1151* (2011) (cited on page 94).

[25] Mark Rudelson and Roman Vershynin. 'On sparse reconstruction from Fourier and Gaussian measurements'. In: *Communications on Pure and Applied Mathematics* 61.8 (2008) (cited on page 94).

[26] Alexandre Belloni and Victor Chernozhukov. 'High Dimensional Sparse Econometric Models: An Introduction'. In: *Inverse Problems and High-Dimensional Estimation: Stats in the Château Summer School, August 31 - September 4, 2009*. Ed. by Pierre Alquier, Eric Gautier, and Gilles Stoltz. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 121–156. DOI: 10.1007/978-3-642-19989-9_3 (cited on page 95).

Statistical Inference on Predictive and Causal Effects in High-Dimensional Linear Regression Models

4

"The partial trend regression method can never, indeed, achieve anything which the individual trend method cannot, because the two methods lead by definition to identically the same results."
(An in-words restatement of the FWL theorem.)

– Ragnar Frisch and Frederick V. Waugh [1].

Here we discuss inference on predictive effects using Double Lasso methods, where we use Lasso (at least) twice to residualize outcomes and a target covariate of interest whose predictive effect we'd like to infer. Double Lasso methods rely on the approximate sparsity of the best linear predictors for the outcome and for the target covariate. The resulting estimator concentrates in a $1/\sqrt{n}$ neighborhood of the true value and is approximately Gaussian, enabling the construction of confidence bands. We explain the low bias property of the Double Lasso method using Neyman orthogonality, and isolate the latter as a critical property for further generalizations.

4.1 Introduction

We recall the predictive effect question:[1]

- How does the predicted value of Y change if a regressor D increases by a unit, while other regressors W remain unchanged?

As before, we denote the set of regressors as $X = (D, W)$. In Chapter 1, we discussed how we could use the population regression coefficient corresponding to the variable D, denoted α, to answer this question. We also discussed how to estimate this effect and construct confidence intervals with regression. Now we turn to estimation and construction of confidence intervals for α in the high-dimensional setting, using the tools we developed in Chapter 3.

Here we focus on using Lasso methods. We can use other penalized methods with the caveat that theoretical guarantees are not available unless we perform additional data splitting. We will discuss the use of data splitting and more general machine learning methods in detail when we introduce "double machine learning" or "debiased machine learning" in Chapter 10.

1: We discuss assumptions and modeling frameworks under which the predictive effect question has a causal interpretation in detail in Chapter 5 through Chapter 8. Under the framework developed in those chapters, the tools in this chapter offer one approach to performing statistical inference for causal effects. Here, we simply note that we may be interested in providing statistical inference for predictive effects regardless of whether they have a causal interpretation.

4.2 Inference with Double Lasso

Inference on One Coefficient

The key to inference will be the application of Frisch-Waugh-Lovell partialling-out. Consider the simple predictive model:

$$Y = \alpha D + \beta' W + \epsilon, \tag{4.2.1}$$

where D is the target regressor and W consists of p controls. After partialling-out W,

$$\tilde{Y} = \alpha \tilde{D} + \epsilon, \quad \mathrm{E}[\epsilon \tilde{D}] = 0, \tag{4.2.2}$$

where the variables with tildes are residuals retrieved from taking out the linear effect of W (practically, via linear regression):

$$\tilde{Y} = Y - \gamma_{YW}' W, \quad \gamma_{YW} \in \arg\min_{\gamma \in \mathbb{R}^p} \mathrm{E}[(Y - \gamma' W)^2],$$

$$\tilde{D} = D - \gamma_{DW}' W, \quad \gamma_{DW} \in \arg\min_{\gamma \in \mathbb{R}^p} \mathrm{E}[(D - \gamma' W)^2].$$

α can then be recovered from population linear regression of $\tilde{Y}$ on $\tilde{D}$:

$$\alpha = \arg\min_{a \in \mathbb{R}} \mathrm{E}[(\tilde{Y} - a\tilde{D})^2] = (\mathrm{E}[\tilde{D}^2])^{-1}\mathrm{E}[\tilde{D}\tilde{Y}].$$

Note also that $a = \alpha$ solves the moment equation:

$$\mathrm{E}[(\tilde{Y} - a\tilde{D})\tilde{D}] = 0.$$

We now consider estimation of α in a high-dimensional setting. For estimation purposes, we maintain that we have a random sample $\{(Y_i, X_i)\}_{i=1}^n$ where $X_i = (D_i, W_i)$.

To estimate α, we will mimic the partialling-out procedure in the population in the sample. In Chapter 1, where p/n was small, we employed ordinary least squares as the prediction method in the partialling-out steps. We are now considering cases where p/n is not small, and we instead employ Lasso-based methods in the partialling-out steps.

The estimation procedure for a target parameter α in a high-dimensional linear model setting can be summarized as follows:

The Double Lasso procedure:

1. We run Lasso regressions of Y_i on W_i and D_i on W_i

$$\hat{\gamma}_{YW} = \arg\min_{\gamma \in \mathbb{R}^p} \quad \sum_i (Y_i - \gamma' W_i)^2 + \lambda_1 \sum_j \hat{\psi}_j^Y |\gamma_j|,$$

$$\hat{\gamma}_{DW} = \arg\min_{\gamma \in \mathbb{R}^p} \quad \sum_i (D_i - \gamma' W_i)^2 + \lambda_2 \sum_j \hat{\psi}_j^D |\gamma_j|,$$

   and obtain the resulting residuals:

$$\check{Y}_i = Y_i - \hat{\gamma}'_{YW} W_i,$$

$$\check{D}_i = D_i - \hat{\gamma}'_{DW} W_i.$$

   In place of Lasso, we can use Post-Lasso or other Lasso relatives (the Dantzig selector, square-root Lasso, and others).

2. We run the least squares regression of $\check{Y}_i$ on $\check{D}_i$ to

obtain the estimator $\hat{\alpha}$:

$$\begin{aligned}\hat{\alpha} &= \arg\min_{a\in\mathbb{R}} \mathbb{E}_n[(\check{Y} - a\check{D})^2] \\ &= (\mathbb{E}_n[\check{D}^2])^{-1}\mathbb{E}_n[\check{D}\check{Y}].\end{aligned} \tag{4.2.3}$$

We can use standard results from this regression, ignoring that the input variables were previously estimated, to perform inference about the predictive effect, α.

Good performance of the Double Lasso procedure relies on approximate sparsity of the population regression coefficients γ_{YW} and γ_{DW}, with a sufficiently high speed of decrease in the sorted coefficients and on careful choice of the Lasso tuning parameters. For approximate sparsity, we will impose that the sorted coefficients satisfy

$$|\gamma_{YW}|_{(j)} \leq Aj^{-a} \text{ and } \quad |\gamma_{DW}|_{(j)} \leq Aj^{-a}$$

for $a > 1$ and $j = 1, \ldots, p$.[2] Under these sparsity conditions, we can use the plug-in rule outlined in Chapter 3 for choosing λ_1 and λ_2. Importantly, using these tuning parameters theoretically guarantees that we produce high quality prediction rules for D and Y while simultaneously avoiding overfitting under approximate sparsity. Absent these guarantees, we cannot theoretically ensure that first step estimation of $\check{D}$ and $\check{Y}$ does not have first-order impacts on the final estimator $\hat{\alpha}$. Practically, we have found that Lasso with penalty parameter selected via cross-validation can perform poorly in simulations in moderately sized samples. We return to this issue in Chapter 10 where we discuss a method that allows the use of complex machine learners, including Lasso and other regularized estimators, and data-driven tuning (e.g. cross-validation).

2: Note that in this case the effective dimension s of the problem is $s \approx A^{1/a}n^{1/2a} \ll n^{1/2}$. Intuitively, the effective number of non-zero coefficients grows slower than $\sqrt{n}$.

The following theorem can be shown for the Double Lasso procedure:

Theorem 4.2.1 (Adaptive Inference with Double Lasso in High-Dimensional Regression) *Under the stated approximate sparsity, the conditions required for Theorem 3.2.1 (e.g. restricted isometry), and additional regularity conditions, the estimation error in $\check{D}_i$ and $\check{Y}_i$ has no first order effect on $\hat{\alpha}$, and*

$$\sqrt{n}(\hat{\alpha} - \alpha) \approx \sqrt{n}\mathbb{E}_n[\tilde{D}\epsilon]/\mathbb{E}_n[\tilde{D}^2] \overset{a}{\sim} N(0, \mathsf{V}),$$

where

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

The above statement means that $\hat{\alpha}$ concentrates in a $\sqrt{\mathsf{V}/n}$-neighborhood of α, with deviations controlled by the normal law. Observe that the approximate behavior of the Double Lasso estimator is the same as the approximate behavior of the least squares estimator in low-dimensional models; see Theorem 1.3.2 in Chapter 1.

Just like in the low-dimensional case, we can use these results to construct a confidence interval for α. The standard error of $\hat{\alpha}$ is

$$\sqrt{\hat{\mathsf{V}}/n},$$

where $\hat{\mathsf{V}}$ is a plug-in estimator of V. The result implies, for example, that the interval

$$[\hat{\alpha} \pm 1.96\sqrt{\hat{\mathsf{V}}/n}]$$

covers α about 95% of the time.

Application to Testing the Convergence Hypothesis

R Notebook on Double Lasso for Growth Convergence and Python Notebook on Double Lasso for Growth Convergence provides code for the convergence hypothesis example.

We provide an empirical example of partialling-out with Lasso to estimate the regression coefficient α in the high-dimensional linear regression model:

$$Y = \alpha D + \beta' W + \epsilon.$$

In this example, we are interested in how economic growth rates (Y) are related to the initial wealth levels in each country (D) controlling for a country's institutional, educational, and other similar characteristics (W).

The relationship is captured by α, the "speed of convergence/-divergence," which predicts the speed at which poor countries catch up ($\alpha < 0$) or fall behind ($\alpha > 0$) rich countries, after controlling for W. Here, we are interested in understanding if poor countries grow faster than rich countries, controlling for educational and other characteristics. In other words, is the speed of convergence negative: Is $\alpha < 0$?

$\alpha < 0$ corresponds to the Convergence Hypothesis predicted by the Solow growth model. Robert M. Solow is a world-renowned MIT economist who won the Nobel Prize in Economics in 1987.

In our data, the outcome (Y) is the realized annual growth rate of a country's wealth (Gross Domestic Product per capita). The target regressor (D) is the initial level of the country's

wealth. The controls (W) include measures of education levels, quality of institutions, trade openness, and political stability in the country. The sample, which is based on the Barro-Lee data set [2], contains 90 countries and about 60 controls. Thus $p \approx 60, n = 90$ and p/n is not small. We expect the least squares method to provide a poor/ noisy estimate of α. We expect the method based on partialling-out with Lasso to provide a high-quality estimate of α.

	Estimate	Std. Error	95% CI
OLS	-0.009	0.032	[-0.073, 0.054]
Double Lasso	-0.045	0.018	[-0.080, -0.010]

Table 4.1: Estimates for the convergence coefficient. We report specification robust standard errors with finite sample correction, i.e., "HC1."

Least squares provides a rather noisy estimate of convergence speed, which does not allow drawing strong conclusions about the convergence hypothesis. For example, the 95% confidence interval is wide and includes both positive and negative values. Given that p/n is not small in this example, we should also be highly skeptical of the OLS results and especially the standard error. For example, [3] show that conventional robust standard errors are not even consistent in linear models when p/n is not small. In sharp contrast, Double Lasso provides a precise estimate for which we can obtain theoretically justified inferential statements even though p/n is not close to 0. The Lasso-based point estimate is -4.5% and the 95% confidence interval for the (annual) convergence rate is -8% to -1%. This empirical evidence is consistent with the conditional convergence hypothesis.

4.3 Why Partialling-out Works: Neyman Orthogonality

Neyman Orthogonality

In the Double Lasso approach, α is the target parameter and η are *nuisance* projection *parameters*[3] with true value

$$\eta^o = (\gamma'_{DW}, \gamma'_{YW})'.$$

As the learned value $\hat{\alpha}$ of α depends on the values of the nuisance parameters, it is useful to explicitly consider the dependence of $\hat{\alpha}$ on the nuisance parameters:

$$\hat{\alpha}(\eta).$$

3: *Nuisance parameters* refer to parameters that must be learned or otherwise adjusted for in order to learn the parameter of interest but are not of direct interest themselves. That is, they are nuisances - we'd like to ignore them if we could.

For the majority of the estimation processes we will describe in this book, we can construct a population analogue

$$\alpha(\eta)$$

of the estimator $\hat{\alpha}(\eta)$, such that the in-sample estimation procedure converges to it, in a formal sense.

For instance, the Double Lasso process constructs the residuals

$$\check{Y}_i(\eta) = Y_i - \eta_1' W_i, \quad \check{D}_i(\eta) = D_i - \eta_2' W_i$$

and then obtains $\hat{\alpha}(\eta)$ as the solution to the empirical estimating equation

$$\widehat{\mathsf{M}}(a, \eta) := \mathbb{E}_n[(\check{Y}(\eta) - a\check{D}(\eta))\check{D}(\eta)] = 0.$$

This process implicitly defines the function $\hat{\alpha}(\eta)$. We can think of the population analog of this process, where we construct the residuals

$$\tilde{Y}(\eta) = Y - \eta_1' W, \quad \tilde{D}(\eta) = D - \eta_2' W$$

and solve the population moment equation

$$\mathsf{M}(a, \eta) := \mathrm{E}[(\tilde{Y}(\eta) - a\tilde{D}(\eta))\tilde{D}(\eta)] = 0, \tag{4.3.1}$$

which again implicitly defines the function $\alpha(\eta)$.

The main idea of the Double Lasso approach is that, in the population limit, it corresponds to a procedure for learning the target parameter α that is first-order insensitive to local perturbations of the nuisance parameters around their true values, η^o:

$$\partial_\eta \alpha(\eta^o) = 0. \tag{4.3.2}$$

Formally, we use ∂_η to denote the Gateaux derivative. See Remark 10.4.2 in Chapter 10 for more details.

We will call the local insensitivity of target parameters to nuisance parameters as in (4.3.2) Neyman orthogonality of the estimation process.

Neyman orthogonality is important for providing high-quality estimation and inference, especially in high-dimensional settings. In high-dimensional settings, we use regularization procedures to estimate the nuisance parameters as solutions to suitable prediction problems. The use of regularization generally results in bias, and we may heuristically view using regularized estimates of nuisance parameters as plugging in estimates of these parameters that are close to, but not exactly equal to, the true values of the nuisance parameters η^o. Neyman

orthogonality, which guarantees that the target parameter is locally insensitive to perturbations of the nuisance parameters around their true values, then ensures that this bias does not transmit to the estimation of the target parameter, at least to the first order.

Let us prove the claim $\partial_\eta \alpha(\eta^o) = 0$ for the Double Lasso process. Since the function $\alpha(\eta)$ is implicitly defined as the solution to the equation $\mathsf{M}(a, \eta) = 0$, by the implicit function theorem and letting $\alpha = \alpha(\eta^o)$:

$$\partial_\eta \alpha(\eta^o) = -\partial_a \mathsf{M}(\alpha, \eta^o)^{-1} \partial_\eta \mathsf{M}(\alpha, \eta^o).$$

Here

$$\partial_\eta \mathsf{M}(\alpha, \eta^o)$$

consists of two components

$$\partial_{\eta_1} \mathsf{M}(\alpha, \eta^o) = \mathrm{E}[W \tilde{D}(\eta^o)] = \mathrm{E}[W(D - \gamma'_{DW} W)] = 0$$

and

$$\begin{aligned} \partial_{\eta_2} \mathsf{M}(\alpha, \eta^o) = & -\mathrm{E}[W \tilde{Y}(\eta^o)] + 2\mathrm{E}[\alpha W \tilde{D}(\eta^o)] \\ = & -\mathrm{E}[W(Y - \gamma'_{YW} W)] + 2\mathrm{E}[\alpha W(D - \gamma'_{DW} W)] = 0. \end{aligned}$$

We summarize the discussion as follows:

Neyman Orthogonality. The parameter of interest α that depends on nuisance parameters η with true value η^o is Neyman orthogonal with respect to these parameters if

$$\partial_\eta \alpha(\eta^o) = 0.$$

If the parameter α is defined as a root in a of the equation $\mathsf{M}(a, \eta) = 0$, which depends on the nuisance parameters η with true value η^o, then the equation is Neyman orthogonal if

$$\partial_\eta \mathsf{M}(\alpha, \eta^o) = 0.$$

The principle is applicable to problems outside the high-dimensional linear model problem considered in this chapter.

What Happens if We Don't Have Neyman Orthogonality?

If we don't have Neyman orthogonality, we should not expect to get high-quality estimates of the target parameters. For example, a seemingly sensible approach that one might consider for statistical inference in the high-dimensional linear model context is as follows:

> **(Invalid) Single Selection/Naive Method.**
> In this invalid method, one applies Lasso regression of Y on D and W to select relevant covariates W_Y, in addition to the covariate of interest, then refits the model by least squares of Y on D and W_Y. Inference for the target parameter is then carried out using conventional inference based on the latter regression.

Despite its simplicity and seeming intuitive appeal, the approach outlined above is not a valid approach if the goal is to perform inference on α. It is a fine approach if the goal is solely the prediction of the outcome, but it can result in very misleading conclusions about the parameter of interest α, as we demonstrate in Example 4.3.1 below.

The naive approach outlined above relies on the moment condition

$$\mathsf{M}(a, b) = \mathrm{E}[(Y - aD - b'W)D] = 0.$$

When $b = \beta$, this moment condition is satisfied by the true value, $a = \alpha$. In this case, t coincides with the classical moment condition for α underlying low-dimensional ordinary least squares which sets prediction errors to be orthogonal to each predictor variable.

However, this moment condition does not exhibit Neyman orthogonality since

$$\partial_b \mathsf{M}(\alpha, \beta) = \mathrm{E}[DW] \neq 0$$

unless D is orthogonal to W.[4] Because $\mathsf{M}(a, b)$ is not Neyman orthogonal, the bias and the slower than parametric rate of convergence,

$$\sqrt{s \log(p \vee n)/n},$$

of our estimate of $\beta'W$ will transmit to bias and slower than $\sqrt{n}$ convergence in estimates of α provided by solving the empirical analog of $\mathsf{M}(a, b)$. The "Single Selection" procedure outlined

4: In "pure" RCTs where treatment is assigned independently of everything, D's are orthogonal to W, after de-meaning D, so Neyman orthogonality automatically holds in this setting.

above exactly provides the solution to this moment condition. Consequently, while this naive procedure provides an estimator of α that will approach the true value in large samples (at a slower than $\sqrt{n}$-rate), the bias of the estimator converges too slowly for standard inference methods to provide reliable inference.

We can set up a simulation experiment to verify that this naive approach provides low-quality estimates for α.

Example 4.3.1 In R Notebook with Experiment on Orthogonal vs Non-Orthogonal Learning and Python Notebook with Experiment on Orthogonal vs Non-Orthogonal Learning, we compare the performance of the naive and orthogonal methods in a computational experiment where $p = n = 100$, $\beta_j = 1/j^2$, $(\gamma_{DW})_j = 1/j^2$, and

$$Y = 1 \cdot D + \beta' W + \varepsilon_Y, \quad W \sim N(0, I), \quad \varepsilon_Y \sim N(0, 1)$$

$$D = \gamma'_{DW} W + \tilde{D}, \quad \tilde{D} \sim N(0, 1)/4.$$

From the histograms shown in Figure 4.1, we see that the naive estimator is heavily biased, as expected from the lack of Neyman orthogonality in its estimation strategy. We also see that the Double Lasso estimator, which is based on principled partialling-out such that Neyman orthogonality is satisfied, is approximately unbiased and Gaussian.

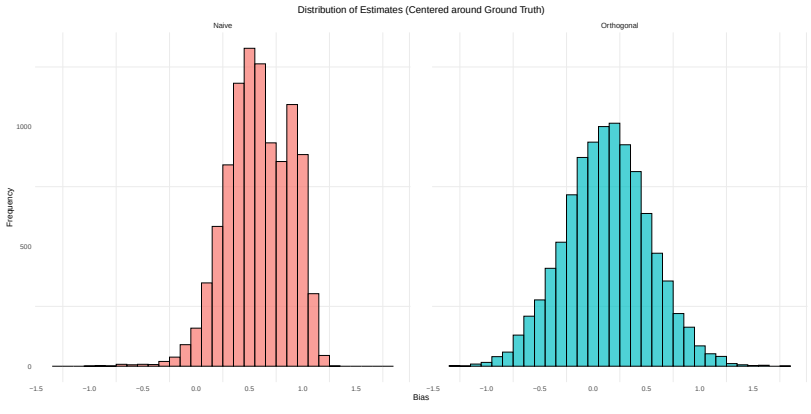


Figure 4.1: Left Panel: Simulated distribution of the naive (single-selection) non-orthogonal estimator centered around the true value. **Right Panel:** Simulated distribution of the orthogonal estimator centered around the true value.

The reason that the naive estimator does not perform well is that it only selects controls that are strong predictors of the outcome, thereby omitting weak predictors of the outcome. However, weak predictors of the outcome could still be strong predictors of D, in which case dropping these controls results in a strong omitted variable bias. In contrast, the orthogonal approach solves two prediction problems – one to predict Y and another to predict D – and finds controls that are relevant

for either. The resulting residuals are therefore approximately "de-confounded."

4.4 Inference on Many Coefficients

If we are interested in more than one coefficient, we can repeat the one-by-one Double Lasso procedure for each of the coefficients of interest and obtain valid estimation and inference on each component under regularity conditions.

We consider the model

$$\underbrace{Y}_{\text{Outcome}} = \underbrace{\sum_{\ell=1}^{p_1} \alpha_\ell D_\ell}_{\text{Target Predictors}} + \underbrace{\sum_{j=1}^{p_2} \beta_j \bar{W}_j}_{\text{Controls}} + \epsilon,$$

where we use D_ℓ for $\ell = 1, ..., p_1$ to denote the predictors of interest and $\bar{W}_j$ for $j = 1, ..., p_2$ to denote other predictors in the model. Here, both the number of predictors of interest, p_1, and the number of additional variables, p_2, can both be very large.

There are at least three motivations for considering many coefficients of interest:

- there can be multiple policies whose predictive effect we would like to infer;
- we can be interested in heterogeneous predictive effects across pre-specified groups;
- we can be interested in nonlinear effects of policies.

This setting encompasses examples where we are interested in *heterogeneous effects*, where $D_\ell's$ are generated as

$$D_\ell = D_0 \bar{X}_\ell, \quad \ell = 1, ..., p_1,$$

where D_0 is a base variable of interest – for example, a treatment indicator, a price, or a group indicator – and $(\bar{X}_\ell)_{\ell=1}^{p_1}$ are known transformations of controls $\bar{W}$ – for example, various subgroup indicators.

The setting also encompasses cases where *nonlinear effects* are of interest. For example, we could consider D_ℓ's generated as polynomial transformations of a multi-valued base variable, such as a price:

$$D_\ell = D_0^\ell, \quad \ell = 1, ..., p_1.$$

We could further interact these transformations with other variables to study nonlinear heterogeneous effects.

One by One Double Lasso for Many Target Parameters. For each $\ell = 1, \ldots, p_1$, we apply the one-by-one Double Lasso procedure for estimation and inference on the coefficient α_ℓ in the model

$$Y = \alpha_l D_\ell + \gamma_\ell' W_\ell + \epsilon, \quad W_\ell = ((D_k)_{k \neq \ell}', \bar{W}')'.$$

Under approximate sparsity conditions, the Double Lasso method provides a high-quality estimate $\hat{\alpha} = (\hat{\alpha}_\ell)_{\ell=1}^{p_1}$ of $\alpha = (\alpha_\ell)_{\ell=1}^{p_1}$ that is approximately Gaussian. We can thus easily construct individual confidence intervals or even joint confidence bands. Under regularity conditions, these results allow for simultaneous inference on $p_1 > n$ coefficients.

Theorem 4.4.1 (Double Lasso for Many Coefficients) *Under regularity conditions including approximate sparsity as in Definition 3.1.1 with parameters (A, a) with $a > 1$ in all partialling out steps and provided $(\log p_1)^5/n$ is small, we have the adaptivity property,*

$$\sqrt{\log p_1} \max_{\ell \leq p_1} \left| \sqrt{n}(\hat{\alpha}_\ell - \alpha_\ell) - (\mathbb{E}_n[\tilde{D}_\ell^2])^{-1} \sqrt{n} \mathbb{E}_n[\tilde{D}_\ell \epsilon] \right| \approx 0,$$

and, consequently, the Gaussian approximation

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

where

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

Recall that the above distributional approximation formally means that

$$\sup_{R \in \mathcal{R}} \left| \mathrm{P}\left(\sqrt{n}(\hat{\alpha} - \alpha) \in R \right) - \mathrm{P}\left(N(0, \mathsf{V}) \in R \right) \right| \to 0,$$

where $\mathcal{R}$ is a collection of all (hyper) rectangles. The latter result allows the construction of *simultaneous confidence bands* on all target parameters α_ℓ's of the form:

$$\widehat{CR} = \times_{\ell=1}^{p_1} \left[\hat{\alpha}_\ell \pm c \sqrt{\hat{\mathsf{V}}_{\ell\ell}/n} \right],$$

The critical value c in the simultaneous confidence band is

chosen so that

$$\begin{aligned} \mathrm{P}(\alpha \in \widehat{CR}) &= \mathrm{P}\left(\sqrt{n}(\alpha - \hat{\alpha}) \in \sqrt{n}(\hat{CR} - \hat{\alpha})\right) \\ &= \mathrm{P}\left(\sqrt{n}(\alpha_\ell - \hat{\alpha}_\ell) \in [\pm c\hat{\mathrm{V}}_{\ell\ell}^{1/2}] \; \forall \; \ell \in \{1, ..., p_1\}\right) \\ &\approx 1 - \mathsf{a} \end{aligned}$$

where $1 - \mathsf{a}$ denotes the confidence level.

The use of a simultaneous confidence band when looking at multiple coefficients allows us to control the probability that even one coefficient from the set we are investigating falls outside of the interval. For instance, a 95% simultaneous confidence band implies that, if we were to repeat the data sampling process many times, then in 95% of these repetitions *all coefficients* would lie within their respective interval.

There is nothing special about 95% here. You could replace all instances with $1 - \mathsf{a}$ if you were interested in $(1 - \mathsf{a})$% confidence statements.

On the contrary, standard 95% confidence intervals for each coefficient – typically referred to as "marginal confidence intervals" – only guarantee that *separately* each coefficient falls in its interval in 95% of the experiments. However, these *success events* for different coefficients can happen on different repetitions. In the worst-case, these success events could be independent random variables with success probability 95%. In this case, the probability that we observe one failure when we look at p_1 coefficients could be much larger than 5%; i.e. $1 - \mathrm{P}(\text{no confidence interval failed}) = 1 - (1 - 0.05)^{p_1} \gg 0.05$ and approaches 1 as p_1 grows.

These properties mean that marginal confidence intervals are generally inappropriate for judging statistical relevance when multiple coefficients are of interest. For example, if we declare any variable whose marginal 95% confidence interval excludes zero "statistically significant" or a "discovery," the probability that we mistakenly make discoveries – in the sense of claiming a coefficient is not zero when it in fact is – is not 0.05 but potentially substantially larger, e.g. $1 - (1 - 0.05)^{p_1}$ under independence of success events. If instead we report a 95% simultaneous confidence band, this probability of making false discoveries is at most 0.05. Of course, false discovery rate control is only one reason why one might care about the stronger guarantee that a simultaneous confidence band provides. [5] For a survey on simultaneous inference in high dimensions see [6].

5: If one is particularly interested in false discovery rate (FDR) control, then more tailored procedures could potentially be less conservative than the simultaneous confidence band and can be combined with the marginal confidence interval and marginal p-value constructions we provide in this book. See e.g. [4]. See also [5] for more on FDR control and the use of multidimensional Gaussian approximations.

Remark 4.4.1 (Details on critical values) It can be shown that

an "ideal" choice of c is

$$c = (1 - \mathsf{a}) - \text{ quantile of } \left\| N\left(0, \mathsf{D}^{-1/2}\mathsf{V}\mathsf{D}^{-1/2}\right)\right\|_\infty,$$

where $\mathsf{D} = \text{diag}(\mathsf{V})$ is a matrix with variances $(\mathsf{V}_{\ell\ell})_{\ell=1}^{p_1}$ on the diagonal and zeroes off the diagonal. The critical value c can therefore be approximated by simulation plugging in $\mathsf{V} = \hat{\mathsf{V}}$. Please see [6], for example, for more details. Note that c is generally no smaller than the $(1 - \mathsf{a}/2)$-quantile of a $N(0, 1)$, so the simultaneous confidence bands are always no smaller than the component-wise confidence bands.

Discovering Heterogeneity in the Wage Gap Analysis

R Notebook on Double Lasso for the Heterogeneous Wage Gap and Python Notebook on Double Lasso for the Heterogeneous Wage Gap provide code for the wage gap illustration.

We apply the Double Lasso method to analyze heterogeneity of wage gaps using our CPS 2015 data. As in Chapter 1, we use the log hourly wage as the outcome variable. To explore heterogeneity, we interact the female indicator with group indicators capturing education groups (Some High School (shs), High School Graduate (hsg), Some College (scl), College Graduate (clg), Advanced Degree (ad)), region indicators – Midwest (mw), South (so), West (we)) and a fourth degree polynomial in experience (exp1= Experience, exp2= Experience^2/100, exp3= Experience^3/1000, exp4= Experience^4/10000). In total these are 12 target parameters corresponding to the 11 interactive variables and the non-interactive variable that corresponds to the female indicator. All engineered variables used for heterogeneity were de-meaned prior to taking the interaction with sex, while the sex variable was not de-meaned. Hence, the interaction coefficients can be interpreted as "predictive effect modifiers," and the coefficient associated with the non-interactive variable sex as the average predictive effect. As additional variables, we also include all pairwise interactions of the aforementioned variables (excluding sex), as well as one-hot-encodings for occupation and industry sector, providing 990 engineered features. All engineered variables used as controls were also de-meaned prior to estimation.

Table 4.2 provides estimated coefficients, standard errors, pointwise p-values, and the 95% simultaneous confidence band for the coefficients on sex and its interactions with the schooling (shs, hsg, scl, clg, and ad), region (mw, so, and we), and experience (exp1, exp2, exp3, and exp4) variables described above. Rows give variable names with "*" indicating interaction; e.g.

	Estimate	Std. Error	p-value	Sim. Band lower	Sim. Band upper
sex	-0.07	0.02	0.00	-0.11	-0.02
sex:shs	-0.20	0.11	0.07	-0.53	0.14
sex:hsg	0.01	0.05	0.80	-0.14	0.16
sex:scl	0.02	0.05	0.65	-0.12	0.17
sex:clg	0.06	0.04	0.16	-0.08	0.20
sex:mw	-0.11	0.04	0.01	-0.23	0.01
sex:so	-0.07	0.04	0.07	-0.19	0.04
sex:we	-0.05	0.04	0.22	-0.18	0.07
sex:exp1	0.02	0.01	0.01	-0.00	0.04
sex:exp2	0.02	0.05	0.64	-0.12	0.17
sex:exp3	-0.05	0.03	0.10	-0.16	0.06
sex:exp4	-0.01	0.00	0.00	-0.01	-0.00

Table 4.2: Estimates of Heterogeneous Predictive Effects in the CPS 2015 data. Row labels correspond to variable names as described in the text; e.g. the row "sex*shs" corresponds to the interaction between sex and shs (a dummy for having completed some high school). Estimated coefficients and standard errors are given in the "Estimate" column and "Std. Error" column respectively. The marginal p-value is given in the "p-value" column. The remaining columns "Sim. Band lower" and "Sim. Band upper" provide the lower and upper bounds of the simultaneous confidence band for each variable.

the row sex*shs provides results for the interaction between sex and shs.

Looking coefficient by coefficient, we see evidence that having a college degree increases the predictive effect, i.e. decreases the wage gap, while the largest increase in wage gap occurs for the least educated workers. However, as judged by pointwise p-values, these heterogeneities are not statistically significant at the usual 5% level. We also see that the wage gap is predicted to be larger in the Midwest region, and this is effect is statistically significant at the 5% level based on the marginal p-value. However, care should be taken when looking at pointwise results. The simultaneous confidence regions are relatively wide and include 0 for all coefficients except for the main effect on sex, suggesting that it may be difficult to draw any strong conclusions about heterogeneity of predictive effects in this example.

4.5 Other Approaches That Have the Neyman Orthogonality Property

Double Selection

One way to fix the naive "single selection" approach outlined in Section 3 would be to have "double selection":

Double Selection

- find controls W_Y that predict Y as judged by Lasso;
- find controls W_D that predict D as judged by Lasso;
- regress Y on D and the union of controls $W_Y \cup W_D$; proceed with standard inference.

This procedure is approximately equivalent to the partialling out approach, and therefore inherits the orthogonality property. This approach is more conservative compared to single selection, as it makes sure that we have not omitted controls that are strong confounders for D. It therefore guards against large omitted variable biases.

Desparsified Lasso

Yet another procedure that has the orthogonality property and is approximately equivalent to the partialling out approach under suitable conditions is desparsified Lasso.

This approach uses the fact that $a = \alpha$ solves the equation,

$$\mathsf{M}(a, \eta) = \mathrm{E}[(Y - aD - b'W)\tilde{D}(\gamma)] = 0,$$

when $\eta = (b', \gamma')' = \eta^o := (\beta', \gamma'_{DW})'$ for γ_{DW} the best linear predictor coefficient from regressing D onto W and

$$\tilde{D}(\gamma) = D - \gamma' W.$$

One can verify that

$$\alpha(\eta) = \left(\mathrm{E}[D\tilde{D}(\gamma)]\right)^{-1} \mathrm{E}\left[(Y - b'W)\tilde{D}(\gamma)\right],$$

and that

$$\alpha = \alpha(\eta^o).$$

Further, the moment condition is Neyman orthogonal – verification of which is left to the reader – which implies that

$$\partial_\eta \alpha(\eta^o) = 0,$$

similarly to the argument for Double Lasso.

Desparsified Lasso

- Run a Lasso estimator with suitable choice of λ as discussed in Chapter 3 of Y on D and W, and save the coefficient estimate $\hat{\beta}$.

- Run a Lasso estimator with suitable choice of λ as discussed in Chapter 3 of D on W and save the coefficient estimate $\hat{\gamma}$.
- The estimator $\hat{\alpha}$ is then the solution of the empirical analog of the moment condition above:

$$\mathbb{E}_n[(Y - \hat{\alpha}D - \hat{\beta}'W)\tilde{D}(\hat{\gamma})] = 0,$$

which has the explicit form

$$\hat{\alpha} = \left(\mathbb{E}_n[D\tilde{D}(\hat{\gamma})]\right)^{-1} \mathbb{E}_n\left[(Y - \hat{\beta}'W)\tilde{D}(\hat{\gamma})\right],$$

where $\hat{\beta}$ and $\hat{\gamma}$ are Lasso estimators.

Estimators of this form are referred to in econometrics as "instrumental variable estimators." In purely technical terms, we are using residualized $\tilde{D}$ to "instrument" for D.

Revisiting the Price Elasticity for Toy Cars

Next, we revisit the example from Chapter 0. We are interested in the coefficient α in the high-dimensional linear regression model:

$$Y = \alpha D + \beta'X + \epsilon,$$

where Y is log-reciprocal=sales-rank, D is log-price, and $X = (1, W)$ with product features W. We here take X to be the same 11546-dimensional transformed regressors as described in Chapter 0, constructed from product brand, subcategory, and physical dimensions. Here we have $p > n = 9212$, so OLS is underspecified, and even if we consider a specific solution to the normal equations such as the one with the minimum norm, standard errors are unavailable or unreliable. We can still run OLS when we subset the regressors, or equivalently impose that the coefficients on the rest are zero. In Table 4.3 we report the results for such an approach with OLS with three specifications of increasing size: $p = 243$ with only subcategory features (as in Chapter 0), $p = 2069$ after also adding brand features, and $p = 2073$ after also adding log of the physical dimensions features (but without any transformations or interactions). We see that in all cases we cannot exclude 0 from the confidence interval, while the more flexible we make our model (larger p), the more negative our estimates and confidence intervals.

Next, we consider estimating elasticities using double lasso, double selection, and desparsified lasso applied to all $p = 11546$

features. In all cases, we pick the regularization parameter by 5-fold cross validation (for the regression of each of Y and D). Then we apply the three methods using the lasso models fit or the variables chosen by them. The results are reported in Table 4.3. We see that all three methods result in confidence intervals that are strictly negative, in agreement with the theory that increasing price for any one product decreases its sales.

	Estimate	Std. Error	95% CI
OLS ($p = 242$)	0.005	0.016	[-0.026, 0.036]
OLS ($p = 2068$)	-0.003	0.021	[-0.045, 0.039]
OLS ($p = 2072$)	-0.033	0.022	[-0.076, 0.010]
Double Lasso	-0.064	0.018	[-0.099, -0.029]
Double Selection	-0.074	0.019	[-0.111, -0.037]
Desparsified Lasso	-0.062	0.017	[-0.096, -0.028]

Table 4.3: Estimates for price elasticity. We report specification robust standard errors with finite sample correction, i.e., "HC1." All non-OLS methods have $p = 11546$.

Notebooks

- R Notebook with Experiment on Orthogonal vs Non-Orthogonal Learning and Python Notebook with Experiment on Orthogonal vs Non-Orthogonal Learning presents the simulation experiment comparing orthogonal (partialling-out) with non-orthogonal learning (naive method).

- R Notebook with Hard Sparsity on Orthogonal vs Non-Orthogonal Learning and Python Notebook with Hard Sparsity on Orthogonal vs Non-Orthogonal Learning presents an alternative simulation to that shown in the main text comparing orthogonal (partialling-out) with non-orthogonal learning. In this simulation, we consider orthogonal and non-orthogonal learning in a stylized treatment effects simulation.

- R Notebook on Double Lasso for Growth Convergence and Python Notebook on Double Lasso for Growth Convergence presents a Double Lasso analysis of the conditional convergence hypothesis in growth economics.

- R Notebook on Double Lasso for the Heterogeneous Wage Gap and Python Notebook on Double Lasso for the Heterogeneous Wage Gap presents a Double Lasso analysis of the heterogeneous wage gap.

Notes

We mainly follow the Double Lasso approach developed in [7] and [8], because it is nicely connected to partialling out and will later generalize seamlessly to double machine learning [9]. Desparsified Lasso was developed by [10] and [11]; a closely related approach is the debiased Lasso proposed by [12]. The double selection method was developed by [13] and [14]. Inference on many coefficients using Double Lasso was first developed by [15] and [16]. [17] provide results for Double Lasso with clustered dependence. The Double Lasso and desparsified Lasso approaches have also been extended to time series and many time series by [18]. Both [17] and [18] take into account the temporal dependencies in the data when fitting Lasso and performing inference on the coefficients of interest.

Failure of single selection even when p is small is discussed in simple terms in [14], but the problem was first systematically examined by [19]. A recent paper [20] develops debiasing methods for shape constrained high-dimensional linear regression models.

[6] provide a recent survey on methods for simultaneous inference in high-dimensional settings.

For an in-depth analysis of heterogeneity in the wage gap based on Lasso, we refer to [21].

Study Problems

1. Experiment with the first notebook, R Notebook with Experiment on Orthogonal vs Non-Orthogonal Learning or Python Notebook with Experiment on Orthogonal vs Non-Orthogonal Learning. Try different models. For example, try different coefficient structures for β and γ_{DW} and/or different covariance structures for W. Provide an explanation to a friend for what each step in the Double Lasso procedure is doing.

2. Explore R Notebook on Double Lasso for Growth Convergence or Python Notebook on Double Lasso for Growth Convergence. Provide an explanation to a friend for what each step in the Double Lasso procedure is doing. Explain the empirical results to a friend. Experiment with making the set of controls more flexible and higher-dimensional by adding nonlinear and/or interaction terms that seem potentially interesting. Comment on how the results differ

from the baseline results.

3. Explore R Notebook on Double Lasso for the Heterogeneous Wage Gap and Python Notebook on Double Lasso for the Heterogeneous Wage Gap. Provide an explanation to a friend for what each step in the inference procedure is doing. Explain the empirical results to a friend.

4. Verify that Neyman orthogonality holds for the "desparsified" Lasso strategy.

4.A High-Dimensional Central Limit Theorems*

Let $X_1, \ldots, X_n$ be independent (but not necessarily identically distributed) random vectors with dimension p. Assume that X_i's have mean zero (otherwise, work with $X_i - \mathrm{E}[X_i]$ instead of X_i). Consider the scaled sample mean

$$S_n = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} X_i.$$

Let $\bar{\sigma}, \underline{\sigma}$ be given positive constants such that $\underline{\sigma} \leqslant \bar{\sigma}$, and let $B_n \geqslant 1$ be a sequence of constants that may diverge as $n \to \infty$. Let $\Sigma_n = \mathrm{E}[S_n S_n'] = n^{-1} \sum_{i=1}^{n} \mathrm{E}\left[X_i X_i'\right]$. Also, let $\mathcal{R}$ denote the collection of closed rectangles in $\mathbb{R}^p$.

We first present a high-dimensional CLT over the rectangles under a sub-exponential condition on the coordinates. Suppose that the coordinates of $X_1, \ldots, X_n$ are sub-exponential with scale B_n, then

$$\sup_{R \in \mathcal{R}} |\mathrm{P}(S_n \in R) - \mathrm{P}(N(0, \Sigma_n) \in R)| \approx 0, \tag{4.A.1}$$

provided that $B_n^2 \log^5(pn)/n \approx 0$. Note that this allows p to be much larger than n. It turns out that a similar result applies without sub-exponential conditions, as stated formally below.

To state the results in a finite-sample form, let

$$\delta_{1,n} := \left(\frac{B_n^2 \log^5(pn)}{n}\right)^{1/4} \text{ and } \delta_{2,n}^{[q]} := \sqrt{\frac{B_n^2 (\log(pn))^{3-2/q}}{n^{1-2/q}}},$$

for $q > 2$.

Theorem 4.A.1 (High-Dimensional CLT, [22]) *Suppose second moments are non-degenerate,* $\min_{j\leqslant p} n^{-1}\sum_{i=1}^{n} \mathrm{E}\left[X_{ji}^2\right] \geqslant \underline{\sigma}^2$, *and fourth moments obey* $\max_{j\leqslant p} n^{-1}\sum_{i=1}^{n} \mathrm{E}\left[X_{ji}^4\right] \leqslant B_n^2\bar{\sigma}^2$.

(A) If coordinates are subexponential, i.e., $\max_{i\leqslant n;j\leqslant p} \mathrm{E}\left[e^{|X_{ji}|/B_n}\right] \leqslant 2$, *then*

$$\sup_{R\in\mathcal{R}} |\mathrm{P}(S_n \in R) - \mathrm{P}(N(0,\Sigma_n)\in R)| \leq C\delta_{1,n},$$

where C *is a constant that depends only on* $\underline{\sigma}$ *and* $\bar{\sigma}$.

(B) If the envelope of the coordinates admits a moment bound $\max_{i\leqslant n} \mathrm{E}\left[\|X_i\|_\infty^q\right] \leq B_n^q$ *for some* $q > 2$, *then*

$$\sup_{R\in\mathcal{R}} |\mathrm{P}(S_n \in R) - \mathrm{P}(N(0,\Sigma_n)\in R)| \leq C\left(\delta_{1,n} \vee \delta_{2,n}^{[q]}\right)$$

where C *is a constant that depends only on* q, $\underline{\sigma}$ *and* $\bar{\sigma}$.

Notably, the above theorem does not impose any restrictions on the correlation structure between the coordinates of the random vectors, so Σ_n is permitted to be singular.

As discussed in [23], the assumption of Part (A) is satisfied if, for example, $\left|X_{ji}\right| \leqslant B_n$ for all (i,j), but also allows for unbounded coordinates. Part (B) covers the following scenario relevant to regression applications: $X_i = \epsilon_i v_i$ where ϵ_i is a univariate "error" term while $v_i \in \mathbb{R}^p$ is a vector of fixed "covariates." In this case, $\mathrm{E}\left[\|X_i\|_\infty^q\right] \leqslant \|v_i\|_\infty^q \mathrm{E}\left[|\epsilon_i|^q\right]$, so if the covariates are uniformly bounded and the q-th moments of the error terms are bounded, then $B_n = O(1)$. Notably this only requires ϵ_i to have $q = 2+\delta$ bounded moments.

Often, statistics of interest are not exactly sample means, but can be well approximated by sample means. For example, the Double Lasso estimator, $\widehat{\alpha} = (\mathbb{E}_n[\check{D}^2])^{-1}\mathbb{E}_n[\check{D}\check{Y}] \approx (\mathrm{E}[\tilde{D}^2])^{-1}\mathbb{E}_n[\tilde{D}\tilde{Y}]$, takes this form. In order to claim a High-Dimensional CLT for such statistics, we need the approximation error to vanish at the rate faster than $1/\sqrt{\log p}$.[6]

Lemma 4.A.2 (High-dimensional CLT for approximate sample mean) . *Suppose that* S_n *obeys (4.A.1), but* S_n *is not directly available. Suppose instead that we have access to* $\widehat{S}_n$ *that approximates* S_n *such that* $\widehat{S}_n = S_n + \mathrm{R}_n$ *with* $\sqrt{\log p}\,\|\mathrm{R}_n\|_\infty \approx 0$. *Assume* $\min_{j\leqslant p}\Sigma_{jj} \geqslant \underline{\sigma}^2$. *Then the same conclusion holds with* S_n *replaced by* $\widehat{S}_n$.

6: The requirement that approximation error, denoted R_n, vanishes faster than $1/\sqrt{\log p}$ arises from the fact that the maximum of a Gaussian random vector $N(0,\Sigma)$ concentrates in (i.e., places a probability mass of near 1 to) a $1/\sqrt{\log p}$- neighborhood of its expected value, but not in smaller neighborhoods (anti-concentration). The approximation error R_n needs to be much smaller than the size of the neighborhood. Otherwise, the probabilistic errors incurred by Gaussian approximation to the distribution of $\hat{S}$ can be as large as 1, meaning that the Gaussian approximation fails.

The lemma follows from Nazarov's anticoncentration inequality for Gaussian vectors over rectangles; see [23] for the proof.

Bibliography

[1] Ragnar Frisch and Frederick V Waugh. 'Partial time regressions as compared with individual trends'. In: *Econometrica* (1933), pp. 387–401 (cited on page 101).

[2] Robert Barro and Jong-Wha Lee. 'A new data set of educational attainment in the world, 1950–2010'. In: *Journal of Development Economics* 104.C (2013), pp. 184–198 (cited on page 106).

[3] Matias D. Cattaneo, Michael Jansson, and Whitney K. Newey. 'Inference in linear regression models with many covariates and heteroscedasticity'. In: *Journal of the American Statistical Association* 113.523 (2018), pp. 1350–1361 (cited on page 106).

[4] Yoav Benjamini and Daniel Yekutieli. 'False Discovery Rate–Adjusted Multiple Confidence Intervals for Selected Parameters'. In: *Journal of the American Statistical Association* 100.469 (2005), pp. 71–81. DOI: 10.1198/016214504000001907 (cited on page 113).

[5] Alexandre Belloni, Victor Chernozhukov, Denis Chetverikov, Christian Hansen, and Kengo Kato. 'High-dimensional econometrics and regularized GMM'. In: *arXiv preprint arXiv:1806.01888* (2018) (cited on page 113).

[6] Philipp Bach, Victor Chernozhukov, and Martin Spindler. *Valid Simultaneous Inference in High-Dimensional Settings (with the hdm package for R)*. 2018. DOI: 10.48550/ARXIV.1809.04951. URL: https://arxiv.org/abs/1809.04951 (cited on pages 113, 114, 119).

[7] Victor Chernozhukov, Christian Hansen, and Martin Spindler. 'Valid post-selection and post-regularization inference: An elementary, general approach'. In: *Annual Review of Economics* 7.1 (2015), pp. 649–688 (cited on page 119).

[8] Alexandre Belloni, Victor Chernozhukov, and Lie Wang. 'Pivotal estimation via square-root lasso in nonparametric regression'. In: *Annals of Statistics* 42.2 (2014), pp. 757–788 (cited on page 119).

[9] Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. 'Double/debiased machine learning for treatment and structural parameters'. In: *Econometrics Journal* 21.1 (2018), pp. C1–C68 (cited on page 119).

[10] Cun-Hui Zhang and Stephanie S. Zhang. 'Confidence intervals for low dimensional parameters in high dimensional linear models'. In: *Journal of the Royal Statistical Society: Series B* 76.1 (2014), pp. 217–242 (cited on page 119).

[11] Sara Van de Geer, Peter Bühlmann, Ya'acov Ritov, and Ruben Dezeure. 'On asymptotically optimal confidence regions and tests for high-dimensional models'. In: *Annals of Statistics* 42.3 (2014), pp. 1166–1202 (cited on page 119).

[12] Adel Javanmard and Andrea Montanari. 'Confidence intervals and hypothesis testing for high-dimensional regression'. In: *Journal of Machine Learning Research* 15.1 (2014), pp. 2869–2909 (cited on page 119).

[13] Alexandre Belloni, Victor Chernozhukov, and Christian B. Hansen. 'Inference for High-Dimensional Sparse Econometric Models'. In: *Advances in Economics and Econometrics: Tenth World Congress*. Ed. by Daron Acemoglu, Manuel Arellano, and Eddie Dekel. Vol. 3. Econometric Society Monographs. Cambridge University Press, 2013, pp. 245–295. DOI: 10.1017/CBO9781139060035.008 (cited on page 119).

[14] Alexandre Belloni, Victor Chernozhukov, and Christian Hansen. 'Inference on Treatment Effects After Selection Amongst High-Dimensional Controls'. In: *Review of Economic Studies* 81.2 (2014), pp. 608–650 (cited on page 119).

[15] Alexandre Belloni, Victor Chernozhukov, and Kengo Kato. 'Uniform post-selection inference for least absolute deviation regression and other Z-estimation problems'. In: *Biometrika* 102.1 (2015), pp. 77–94 (cited on page 119).

[16] Alexandre Belloni, Victor Chernozhukov, Denis Chetverikov, and Ying Wei. 'Uniformly valid post-regularization confidence regions for many functional parameters in z-estimation framework'. In: *Annals of statistics* 46.6B (2018), p. 3643 (cited on page 119).

[17] Alexandre Belloni, Victor Chernozhukov, Christian Hansen, and Damian Kozbur. 'Inference in High-Dimensional Panel Models With an Application to Gun Control'. In: *Journal of Business & Economic Statistics* 34.4 (2016), pp. 590–605 (cited on page 119).

[18] Victor Chernozhukov, Wolfgang Karl Härdle, Chen Huang, and Weining Wang. 'Lasso-driven inference in time and space'. In: *Annals of Statistics* 49.3 (2021), pp. 1702–1735 (cited on page 119).

[19] Hannes Leeb and Benedikt M. Pötscher. 'Model selection and inference: Facts and fiction'. In: *Econometric Theory* 21.1 (2005), pp. 21–59 (cited on page 119).

[20] Yufei Yi and Matey Neykov. 'A New Perspective on Debiasing Linear Regressions'. In: *arXiv preprint arXiv:2104.03464* (2021) (cited on page 119).

[21] Philipp Bach, Victor Chernozhukov, and Martin Spindler. *Closing the U.S. gender wage gap requires understanding its heterogeneity*. 2018. DOI: 10.48550/ARXIV.1812.04345. URL: https://arxiv.org/abs/1812.04345 (cited on page 119).

[22] Victor Chernozhukov, Denis Chetverikov, Kengo Kato, and Yuta Koike. 'Improved central limit theorem and bootstrap approximations in high dimensions'. In: *Annals of Statistics* 50.5 (2022), pp. 2562–2586 (cited on page 121).

[23] Victor Chernozhukov, Denis Chetverikov, Kengo Kato, and Yuta Koike. 'High-dimensional Data Bootstrap'. In: *Annual Review of Statistics and Applications; arXiv preprint arXiv:2205.09691* (2023) (cited on pages 121, 122).

Causal Inference via Conditional Ignorability

5

"compare apples and/to/with apples: to compare things that are very similar."

– Merriam Webster Dictionary [1].

Here we discuss how average causal effects may be identified using regression when treatment is not randomly assigned but instead depends on observed covariates. We discuss the conditional or adjustment method, which relies on comparing the average difference between expected outcomes for treated and untreated units that are comparable (formally, identical) in terms of their characteristics X. If treatment is as good as randomly assigned conditional on X, then this approach recovers average causal or treatment effects. This key condition is commonly referred to as conditional ignorability, conditional exogeneity, or unconfoundedness.

5.1 Introduction

In a cross-country analysis, higher chocolate consumption predicts a higher number of Nobel laureates per capita.

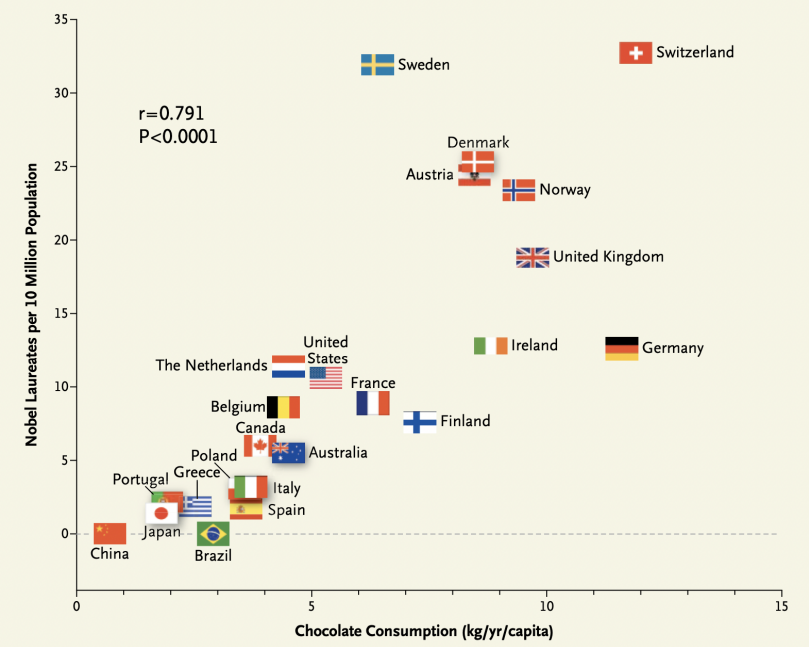


Figure 5.1: Source: Franz H. Messerli, "Chocolate Consumption, Cognitive Function, and Nobel Laureates," New England Journal of Medicine. 2012

Is this a reflection of a true causal effect and therefore an actionable insight? If it were, countries could generate more Nobel laureates per capita by making chocolate abundant to everyone. (This wouldn't be a bad thing.) Is this perhaps what Switzerland did? Switzerland has the highest number of Nobel laureates per capita.

Or is there a common cause[1] that creates non-causal association? Perhaps wealthy countries invest more in science and higher wealth causes people to consume luxury goods like chocolate. See for instance plots (D) and (E) in Figure 5.3. Comparative analysis, where we compare nations with identical or similar wealth, would probably reveal that the correlation is not causal.[2]

Probably we should be comparing Switzerland to similar countries in terms of wealth – the "apples-to-apples" comparison, so to speak. This type of analysis is very common in causal

1: We often refer to these common causes as "omitted variables" that give rise to "omitted variable bias."

2: It remains a fundamental empirical problem to confirm this conjecture or disprove this conjecture. The causal channel through which chocolate (and other flavonoids) may affect Nobel production is by documented improvement in the cognitive function.

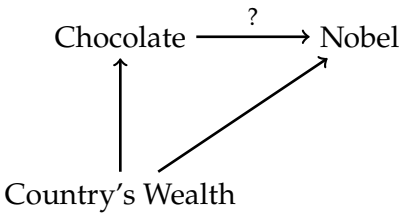


Figure 5.2: A Contrived Causal Path Diagram for the Effect of Country's Wealth on Chocolate Consumption and Nobel Prize Production per capita.

inference and is implemented via a set of tools introduced in this chapter.

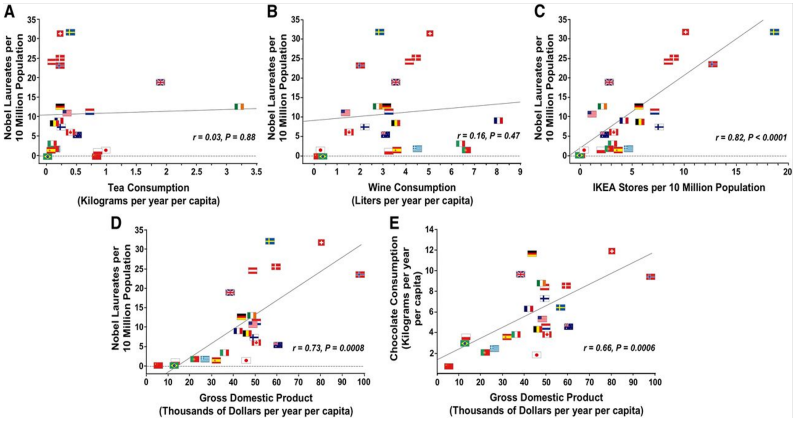


Figure 5.3: Source: J Nutr, Volume 143, Issue 6, June 2013, Pages 931–933, "Does Chocolate Consumption Really Boost Nobel Award Chances? The Peril of Over-Interpreting Correlations in Health Studies," ©2013 American Society for Nutrition

In what follows, we work within Rubin's [2] potential outcomes framework, as introduced in Chapter 2. The idea is that if we can think of observed treatment D as generated randomly – independently of potential outcomes – conditional on some pre-treatment variables X, then we can learn the average causal (treatment) effects by regression

$$\text{of } Y \text{ on } D \text{ and } X,$$

or, as is often said, by "adjusting" or "controlling" for X.

Notation

Recall that we denote the independence of two random variables (these can include random vectors) U and V as

$$U \perp\!\!\!\perp V.$$

Independence, conditional on a third variable X, is denoted by

$$U \perp\!\!\!\perp V \mid X.$$

5.2 Potential Outcomes and Ignorability

Recall that we use $Y(d)$ to denote potential outcome in the treatment state d, where we consider only the case $d \in \{0, 1\}$ for simplicity. We also recall our example of smoking from Chapter 2. Suppose we want to study the impact of smoking

marijuana on life longevity. Suppose that smoking marijuana has no causal/treatment effect on life longevity:

$$Y = Y(0) = Y(1), \text{ so that } \delta = \mathrm{E}[Y(1)] - \mathrm{E}[Y(0)] = 0.$$

However, the observed smoking behavior, D, results not from an experimental study, but from observational data in which an individual's smoking decisions are driven by other behavioral choices X (drinking alcohol for example) which cause shorter life longevity. In this case, the predictive effect recovered by regression without adjusting for X does not match the average causal effect

$$\mathrm{E}[Y \mid D = 1] - \mathrm{E}[Y \mid D = 0] < 0 = \delta,$$

because higher D predicts higher X, which predicts lower Y. This difference between the predictive effect and average causal effect is the result of confounding or *selection bias*.

In this example, conditioning on X can remove the selection bias (see Figure 5.4)

$$\mathrm{E}[\mathrm{E}[Y \mid D = 1, X] - \mathrm{E}[Y \mid D = 0, X]] = \delta,$$

provided that conditional on X variation in D is independent of the potential health outcomes.

The following provides a formal assumption under which we can eliminate the confounding bias by controlling for X.[3]

Assumption 5.2.1 (Conditional Ignorability and Consistency) *Ignorability: Suppose that treatment status D is independent of potential outcomes $Y(d)$ conditional on a set of covariates X: For each d,*

$$D \perp\!\!\!\perp Y(d) \mid X.$$

Consistency: Suppose that Y is generated as $Y := Y(D)$.

3: The assumption is fundamentally untestable and is an assumption in the purest sense. Given assumed domain knowledge encoded in causal DAGs, we study a systematic way of finding X that satisfy this assumption in subsequent chapters.

Identification by Conditioning

The ignorability assumption[4] says that variation in treatment assignment D is as good as random conditional on X. This assumption means that if we look at units with the same value of the covariates, e.g. units with $X = x$, then treatment variation among these observationally identical units, $D \mid X = x$, is indeed produced as if by a formal randomized control trial.

4: You may wonder why the term "ignorability" is used. The distribution of $Y(d)$ depends only on X and not on D, so the latter is "ignorable." Note that the conventional name used in econometrics for the ignorability assumption is the *conditional exogeneity* or *conditional independence* assumption.

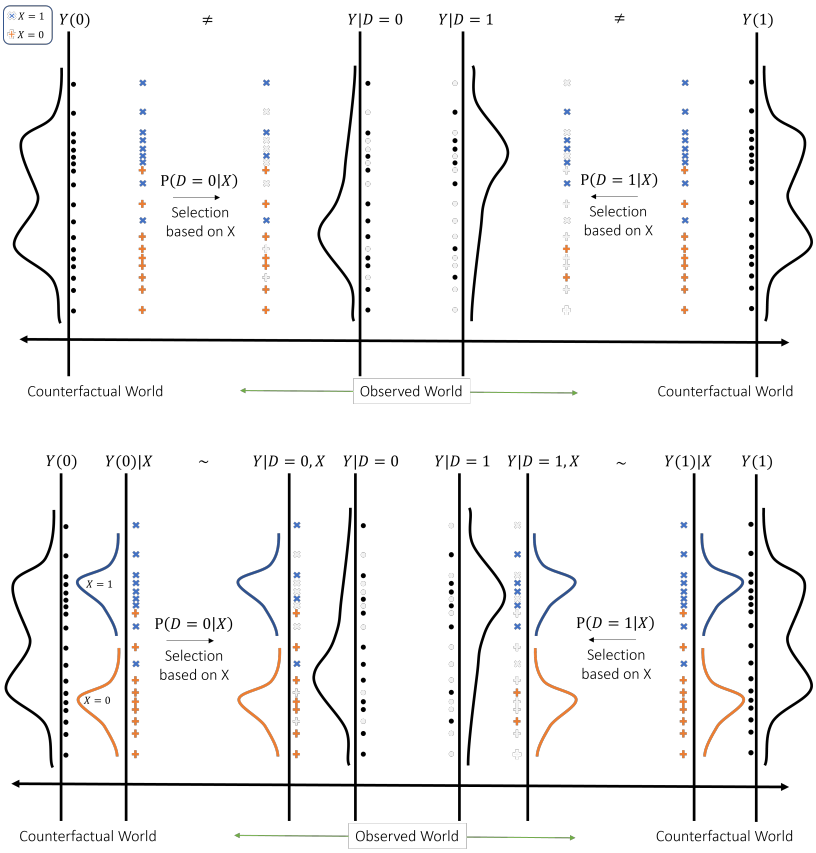


Figure 5.4: Pictorial representation of how selection on X can lead to biased observed outcomes between treated and control populations, while conditioning on X removes the selection bias. In this example, the potential outcomes $Y(0)$ and $Y(1)$ have identical distributions shown in the far left and right of the figure. We also have a binary covariate X that is related to treatment probability in the sense that $\mathrm{P}(D = 1|X = 1) > \mathrm{P}(D = 1|X = 0)$ and $\mathrm{P}(D = 0|X = 1) < \mathrm{P}(D = 0|X = 0)$ which leads to selection bias when we do not condition on X. This bias is illustrated by the difference in the distribution of (observed) Y given $D = 0$ and $D = 1$ shown in the black curves in the middle of the figure. The bottom panel then shows that selection bias is removed by conditioning on X as the distribution of potential outcomes given X (blue and orange curves under $Y(0)|X$ and $Y(1)|X$) equals the distribution of observed outcomes given D and X (blue and orange curves under $Y|D = 0, X$ and $Y|D = 1, X$).

Therefore, we can learn about the causal effect of D by comparing outcomes across treated and control units who have identical characteristics $X = x$ under the conditional ignorability assumption. The idea of comparing observations who have identical characteristics is the essence of the so-called *conditioning* or *adjustment* strategy to learning causal effects. As conditioning approaches produce a different contrast for every potential value of X, we may also wish to average the contrasts at different values of X over the distribution of characteristics to produce a summary measure of the causal effects.

The conditional probability of receiving treatment, *the propensity score*, plays an important role in this approach.

Assumption 5.2.2 (Overlap/Full Support) *The probability of receiving treatment given X, the propensity score*

$$p(X) := \mathrm{P}(D = 1|X),$$

is non-degenerate:

$$\mathrm{P}(0 < p(X) < 1) = 1.$$

The overlap assumption requires that there is proper randomization or variation in D at each value x in the support of X. Without this condition, there are values x in the support of X where we cannot construct a contrast between treatment and control units. We cannot learn the conditional average treatment effect at these values of X and thus are also unable to learn the unconditional average effect of the treatment.

Remark 5.2.1 Assumption 5.2.2 is also often called the *full support* condition because it requires

$$\text{support}(\text{D}, \text{X}) = \{0, 1\} \times \text{support}(X).$$

The following is the most important theoretical result that states that we can recover expectations of potential outcomes from regressions.

Theorem 5.2.1 (Conditioning on X Removes Selection Bias) *Under Conditional Ignorability and Overlap, the conditional expectation function of observed outcome Y given $D = d$ and X recovers the conditional expectation of the potential outcome $Y(d)$ given X:*

$$\text{E}[Y \mid D = d, X] = \text{E}[Y(d) \mid D = d, X] = \text{E}[Y(d) \mid X].$$

To prove Theorem 5.2.1, note that the overlap assumption makes it possible to condition on the events $\{D = 0, X\}$ and $\{D = 1, X\}$ at any value in the support of X and that the second equality holds by ignorability.

Hence, the Conditional Average Predictive Effect (CAPE),

$$\pi(X) = \text{E}[Y \mid D = 1, X] - \text{E}[Y \mid D = 0, X],$$

is equal to the Conditional Average Treatment Effect (CATE),

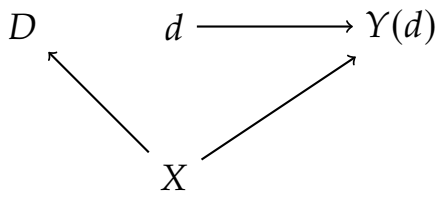
$$\delta(X) = \text{E}[Y(1) \mid X] - \text{E}[Y(0) \mid X].$$

Thus, the APE and ATE also agree:

$$\delta = \text{E}[\delta(X)] = \text{E}[\pi(X)] = \pi.$$

Conditional Ignorability via Causal Diagrams

It is possible to illustrate the key ignorability assumption, Assumption 5.2.1, graphically as follows:[5]



5: Note that what we present is just one of many casual diagrams that are compatible with the conditional ignorability condition. There are others, as will become apparent in subsequent chapters.

Figure 5.5: A Causal Diagram for the Conditional Ignorability Research Design

In this graph, we show the potential outcome $Y(d)$ as a node and the potential treatment status d as another node. The latter node is deterministic. There is an arrow from d to $Y(d)$ indicating the dependency. The pre-treatment covariates X affect both the realized treatment variable D and the potential outcomes $Y(d)$, as shown by the arrow from X to D and from X to $Y(d)$. The assigned treatment variable D is independent of the node $Y(d)$, conditional on X. Independence can be derived from the graph by observing the absence of any path between the D and $Y(d)$ nodes other than the path through the variable X upon which we've conditioned. Note that Assumption 5.2.2, the overlap condition, is not illustrated in the graph.

The potential outcome process $d \mapsto Y(d)$ and treatment assignment jointly determine the realized outcome variable Y via the assignment $Y := Y(D)$. This generates the following causal diagram. This graph says that X is generated first. D is then

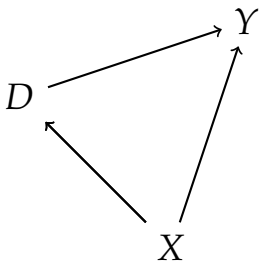


Figure 5.6: A Causal Diagram with Conditional Ignorability

generated, with the distribution of D depending on X. Finally, Y is generated, with its distribution depending on both D and X. Here, after conditioning on X, the statistical dependence (association) between D and Y only reflects the causal channel, $D \to Y$ allowing us to uncover the ATE, for example.

Connections to Linear Regression

The tools from Chapter 1 and Chapter 4 can be used to perform statistical inference on ATEs. We briefly discuss how (high-dimensional) regression can be used to retrieve causal estimates when conditional ignorability holds in this section.

The simplest instance of the problem is when the conditional expectation function of Y given D and X is linear,

$$\mathrm{E}[Y \mid D, X] = \alpha D + \beta' W,$$

which gives a model

$$Y = \alpha D + \beta' W + \epsilon, \quad \mathrm{E}[\epsilon \mid D, X] = 0.$$

Here it is understood that W may include X as well as pre-specified nonlinear transformations of X.

In this model, α identifies δ

$$\delta = \alpha$$

under the linearity assumption and ignorability, and our inference tools for α automatically carry over to δ. Note that the linearity assumption and ignorability assumptions imply that treatment effects are homogeneous; that is, $\delta(x) = \delta$ for all x in the support of X.

Of course, the assumption of linearity and homogeneous treatment effects is restrictive. A simple way to relax this is to consider interactions. One version of this approach takes all interactions between W and D and assumes

$$\mathrm{E}[Y \mid D, X] = \alpha_1 D + \alpha_2' W D + \beta_1 + \beta_2' W,$$

where we also maintain that we are working with centered covariates: $\mathrm{E}W = 0$.[6]

6: This model is still linear and results for linear models carry over to this case as well.

We then recover the ATE as

$$\delta = \alpha_1$$

and CATE as

$$\delta(X) = \alpha_1 + \alpha_2' W.$$

We can use partialling out methods, such as OLS in the low-dimensional case and Double Lasso (and variants) in

the high-dimensional case, to perform inference on α_1 and components of α_2. We can use these same methods to perform inference over β_1 and components of β_2, though these parameters will often not be of interest.

Note, we used this approach in the heterogeneous wage gap example in Chapter 1. The discussion of whether the wage gap analysis has a causal interpretation is given in the next causal inference chapter, Chapter 6.

As demonstrated in Theorem 5.2.1, the ultimate targets are the conditional expectation functions $\mathrm{E}[Y(d)|X]$ if our goal is to learn average causal effects under ignorability. This being our target makes the relevance of considering transformations $W = T(X)$ of X important as we would like to have the linear model provide a good approximation to these conditional expectation functions. See the discussion in "From Best Linear Predictor to Best Predictor" in Chapter 1. If the linear model is misspecified in the sense that it does not approximate the conditional expectation functions well, the estimated causal effects - e.g. α_1 in the interactive model - do not necessarily have any causal interpretation. This potential failure is a major reason we consider more flexible, modern machine learning methods.

What about fully nonlinear strategies? We will explore them in Chapter 10.

5.3 Identification Using Propensity Scores

The identification by conditioning approach requires being able to accurately model the "outcome process," i.e. the conditional expectation function $\mathrm{E}[Y \mid D, X]$. This conditional expectation function might correspond to a complicated real world process that is hard to model or approximate.

When the outcome process is hard to model, we might have a much better handle on the "treatment selection process," i.e. the propensity score:

$$p(X) = \mathrm{P}(D = 1 \mid X).$$

An alternative approach, known as the Horvitz-Thompson method [3], uses propensity score reweighting to recover aver-

ages of potential outcomes. Using the propensity score rather than identification by conditioning on X is a useful empirical strategy when X is high-dimensional and $p(X)$ is available or can be approximated accurately.[7] An example of a setting where the propensity score is known is a *stratified RCT*, which is an experiment where treatment is assigned at random with probability $p(X)$ to individuals with different observed covariates X. In this case, the treatment assignment probability $p(X)$ is exactly the propensity score.

7: An interesting example where the propensity score is not known but can be well-approximated is the examination in [4] of the causal effect of attendance at a particular school or group of schools relative to one or more alternative schools (e.g., "elite" vs. "non-elite" schools) in settings where matching algorithms are used to assign students to schools. In this example, we can think of these student assignment mechanisms as $p(X)$.

Theorem 5.3.1 (Horvitz-Thompson: Propensity Score Reweighting Removes Bias) *Under Conditional Ignorability and Overlap, the conditional expectation of an appropriately reweighted observed outcome Y, given X, identifies the conditional average of potential outcome $Y(d)$ given X:*

$$\mathrm{E}\left[Y\frac{1(D=d)}{\mathrm{P}(D=d|X)} \mid X\right] = \mathrm{E}[Y(d) \mid X]$$

Then, averaging over X identifies the average potential outcome:

$$\mathrm{E}\left[Y\frac{1(D=d)}{\mathrm{P}(D=d|X)}\right] = \mathrm{E}[Y(d)]$$

To prove this result, note

$$\begin{aligned}\mathrm{E}\left[Y\frac{1(D=d)}{\mathrm{P}(D=d|X)} \mid X\right] &= \frac{\mathrm{E}[Y1(D=d) \mid X]}{\mathrm{P}(D=d|X)} \\ &= \mathrm{E}[Y(d) \mid X]\frac{\mathrm{E}[1(D=d) \mid X]}{\mathrm{P}(D=d|X)} \\ &= \mathrm{E}[Y(d) \mid X],\end{aligned}$$

where we used conditional ignorability in the second equality.

As a consequence, we can identify average treatment effects by simple averaging of transformed outcomes:

$$\delta = \mathrm{E}[YH], \quad H = \frac{1(D=1)}{\mathrm{P}(D=1|X)} - \frac{1(D=0)}{\mathrm{P}(D=0|X)},$$

where H is called the Horvitz-Thompson transform. Similarly, we can identify conditional average treatment effects as a conditional average of transformed outcomes:

$$\delta(X) = \mathrm{E}[YH \mid X].$$

Note that propensity score reweighting reduces to the difference of means in the control and treatment groups when the propensity score is constant.

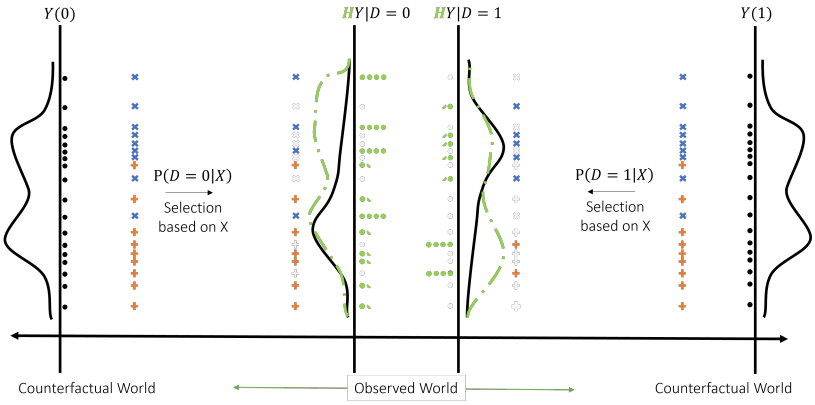


Figure 5.7: Pictorial representation of inverse propensity reweighting. As in Figure 5.4, the outer black curves represent the distribution of potential outcomes and the inner black curves represent the distribution of observed Y given only D – showing the selection bias. The green curves represent the observed distribution of HY given D which align and illustrate that the selection bias has been removed.

Stratified RCTs

In the case where the propensity score $p(X)$ is known, we are essentially back to a classical RCT.

Definition 5.3.1 (Generalized/Stratified RCT) *If under Assumption 5.2.1, the propensity score $p(X)$ is known, the setting is called a generalized or stratified RCT.*

Remark 5.3.1 Propensity score reweighting is generally not the most efficient approach to estimating treatment effects from a statistical point of view because it ignores any dependence between the outcomes and controls, X, that is not captured by the propensity score. By exploiting dependence between the outcomes and X not captured by the propensity score, more efficient estimation of treatment can occur as using this dependence "de-noises" the outcome. Moreover, estimation based on only propensity score reweighting fails under imbalances that might arise due to imperfect data collection. Later, we will use *both* regression and reweighting as part of "double machine learning" to operationalize efficient statistical inference on treatment effects in fully nonlinear (nonparametric) models.

Covariate Balance Checks

Given a propensity score $p(X)$, we can check if the RCT is valid (randomization is successful) by performing a *covariate balance*

check.. Specifically, conditional ignorability implies that

$$\mathrm{E}[H \mid X] = 0.$$

Thus, if covariates predict H, we can conclude that conditional ignorability does not hold. Heuristically, covariates predicting H means that covariates are imbalanced in the sense that, after reweighting by X dependent treatment probability, there are systematic differences in X across treatment and control observations which can be exploited to predict treatment assignment.

In a low-dimensional linear model framework, a covariate balance check can be done by regressing H on W, a dictionary of transformations of X, and testing if W predicts H. W predicting H suggests that the RCTs randomization protocol did not go as planned.

Connections to Linear Regression

Note that by the Horvitz-Thompson transform characterization of the CATE, $\delta(X) = \mathrm{E}[YH \mid X]$, we can view the conditional average treatment effect as the solution to a prediction problem of predicting the transformed outcome YH from the regressors X.

A useful strategy is to consider (potentially high-dimensional) linear regression models where HY is the dependent variable; see, e.g., [5]. Note that if we assume that $\mathrm{E}[Y \mid D, X] = \alpha_1 D + \alpha_2' WD + \beta_1 + \beta_2' W$, where W is a dictionary of transformations of X, then we have

$$\mathrm{E}[YH \mid X] = \alpha_1 + \alpha_2' W.$$

Thus, we can simply run a regression of YH on $(1, W')'$. In this regression model, we recover the ATE as

$$\delta = \alpha_1$$

and CATE as

$$\delta(X) = \alpha_1 + \alpha_2' W.$$

We can use partialling out methods, such as Double Lasso, to perform inference on α_1 and components of α_2. We also discuss estimating CATE using more general machine learning methods in Chapter 14 and Chapter 15.

5.4 Conditioning on Propensity Scores*

The fact that conditioning on the right set of controls removes selection bias has long been recognized by researchers employing regression methods. Rosenbaum and Rubin [6] made the much more subtle point that conditioning on only the propensity score

$$p(X) = \mathrm{P}(D = 1 \mid X)$$

also suffices to remove the selection bias.

Theorem 5.4.1 (Rosenbaum and Rubin: Conditioning on the Propensity Score Removes Selection Bias) *Under Ignorability and Overlap, D is generated independently of $Y(d)$ for each d, conditional on the propensity score $p(X)$: For each d,*

$$D \perp\!\!\!\perp Y(d) \mid p(X).$$

In other words, conditional on $p(X) = p$, variation in D is as good as randomly assigned. Hence, whenever it suffices to use X for identification by conditioning, it also suffices to use $p(X)$. This fact makes $p(X)$ a "minimal sufficient" statistic, conditioning on which removes selection bias under ignorability.

In scenarios with a known propensity score, we can simply use $p(X)$ as a control in place of the high-dimensional set of characteristics, X, and thus bypass a potentially complicated high-dimensional estimation problem. In other words, we can identify the conditional average potential outcome as

$$\mathrm{E}[Y(d) \mid p(X)] = \mathrm{E}[Y \mid D = d, p(X)].$$

Thus, it suffices to learn the CEF $\mathrm{E}[Y \mid D, p(X)]$. We learn good approximations of these CEFs by incorporating polynomials or other transformations of $p(X)$ to make things more flexible and running linear regression methods. Finally, we can also employ nonlinear machine learning methods introduced in Chapter 9 to overcome the limitations of linear models.

After controlling for $p(X)$, we can also consider the use of high-dimensional methods to include other transformations W of the raw variables X in order to improve precision, estimating the more flexible CEF $\mathrm{E}[Y \mid D, p(X), W]$. It is especially advisable to include transformations W that fail the covariate balance checks discussed in Section 5.3. Including W can reduce the selection bias (and, hopefully, set it equal to zero). In the reemployment experiment, for example, we observed that balance did not seem satisfied across age groups. Hence, further controlling for

age makes sense and results in modest changes to estimates of the treatment effect. Of course, there is no guarantee that controlling for observed covariates can overcome selection bias in compromised RCTs in general because unobserved covariates may be driving the bias.

Remark 5.4.1 ("Clever Covariate") Finally, we note that the simple OLS regression of Y on the single constructed regressor

$$\phi(D, X) := \frac{1(D = 1)}{\mathrm{P}(D = 1|X)} - \frac{1(D = 0)}{\mathrm{P}(D = 0|X)} = H$$

can be used to estimate the ATE. Specifically, for β the coefficient in the model $Y = \beta H + \varepsilon$ with $\varepsilon \perp H$, we have that the ATE is equal to $\mathrm{E}[\beta(\phi(1, X) - \phi(0, X))]$. This result holds even though the CEF function is not given by βH; see Section 5.B. As such, incorporating the technical regressor H in a linear regression model (without penalization if high-dimensional estimation tools are used) can be a good idea. This approach is referred to as the "clever covariate" approach in the literature [7, 8].

5.5 Average Treatment Effect for Groups and on the Treated

In addition to unconditional average treatment effects (ATE) or average treatment effects at specific values of the covariates $X = x$, we may be interested in average effects within specific subpopulations.

A leading example of an interesting subpopulation treatment effect is a group ATE (GATE):

$$\delta_G = \mathrm{E}[Y(1) - Y(0)|G = 1]$$

where G is a group indicator defined in terms of X's. For example, we might be interested in the effects of a training program among younger people, say between 18 and 30 years old ($G = 1(18 \leq \texttt{age} \leq 30)$); among people older than 30 years old (so $G = 1(30 < \texttt{age})$); and differences between these two groups.

We can immediately obtain the GATE using the identification

results above and the law of iterated expectations:

$$\begin{aligned}\mathrm{E}[Y(1) - Y(0)|G = 1] \\ &= \mathrm{E}[\mathrm{E}[Y|D = 1, X] - \mathrm{E}[Y|D = 0, X]|G = 1] \\ &= \mathrm{E}[HY|G = 1].\end{aligned}$$

That is, we can identify GATEs either by taking the difference in regression functions or applying propensity score reweighting of outcomes and then averaging over group G.

We next consider treatment effects for the subpopulation of treated units, the *average treatment effect on the treated* (ATET):[8]

$$\delta_1 = \mathrm{E}[Y(1) - Y(0) \mid D = 1].$$

8: Rather than ATET, some use the abbreviation AToT or ATT.

For example, consider training completion as a treatment, D, and X a vector of pre-treatment variables such that unconfoundedness holds. Consider the question:

- On average, how much more do trainees earn after going through the training program than they would have earned had they not gone through the program?

Note that this question is a counterfactual question as it requires us to compare outcomes for trainees in the treated state, where they receive training, and the unobserved control state, where they did not receive training. The ATET, δ_1, is the parameter that answers such questions about counterfactuals. The ATET is identified by

$$\mathrm{E}[\mathrm{E}[Y|D = 1, X] - \mathrm{E}[Y|D = 0, X] \mid D = 1]$$

similarly to what we had above. It is also possible to bypass the use of $\mathrm{E}[Y|D = 1, X]$ in this case; see Appendix 5.C for more details.

Study Problems

1. Use one or two paragraphs to explain conditioning and its use in learning treatment effects/causal effects in observational data and randomized trials where treatment probability depends on pre-treatment variables. This discussion should be non-technical as if you were writing an explanation for a smart friend with relatively little exposure to causal modeling.

2. Use one or two paragraphs to explain the propensity score reweighting approach for identification of average

treatment effects. This discussion should be non-technical as if you were writing an explanation for a smart friend with relatively little exposure to causal modeling.

3. Use one or two paragraphs to explain why group ATE and the ATE on the treated may be of interest in empirical work. This discussion should be non-technical as if you were writing an explanation for a smart friend with relatively little exposure to causal modeling.

5.A Rosenbaum-Rubin's Result

Recall the propensity score is

$$p(X) := \mathrm{P}(D = 1|X),$$

which is the probability of receiving treatment given X. A simple useful intermediate property is the balancing property of the propensity score which states that treatment is independent of X conditional on the propensity score:

$$D \perp\!\!\!\perp X \mid p(X) \quad \Leftrightarrow \quad \mathrm{P}(D = 1|X, p(X)) = \mathrm{P}(D = 1|p(X)).$$

This result follows simply from (i) $\mathrm{P}(D = 1|X, p(X)) = \mathrm{P}(D = 1|X) = p(X)$ and (ii) $\mathrm{P}(D = 1|p(X)) = \mathrm{E}[D = 1|p(X)] = \mathrm{E}[\mathrm{E}[D|X, p(X)]|p(X)] = \mathrm{E}[p(X)|p(X)] = p(X)$. This property underlies covariate balance checks.

We now turn to the theorem of Rosenbaum and Rubin. By Theorem 5.3.1 and the law of iterated expectations, we have that for any function of the form $g(y) = 1(y \leq t), t \in \mathbb{R}$:

$$\begin{aligned}
\mathrm{E}\left[g(Y(1)) \mid p(X)\right] &= \mathrm{E}[\mathrm{E}[g(Y(1))|X, p(X)]|p(X)] \\
&= \mathrm{E}[\mathrm{E}[g(Y(1))|X]|p(X)] \\
&= \mathrm{E}\left[g(Y)\frac{1(D=1)}{p(X)} \mid p(X)\right] \\
&= \mathrm{E}\left[g(Y)\frac{1(D=1)}{p(X)} \mid D = 1, p(X)\right] P(D = 1|p(X)) \\
&\quad + \mathrm{E}\left[g(Y)\frac{1(D=1)}{p(X)} \mid D = 0, p(X)\right] P(D = 0|p(X))
\end{aligned}$$

$$
\begin{aligned}
&= \mathrm{E}[g(Y) \mid D = 1, p(X)] \frac{P(D = 1 \mid p(X))}{p(X)} \\
&= \mathrm{E}[g(Y) \mid D = 1, p(X)] \\
&= \mathrm{E}[g(Y(1)) \mid D = 1, p(X)]
\end{aligned}
$$

where we use $P(D = 1 \mid p(X)) = p(X)$. We can similarly argue for the case of $d = 0$. Thus, the conditional distribution of $Y(1)$ does not depend on D, once we condition on $p(X)$, which verifies Theorem 5.4.1.

5.B Clever Covariate Regression

Here we show that if we care only about estimating the ATE, then it suffices to learn the BLP of the outcome Y using the single covariate

$$\phi(D, X) := H = \frac{1(D = 1)}{p(X)} - \frac{1(D = 0)}{1 - p(X)}.$$

We can then use this BLP model as a proxy for the CEF $\mathrm{E}[Y \mid D, p(X)]$. Specifically, we learn a decomposition $Y = \beta\phi(D, X) + \epsilon, \epsilon \perp \phi(D, X)$ by running OLS of Y on $\phi(D, X)$ and then use $\mathrm{E}[\beta(\phi(1, X) - \phi(0, X))]$ as the ATE. This approach, referred to in the literature as the "clever covariate" approach, was first proposed in [7] and further developed in [8].

Note that the random variable H satisfies

$$\mathrm{E}[f(D, X)H \mid X] = f(1, X) - f(0, X)$$

for any function $f(D, X)$.[9] Then, by Theorem 5.3.1 and orthogonality of ϵ in the BLP decomposition:

9: Verify this as a reading exercise.

$$
\begin{aligned}
\mathrm{E}[Y(1) - Y(0)] = \mathrm{E}[YH] &= \mathrm{E}[\beta\phi(D, X)H] \\
&= \mathrm{E}\left[\beta(\phi(1, X) - \phi(0, X))\right].
\end{aligned}
$$

Note that even though this approach allows us to identify the ATE, it does uncover the CATE $\mathrm{E}[Y(1) - Y(0) \mid X]$. The reason for the failure in learning the CATE is that the residual ϵ does not necessarily satisfy conditional orthogonality; i.e. we do not have $\mathrm{E}[(Y - \beta\phi(D, X))H \mid X] = 0$.

5.C Details of ATET

In observational studies, the ATET is identified under weaker conditions than the ATE because

$$\mathrm{E}[Y(1) \mid D = 1] = \mathrm{E}[Y \mid D = 1],$$

so we only need to identify $\mathrm{E}[Y(0) \mid D = 1]$. We can state the weaker version of the ignorability and overlap conditions as follows:

Assumption 5.C.1 (Ignorability and Overlap for Treated) *(a) Ignorability. Suppose that the treatment status D is independent of $Y(0)$ conditional on a set of covariates X, that is*

$$D \perp\!\!\!\perp Y(0) \mid X.$$

(b) Weak Overlap. Suppose that the propensity score satisfies:

$$\mathrm{P}(p(X) < 1) = 1.$$

Theorem 5.C.1 (Identification of ATET) *Under Assumption 5.C.1,*

$$\delta_1 = \mathrm{E}[Y \mid D = 1] - \mathrm{E}[\mathrm{E}[Y \mid X, D = 0] \mid D = 1].$$

Theorem 5.C.1 follows because, by iterated expectations and ignorability,

$$\begin{aligned}\mathrm{E}[Y(0) \mid D = 1] &= \mathrm{E}[\mathrm{E}[Y(0) \mid D = 1, X] \mid D = 1] \\ &= \mathrm{E}[\mathrm{E}[Y(0) \mid D = 0, X] \mid D = 1] \\ &= \mathrm{E}[\mathrm{E}[Y \mid D = 0, X] \mid D = 1],\end{aligned}$$

where the outer expectation is well-defined because the support of X conditional on $D = 1$ is a subset of the support of X conditional on $D = 0$ by the overlap condition.

The Horvitz-Thompson method can be also used to recover averages of potential outcomes for the treated. Indeed,

$$\frac{\mathrm{E}[DY]}{\mathrm{E}[D]} = \frac{\mathrm{E}[DY(1)]}{\mathrm{E}[D]} = \mathrm{E}[Y(1) \mid D = 1]$$

and

$$\begin{aligned}
\frac{\mathrm{E}\left[\frac{(1-D)}{1-p(X)}p(X)Y\right]}{\mathrm{E}[D]} &= \frac{\mathrm{E}\left[\frac{p(X)}{1-p(X)}\mathrm{E}[(1-D)Y \mid X]\right]}{\mathrm{E}[D]} \\
&= \frac{\mathrm{E}\left[\frac{p(X)}{1-p(X)}\mathrm{E}[(1-D)Y(0) \mid X]\right]}{\mathrm{E}[D]} \\
&= \frac{\mathrm{E}\left[\frac{p(X)}{1-p(X)}\mathrm{E}[1-D|X]\mathrm{E}[Y(0) \mid X]\right]}{\mathrm{E}[D]} \\
&= \frac{\mathrm{E}[p(X)\mathrm{E}[Y(0) \mid X]]}{\mathrm{E}[D]} \\
&= \frac{\mathrm{E}[\mathrm{E}[D \mid X]\mathrm{E}[Y(0) \mid X]]}{\mathrm{E}[D]} \\
&= \frac{\mathrm{E}[\mathrm{E}[DY(0) \mid X]]}{\mathrm{E}[D]} \\
&= \frac{\mathrm{E}[DY(0)]}{\mathrm{E}[D]} = \mathrm{E}[Y(0) \mid D=1]
\end{aligned}$$

where in the second to last step we used that $D \perp\!\!\!\perp Y(0) \mid X$, implies $\mathrm{E}[DY(0) \mid X] = \mathrm{E}[D|X]\mathrm{E}[Y(0) \mid X]$. Hence, we obtain the following result:

Theorem 5.C.2 (Propensity Score Reweighting for the Treated) *Under Assumption 5.C.1,*

$$\mathrm{E}[Y\bar{H}] = \delta_1, \quad \bar{H} = Hp(X)/\mathrm{E}[D].$$

Bibliography

[1] Merriam Webster Dictionary. *Compare apples and/to/with apples*. URL: https://www.merriam-webster.com/dictionary/compare%20apples%20and%2Fto%2Fwith%20apples (cited on page 126).

[2] Donald B. Rubin. 'Estimating causal effects of treatments in randomized and nonrandomized studies.' In: *Journal of Educational Psychology* 66.5 (1974), pp. 688–701 (cited on page 128).

[3] Daniel G. Horvitz and Donovan J. Thompson. 'A generalization of sampling without replacement from a finite universe'. In: *Journal of the American Statistical Association* 47.260 (1952), pp. 663–685 (cited on page 134).

[4] Atila Abdulkadiroğlu, Joshua D. Angrist, Yusuke Narita, and Parag A. Pathak. 'Research Design Meets Market Design: Using Centralized Assignment for Impact Evaluation'. In: *Econometrica* 85.5 (2017), pp. 1373–1432. DOI: 10.3982/ECTA13925 (cited on page 135).

[5] Victor Chernozhukov, Mert Demirer, Esther Duflo, and Iván Fernández-Val. *Generic machine learning inference on heterogeneous treatment effects in randomized experiments, with an application to immunization in India*. Tech. rep. National Bureau of Economic Research, 2018 (cited on page 137).

[6] Paul R. Rosenbaum and Donald B. Rubin. 'The Central Role of the Propensity Score in Observational Studies for Causal Effects'. In: *Biometrika* 70.1 (1983), pp. 41–55 (cited on page 138).

[7] Daniel O. Scharfstein, Andrea Rotnitzky, and James M. Robins. 'Adjusting for Nonignorable Drop-Out Using Semiparametric Nonresponse Models'. In: *Journal of the American Statistical Association* 94.448 (1999), pp. 1096–1120. (Visited on 01/25/2023) (cited on pages 139, 142).

[8] Heejung Bang and James M. Robins. 'Doubly Robust Estimation in Missing Data and Causal Inference Models'. In: *Biometrics* 61.4 (2005), pp. 962–973. DOI: https://doi.org/10.1111/j.1541-0420.2005.00377.x (cited on pages 139, 142).

Causal Inference via Linear Structural Equations

6

"the scientific [. . .] problem of causality is essentially a problem regarding our way of thinking, not a problem regarding the nature of the exterior world."

– Ragnar Frisch [1].

Here we present the linear structural equation model framework and causal diagrams. The advantage of these models is they are closely related to underlying structural models commonly used in economics and other fields. They allow for transparent derivation of the conditional ignorability assumption from the structure of the model. While linearity is imposed in this chapter, it will be dispensed with in later chapters.

6.1 Structural Equation Modelling and Conditional Exogeneity

Basic ideas that appeared in econometrics between the 20s and 40s (P. Wright [2], S. Wright [3], J. Tinbergen [4], T. Haavelmo [5]) provide another take on and language for causality that is closely related to the potential outcomes framework.

A Simple Triangular Structural Equation Model (TSEM)

We illustrate the basic ideas using a simple model of a household's (say weekly) demand for gasoline, motivated by Hausman and Newey [6].

We start with a log-linear (Cobb-Douglas [7]) model for log-demand y given the log-price p

$$y(p) := \delta p,$$

where δ is the elasticity of demand. Demand is random across households, and we may model this randomness as

$$Y(p) := \delta p + U, \quad \mathrm{E}[U] = 0, \tag{6.1.1}$$

where U is a stochastic shock that describes variation of demand across households (or across time, but assume that we are just looking at a particular time point). We immediately recognize that $Y(p)$ plays the same role as a potential outcome in Rubin's potential outcome model.[1]

The stochastic function

$$p \mapsto Y(p)$$

describes a household's log-demand at a given log-price p. The expected log-demand at log-price p is given by $\mathrm{E}[Y(p)] = \delta p$. The function encodes various structural causal effects: If we change p from p_0 to p_1, the expected demand change would be

$$\mathrm{E}[Y(p_1)] - \mathrm{E}[Y(p_0)] = \delta(p_1 - p_0).$$

Model (6.1.1) is very simple, and we may want to introduce covariates to capture other observable factors that may be associated with demand. That is, we may think there are observable parts of the stochastic shock, characterized by X, which help us predict household demand. Leading examples are household

1: The subtle difference here is that U does not depend on the index p, though we could make U be indexed by p at the cost of more complicated exposition. The distinction drawn is not superficial. Later on, when we discuss models with instruments, the dependence of U on p can create non-trivial problems which are not present in this section.

characteristics. For example, we may think demand is associated with features such as family size, income, number of cars, or geographical location. We can incorporate these features by modelling $U = X'\beta + \epsilon_Y$, where ϵ_Y is independent of X and has mean zero. Employing this model structure, we can write our augmented model as

$$Y(p) := \delta p + X'\beta + \epsilon_Y, \quad \epsilon_Y \perp\!\!\!\perp X. \tag{6.1.2}$$

Equation (6.1.2) is a structural stochastic model of economic outcomes. This model has nothing to do with regression or a statistical predictive model. Rather, it is a model that provides counterfactual predictions: If log-price is set to p, then a household with characteristics X can be predicted to purchase

$$\delta p + X'\beta$$

log-units. Here p is not a random variable – it is an index describing potential values of the price.

Then we ask the question:

- What data (Y, P, X) on quantities, prices, and characteristics should we collect to allow us to estimate the structural parameter δ?

Assumption 6.1.1 (Conditional Exogeneity) *(i) (Consistency) Suppose the observed variables* (Y, P, X) *are such that*

$$Y = Y(P)$$

i.e. the outcome is generated from the structural model, (ii) (Conditional Exogeneity) The observed P *is determined outside of the model, independently of* ϵ_Y *conditional on* X*:*

$$P \perp\!\!\!\perp \epsilon_Y \mid X \implies P \perp\!\!\!\perp \{Y(p)\}_{p\in\mathbb{R}} \mid X$$

Assumption 6.1.1 is the econometric analog of ignorability.[2] In the context of household demand, this condition requires that P is determined independently of a household's demand shock ϵ_Y, conditional on characteristics X. This assumption seems plausible for household level decisions, especially if we include geography in the set of covariates X.

2: At a general level, gasoline prices are determined by aggregate supply and demand conditions, with small local geographic adjustments (e.g., gasoline prices in areas with higher prices of land may be higher than in other areas to reflect the higher land costs for gasoline stations). Conditional on being in a given small geographic region, we may think of price fluctuations as independent of household-specific demand shocks.

If the conditional exogeneity condition holds, then

$$Y = Y(P) = \delta P + X'\beta + \epsilon_Y, \quad \epsilon_Y \perp (P, X).$$

This means that the projection parameters of Y on P and X coincide with the structural parameters δ and β.

We stress that our parameters δ and β are not defined by regression; they are defined by the model. Under the conditional exogeneity condition, these parameters coincide with the projection parameters.[3]

We might further postulate a structural equation for log-prices:

$$P(x) := x'\nu + \epsilon_P,$$

where $P(x)$ is the stochastic price process indexed by a household characteristics and ϵ_P describes the centered stochastic price shock. We assume that observed X is independent of price shock ϵ_P,

$$X \perp\!\!\!\perp \epsilon_P.$$

Independence between ϵ_P and observed X implies that ν coincides with the projection coefficient of P on X.

The price process $P(x)$ captures the belief that prices faced by households may differ depending on household characteristics. Note that this notation allows for only a subset of household characteristics to be systematically related to price; that is, we can have $P(x) = P(x_1)$ for some subvector x_1 of x. For example, it seems reasonable that households located in different regions would experience different prices, in which case x_1 could represent a household's geographic characteristics. Independence of the price shock ϵ_P from observed X may be plausible if household characteristics are determined well before gasoline prices faced by individual households in any specific time period are set.

Putting the equations together, we have a triangular structural equation model (TSEM):

$$\begin{aligned} Y &:= \delta P + X'\beta + \epsilon_Y, \\ P &:= X'\nu + \epsilon_P, \\ X&, \end{aligned} \tag{6.1.3}$$

where ϵ_Y, ϵ_P, and X are mutually independent (or at least uncorrelated) and determined outside of the model. They

3: A weaker starting condition than the conditional exogeneity condition for the above result is simply

$$(P, X) \perp \epsilon_Y.$$

That is, the observed P and X are orthogonal to the structural error ϵ_Y.

are called exogenous variables. Y and P are determined within the model and called the endogenous variables. The structural parameter δ can be identified by linear regression provided $\text{Var}(\epsilon_P) > 0$, and the structural parameter ν can be identified by linear regression provided $\text{Var}(X) > 0$.

Under the conditions stated above the parameters of these structural equations coincide with the projection parameters.

What do we mean by the model being structural? The term structural means that each of the equations is *assumed* to provide comparative statics and answers to counterfactual questions. Setting the right-hand-side variables to their potential values, we have

$$Y(p, x) := \delta p + x'\beta + \epsilon_Y,$$
$$P(x) := x'\nu + \epsilon_P.$$

The conceptual operation of "setting" or "fixing" the variables is supposed to leave the structure invariant. More generally, the structural parameters are supposed to be invariant to changes in the distribution of exogenous variables – X, ϵ_Y, ϵ_P – that have been generated outside of the model. Therefore, we can use these structural parameters to generate counterfactual predictions.

The jargon *comparative statics* refers to the determination of how endogenous variables change in response to changes in exogenous variables. Similarly, *counterfactual questions* coincide with asking how outcomes or endogenous variables change when variables are set to new values with other features of the model remaining fixed; e.g. asking how demand changes when price is set to some new value by a firm with household characteristics, price shocks, and demand shocks unaffected.

6.2 Drawing the Model: Causal Diagrams, aka DAGs

Sewall and Philip Wright [2], [3] would have depicted system of equations (6.1.3) graphically as a causal (path) diagram as in Figure 6.1. Observed variables are shown as nodes, causal paths are shown by directed arrows, and the structural (causal) parameters are given by the symbols placed next to the arrows.

The graph represents a structural economic model that can answer causal (comparative statics) questions. For example, the elasticity parameter δ tells us how household demand will respond to a firm *setting* a new price. Note that a firm setting a new price will not alter household characteristics or the other exogenous features of the model, and thus only the parameter δ is relevant for answering this question within the model.

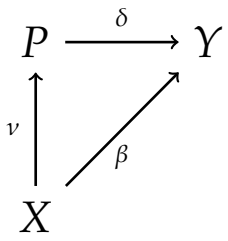


Figure 6.1: A simple causal diagram representation of the TSEM for the household gasoline demand example.

We could have expanded the previous graph to include unobserved shocks ϵ_P and ϵ_Y as follows:

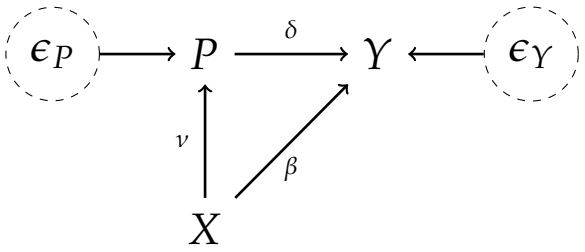


Figure 6.2: An expanded causal diagram representation of the TSEM that shows the unobserved shocks ϵ_P and ϵ_Y as root nodes.

The graph initiates with the *root nodes* ϵ_P, X, and ϵ_Y. The absence of links between the root nodes signifies the orthogonality between the nodes: namely, the absence of correlation. Understanding the orthogonality structure between nodes is an important input into identification of structural parameters via projection. The nodes X and ϵ_P are *parents* of P; the nodes P, X, and ϵ_Y are *parents* of Y. The node Y is a *collider* on all paths, because it contains only incoming arrows.

The main effect of interest is δ, which we call the structural causal effect of P on Y. This effect is identified after adjusting for X. In terms of the graph above, there are two paths connecting P and Y:

$$P \to Y \text{ and } P \leftarrow X \to Y.$$

The second path is called a *backdoor path* because there is an arrow pointing back to P from X. This connection indicates that there is a common cause for P and Y. Figuratively speaking, controlling or adjusting for X is said to be like "closing the backdoor path," shutting down the non-casual sources of statistical dependence between Y and P.

This visual characterization of the adjustment for X is due to J. Pearl [8] and generalizes to much more complicated graphs. We revisit these ideas throughout subsequent chapters.

How do household characteristics impact our model? X affects Y through two paths:

- the direct effect β via $X \to Y$,
- and the indirect effect $v\delta$ via $X \to P \to Y$.

The indirect effect is said to be "mediated" by P. We saw in Section 6.1 that we can identify δ and β from projection of Y on P and X, and we can identify v by projection of P on X. Therefore both the direct and indirect effects are identified.

Mediation structures appeared right at the outset in the Wrights' work [2], [3].

The total effect of X on Y is

$$v\delta + \beta,$$

which can be identified in this case by projection of Y on X. To verify this, we plug the first equation from the TSEM in (6.1.3) into the second equation producing

$$Y = (v\delta + \beta)'X + V; \quad V = \epsilon_Y + \delta\epsilon_P.$$

We see that the composite disturbance V is orthogonal to X,

$$V \perp X,$$

and, therefore, $(v\delta + \beta)$ coincides with the projection coefficient in the projection of Y on X. The latter point can be seen graphically: There are no "backdoor" paths from X to Y, so it is not necessary to adjust or control for anything to identify the total effect of X on Y.

In fact, while conditioning on P would allow us to identify the direct effect of X, β, it would prevent us from retrieving the total effect $v\delta + \beta$. In empirical practice, we may think of conditioning on P as "conditioning on the outcome," as P is determined by its parents, including X, so may be thought of as an outcome relative to X.

Remark 6.2.1 (Statistical Identification) Statistical identification typically relies on a combination of orthogonality or conditional independence restrictions and additional conditions – referred to as "rank conditions" in some settings – that ensure there is variation available for learning parameters of interest. For example, we need that $\text{Var}(\epsilon_P) > 0$ if we wish to learn δ in the TSEM in (6.1.3), and we need overlap for learning ATE as discussed in Chapter 5. Graphical methods provide a tool for representing orthogonality and conditional

independence relationships. They typically do not immediately reveal the additional rank-type conditions one would use in establishing statistical point identification. Examining the graphical structure does reveal what causal effects are potentially learnable within the structure, and additional restrictions, such as $\text{Var}(\epsilon_P) > 0$ in the TSEM, can then be deduced. Throughout the remainder of this book, we abstract away from rank-type conditions when discussing graphical models and talk about identifying parameters from the implied orthogonality or conditional independence structure.

To summarize, to learn a causal parameter, we must first define the causal parameter of interest and then carefully consider the choice of what to condition on to learn this effect. These choices are particularly important given the existence of *collider bias*.

Collider Bias R Notebook and Collider Bias Python Notebook provide a simple simulated example of collider bias based on the SEM (6.3.1).

6.3 When Conditioning Can Go Wrong: Collider Bias, aka Heckman Selection Bias

Consider the following SEM:

$$\begin{aligned} T &:= \epsilon_T \\ B &:= \epsilon_B \\ C &:= T + B + \epsilon_C \end{aligned} \tag{6.3.1}$$

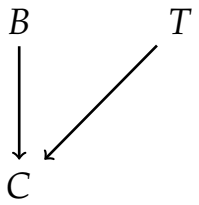


Figure 6.3: DAG with a collider representing SEM (6.3.1).

where ϵ_T, ϵ_B, and ϵ_C are independent $N(0,1)$ shocks. Here the average structural function for T, which does not depend on what values B might take, is zero,

$$\text{E}[T] = 0.$$

Regression without conditioning on C correctly identifies that T is not causally impacted by B:

$$\text{E}[T \mid B = b] = 0.$$

However, further conditioning on C removes the causal interpretation of the projection coefficient:[4]

$$\text{E}[T \mid B, C] = (C - B)/2; \implies \text{E}[\text{E}[T \mid B = b, C]] = -b/2 < 0.$$

This regression suggests that, controlling for C, the predictive effect of B on T is $-1/2$. This predictive effect is not a causal effect.

4: Dividing by 2 may seem counterintuitive, but it is correct. See Collider Bias R Notebook or Collider Bias Python Notebook for detail.

Collider bias illustrates that conditioning on outcomes may produce the wrong conclusions about causality, so conditioning on outcomes should be always approached with care. In econometrics, collider bias is known as a form of sample selection bias[5] ("conditioning on endogenous variables" or Heckman selection bias [9]).

5: J. Heckman was awarded the Nobel Memorial prize "for his development of theory and methods for analyzing selective samples." Source: Nobelprize.org

A Serious Digression on Colliders. Within our toy SEM framework, regression on a collider is clearly the wrong thing to do if one wants to identify the causal effect of B on T. However, we do note that regression on a collider can be *very useful* for other predictive tasks.

The following example draws on the discussion given in the "Book of Why" [10] to illustrate collider bias.

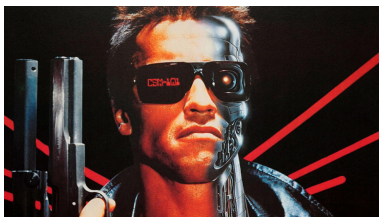


Figure 6.4: Our SEM predicts that this actor, A. Terminator, is (essentially) the most talented actor in Hollywood.

Example 6.3.1 (Structural Model of Hollywood) Suppose that the preceding SEM provides a cartoon depiction of people in Hollywood where T denotes acting talent, C denotes celebrity (i.e. success or popularity), and B denotes bonhomie (i.e. approachability or friendliness). Note that the SEM indicates that more talent and approachability cause more success. Further, for a person to remain in Hollywood, we would expect $C > 0$. As shown above, the causal effect of B on T in this SEM is 0. However, the best linear predictor of T given B conditional on $C > 0$ is

$$\approx .6 - B/4.$$

That is, bonhomie and talent are negatively correlated in Hollywood despite the fact that approachability does not causally impact talent. This correlation is useful for making predictions. For example, the individual depicted in the margin appears quite imposing and not approachable, perhaps with $B = -20$. We would then predict the expected value of his talent to be $t \in [+5.6 \pm 2]$, which is at least 3.6 standard deviations above the average talent of zero in the overall population within our model. From that, we should *predict* that this person is an incredibly talented actor but should not draw any conclusions about causality between B and T.

The example illustrates how simple theoretical models are often used in economics. Causal reasoning is made within a simple model, such as the SEM (6.3.1). This reasoning then leads to some testable restrictions, such as negative correlation between T and B conditional on $C > 0$. Even though we may not believe that the stylized model provides a complete model of reality, the

implications of the simple model provide some insight into how observed phenomena, such as a negative correlation between T and B conditional on $C > 0$, may arise. Such reversion of the correlation between two variables has been observed empirically in several cases, a prominent one being the birth-weight paradox [11] described below.

Example 6.3.2 (Birth-weight "paradox" [11]) In a study conducted in 1991 in the US, it was found that infants born to smokers had higher risk of low birth-weight (LBW) and higher risk of infant mortality than infants born to non-smokers. However, when looking at the sub-group of infants with LBW, the comparison is reversed and the risk of infant mortality is lower for infants born to smokers, than for infants born to non-smokers. How is that possible? Does smoking have a positive causal effect on infant mortality conditional on LBW?

A more plausible alternative explanation can be uncovered through the lens of SEMs and Causal Diagrams if one starts to think of competing risks and collider bias. Let's denote with S the smoking indicator, Y the infant death outcome, and B the low birth-weight indicator. We will also denote with U an abstract variable corresponding to the multitude of competing risks that can cause LBW. It is highly plausible that smoking is a risk factor for LBW and also has a direct effect on mortality. Moreover, LBW and the competing risk factors can also have a direct effect on mortality. Putting these factors together leads to the Causal Diagram depicted in Figure 6.5. In this setting, an infant with a smoking parent may be highly likely to have LBW caused by smoking. At the same time, LBW can be much less frequent for non-smoking parents. When we further focus in on the group of infants of non-smoking parents with LBW, it is highly probable that LBW was caused by some other competing risk which can adversely affect mortality. Thus, conditioning on LBW, we could essentially be comparing infants of smoking parents without competing risks to infants of non-smoking parents with competing risks.

To illustrate how the unconditional association between Y and S uncovers the true causal effect, while conditioning on B introduces bias and can even reverse the sign of the true effect, let's look at a simple linear SEM that corresponds to

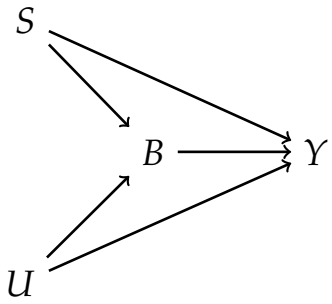


Figure 6.5: DAG with a collider representing low birth-weight "paradox" Example 6.3.2.

the causal diagram depicted in Figure 6.5:

$$\begin{aligned} Y &:= S + B + \kappa U + \epsilon_Y \\ B &:= S + U + \epsilon_B \\ S &:= \epsilon_S \\ U &:= \epsilon_U \end{aligned} \tag{6.3.2}$$

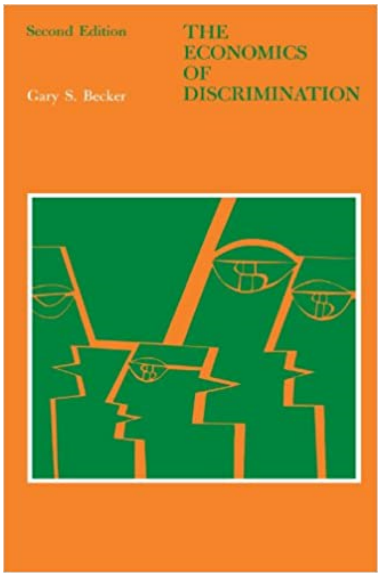
where $\epsilon_Y, \epsilon_B, \epsilon_S$ and ϵ_U are independent $N(0,1)$ shocks. Note that if we simply project Y on S, then we recover the correct positive causal effect of 2, since conditional exogeneity is satisfied. However, when we project Y on S and B, we learn a CEF of the form:

$$\begin{aligned} \mathrm{E}[Y \mid S, B] &= S + B + \kappa \mathrm{E}[U \mid S, B] \\ &= S + B + \kappa(B - S)/2 = (1 - \kappa/2)S + (1 + \kappa/2)B. \end{aligned}$$

If the competing risks increase infant mortality a lot, i.e. $\kappa \gg 1$, then this projection recovers an erroneous large negative(!) effect $1 - \kappa/2$ of smoking on mortality.

6.4 Wage Gap Analysis and Discrimination

> "The central question in any employment-discrimination case is whether the employer would have taken the same action had the employee been of a different race (age, sex, religion, national origin etc.) and everything else had remained the same." (In Carson versus Bethlehem Steel Corp., 70 FEP Cases 921, 7th Cir. (1996) [12]).



Wage regressions are widely used by labor economists to characterize the wage gap between men and women and to link the wage gap to discrimination; see, e.g., [13] and [14]. Some economists have asserted that it is wrong to study discrimination by doing wage gap regressions, e.g. [15], and that we should instead look at the unconditional difference in outcomes across groups. Their reasoning is based on the argument that key job characteristics – e.g., education and occupation – are determined in response to both a group identity and discrimination and are therefore (intermediate) outcomes. Controlling for these characteristics may then introduce a form of selection bias. Which of these two sets of economists is right?

In what follows, we present a simple SEM in (6.4.1), which postulates that different groups receive equal wages if there

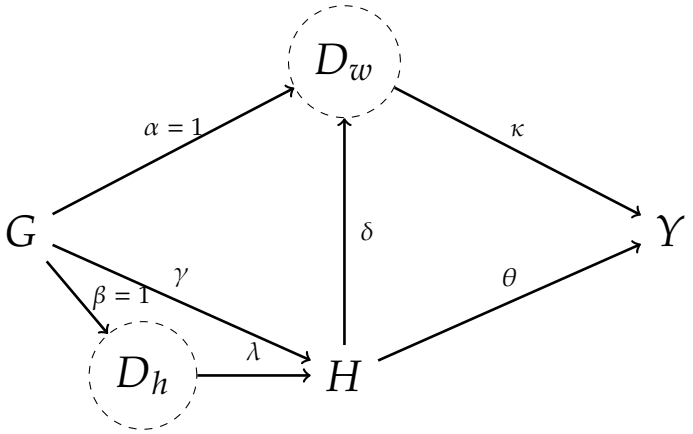


Figure 6.6: A Simple Model of Discrimination. Here G denotes a group (e.g., sex), H is human capital, and Y is the wage. D_w denotes unobserved wage discrimination occuring in the work place, and D_h denotes unobserved discrimination that occurs in the accumulation of human capital.

are no conditional productivity differences between the groups. We will see that, in this SEM, wage gap regressions do uncover well-defined discrimination effects that occur in wage-setting mechanisms. In contrast, the unconditonal average wage gap uncovers a more complicated causal object, which absorbs discrimination in wage setting, discrimination in human capital and occupational acquisitions, as well as group specific preferences for occupations.

Here we begin with the linear SEM and the equivalent DAG shown in Figure 6.6:

$$\begin{aligned} Y &:= \kappa D_w + \theta H + \epsilon_Y, \\ D_w &:= \alpha G + \delta H + \epsilon_{D_w}, \\ H &:= \gamma G + \lambda D_h + \epsilon_H, \\ D_h &:= \beta G + \epsilon_{D_h}, \\ G, & \end{aligned} \tag{6.4.1}$$

where the shocks ϵ_Y, ϵ_{D_w}, ϵ_H, ϵ_{D_h}, and G are all mean zero and uncorrelated.

The outcome Y is wage, G is group (e.g., sex), H is human capital (a scalar index that includes labor-relevant characteristics such as education, occupation, etc.),[6] D_w is latent wage discrimination arising in the work-place, and D_h is latent discrimination arising in acquisition of human capital. There could be other observed confounders that we don't show for the sake of simplicity.

6: H can be easily made a vector with a slightly more complicated notation.

The discrimination variables D_w and D_h are latent variables that are important for our model but cannot be directly observed. We maintain throughout that these variables are non-degenerate and related to group identity G. Under these assumptions,

the scale of these latent variables is non-zero but arbitrary, so we normalize the effect $G \to D_w$ to unity, $\alpha = 1$, and the effect $G \to D_h$ to unity as well, $\beta = 1$. There is no edge from G to Y, reflecting our assumption that there is no systematic group difference in productivity conditional on H and D_w. In the absence of productivity differences between workers, economic reasoning suggests that they would be assigned the same wage in a discrimination-free economy [16]. Thus, we would expect $\kappa = 0$ in a discrimination-free economy in the case that H captures all sources of productivity differences between workers.

Within this model, the parameter of interest is then the causal or structural effect of discrimination on wages given by

$$\kappa.$$

If $\kappa \neq 0$, we can conclude that wages are assigned unfairly within the framework of this SEM.

If we observed D_w directly, we could learn the effect of discrimination on wages, κ, by regression of Y on D_w and H. Identification of κ from this regression follows from the backdoor criterion discussed in Section 6.2. We don't observe D_w directly, but we postulate that this variable is determined only by G, H, and a stochastic shock. Dependence on H captures the idea that discrimination may be larger or smaller depending on education level, profession, etc. We return to using this additional structure to learn about κ below.

Discrimination may operate through channels other than simple wage differences. For example, in the 1960s, there were relatively few women or African American lawyers, a highly paid occupation. Discrimination that operates through occupational choice or human capital formation is captured by latent variable D_h. In our model, H, which captures productivity differences between individuals, can be determined as a result of both discrimination and group preferences.[7] The parameter γ then captures the effect of group preferences on the formation of H, while the effect of discrimination on H is captured by λ. Since D_h is not observed, there is no way to separately identify these two effects.

7: For example, 90% of firefighters in the US are men, which may reflect a genuine preference for this occupation among men. At the same time, even preference for occupation may be a result of cultural institutions that could themselves be interpreted as discriminatory in broader, cross-cultural, contexts.

It is easy to show, within the model, that the population linear regression of Y on G and H recovers the wage dis-

crimination effect,

$$\kappa,$$

and that the linear regression of H on G recovers

$$\gamma + \lambda,$$

the sum of the group preference effect and the human capital discrimination effect; see Appendix 6.A for details. If a further strong assumption is made that there is no group preference effect, $\gamma = 0$, the linear regression of Y on G recovers the total discrimination effect:

$$\kappa + \lambda(\kappa\delta + \theta).$$

Endogenous Sample Selection. There is an important issue with our empirical example. We are only able to look at earnings of people who are employed. Thus, we are conditioning on

$$Y > R,$$

where R is the reservation wage. In other words, we are conditioning on the outcome which may cause major selectivity issues: People get employed, and end up in our data, only if the offered wage is higher than some reservation wage. This sample selection on the basis of the outcome can cause major biases in the analysis. The potential for large biases was recognized by James J. Heckman [9] in the 70s and led to the development of the celebrated Heckman selection correction and related methods.

An alternate approach to applying a selection correction in our example is to select a subset S of people who are employed with probability one (or very close to one). For example, one could look at highly educated, unmarried people. Within this subset, we would then have

$$P(Y > R|S) \approx 1.$$

That is, the value of the wage offer, Y, is approximately unrelated to whether we observe individual wages for this subset of people. This type of strategy has been employed by Casey Mulligan and Yona Rubinstein [17]. Mulligan and Rubinstein continue to find evidence in favor of the existence of wage gaps in their analysis of a subsample where selection effects are likely small. This finding then

suggests that the broad conclusion of the existence of wage gaps is not driven entirely by sample selection issues.

In summary, we have the following observations:

- In general, wage gap regressions just estimate predictive effects or associations.
- When we assume a SEM like the one above holds and there are no endogenous sample selection effects, wage gap regressions estimate wage discrimination effects.
- Unconditional wage gaps generally reflect a combination of different types of discrimination and group preferences and thus do not isolate solely the effects of discrimination.

Figure 6.7: Early 20th century: The work of Sewall and Philip Wright made it possible for humans to begin to "fly" in the space of causal models. Another family of Wrights made it possible for humans to begin to fly in the air.

Notebooks

- Collider Bias R Notebook and Collider Bias Python Notebook provide a simple simulated example of collider bias, informing our discussion of conditioning on Celebrity in our Structural Model of Hollywood.

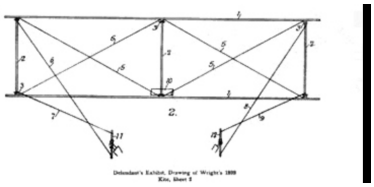


Figure 6.8: An early drawing for an airplane appears very much like an early drawing of a DAG.

Notes

This chapter presented an approach to causal inference that goes back to the works of Sewall and Philip Wright [2], [3], Tinbergen [4], Haavelmo [5], and others. This tradition lives in modern structural casual models used in econometrics (especially, industrial organization) and in the artificial intelligence community. The latter community, inspired by the foundational work of J. Pearl [8], strongly adopted the use of causal diagrams, known as directed acyclical graphs (DAGs). We continue exploring this approach throughout the remainder of our treatment on causal inference.

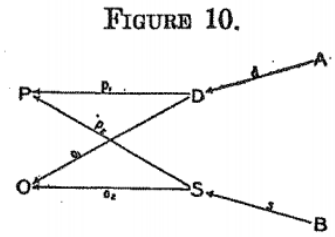


Figure 6.9: DAG for Supply-Demand Systems in P. Wright's work in 1928 [2].

Study Problems

1. Explain collider bias to a friend in simple terms. Use no more than two paragraphs. Illustrate your explanation using a simulation experiment.

2. Empirical: Revisit the group wage gap analysis from Chapter 4, focusing on college-educated workers. Is there a structural/causal interpretation for the estimated wage gap? Is there a group gap in education achievement? Does this group gap in education have a structural/causal interpretation? Some of these questions are open ended and have no simple answers, but it is useful to think about them. (If you have other data sets that might illuminate discrimination in other settings, please use them in place of the wage data set).

3. Free-style exercise: The model for wage discrimination presented in our notes is very stylized and subject to multiple criticisms. For example, it does not deal with promotion and hiring decisions. There are several interesting models of discrimination in hiring, college admissions, and pay. For example, see "The Book of Why"[10] and the Bickel et al. 1975 paper [18] for an analysis of Berkeley undergraduate admissions decisions. Nina Roussile's (2020) [19] paper isolates the ask gap as the central mechanism for the subsequent wage gap. Referring to one such analysis, draw or write down a linear structural causal model that captures the structural idea of the analysis and discuss identification in the model.

6.A Details of the Wage Discrimination Analysis

We write out some of the structural equations corresponding to our stylized DAG for discrimination (Figure 6.6):

$$\begin{aligned} Y &:= \kappa D_w + \theta H + \epsilon_Y, \quad \epsilon_Y \perp D_w, H, G \\ D_w &:= G + \delta H + \epsilon_{D_w}, \quad \epsilon_{D_w} \perp G, H \end{aligned}$$

where the orthogonality relations are implied by the model.

Linear regression analysis would use observable variables only, so we substitute the model for the unobserved D_w in terms of G and H into the equation for Y to obtain

$$Y = \kappa G + (\kappa\delta + \theta)H + U, \quad U := \kappa\epsilon_{D_w} + \epsilon_Y \perp (G, H).$$

The composite error term U is orthogonal to G and H. Therefore, regression of Y on G and H learns κ and $(\kappa\delta + \theta)$, with our main target being κ. We can also see that by partialling out H,

"This is elementary, my dear Watson," said Sherlock Holmes after seeing this.

$$\tilde{Y} = \kappa\tilde{G} + U, \;\; U \perp \tilde{G}.$$

Thus, κ is retrievable only if there is non-zero variation in $\tilde{G}$ after taking out the linear effect of H.

Now suppose we want to study discrimination effects in occupational choices, captured by H in our model. We write out the relevant structural equations:

$$\begin{aligned} H &:= \gamma G + \lambda D_h + \epsilon_H, \quad \epsilon_H \perp (G, D_h), \\ D_h &:= G + \epsilon_{D_h}, \qquad \epsilon_{D_h} \perp G. \end{aligned}$$

Recall that γ is the group preference effect and λ is the discrimination effect. Since D_h is not directly observed, we substitute it out to arrive at

$$H = (\gamma + \lambda)G + V; \quad V := \gamma\epsilon_{D_h} + \epsilon_H \perp G.$$

Therefore, $\gamma + \lambda$ is the projection coefficient in the projection of H on G. Hence, we can identify $\gamma + \lambda$, but we can't identify γ and λ separately.

Going further, suppose that the group preference effect is zero, so $\gamma = 0$. Then, the previous argument would identify λ and we could identify the total discrimination effect arising from two different channels:

$$\kappa + \lambda(\kappa\delta + \theta).$$

from the regression of Y on G.

We can assert that the unconditional difference in wages measures discrimination only if the group preference effect in determining H is zero ($\gamma = 0$). Of course, most economists would probably not agree with the assumption that $\gamma = 0$. Empirically, there are large differences in group composition among different professions. These differences likely reflect both discrimination and genuine preferences.

Bibliography

[1] R. Frisch. 'A Dynamic Approach to Economic Theory: Lectures by Ragnar Frisch at Yale University'. Frisch Archives, Department of Economics, University of Oslo. 1930 (cited on page 146).

[2] Philip G. Wright. *The Tariff on Animal and Vegetable Oils*. New York: The Macmillan company, 1928 (cited on pages 147, 150, 152, 160).

[3] Sewall Wright. 'Correlation and Causation'. In: *Journal of Agricultural Research* 20.7 (Jan. 1921), pp. 557–585 (cited on pages 147, 150, 152, 160).

[4] Jan Tinbergen. 'Bestimmung und Deutung von Angebotskurven Ein Beispiel'. In: *Zeitschrift für Nationalökonomie* 1.5 (1930), pp. 669–679 (cited on pages 147, 160).

[5] Trygve Haavelmo. 'The probability approach in econometrics'. In: *Econometrica* 12 (1944), pp. iii–vi+1–115 (cited on pages 147, 160).

[6] Jerry A. Hausman and Whitney K. Newey. 'Nonparametric estimation of exact consumers surplus and deadweight loss'. In: *Econometrica* 63.6 (1995), pp. 1445–1476 (cited on page 147).

[7] Charles W. Cobb and Paul H. Douglas. 'A Theory of Production'. In: *The American Economic Review* 18.1 (1928), pp. 139–165 (cited on page 147).

[8] Judea Pearl. *Causality*. Cambridge University Press, 2009 (cited on pages 151, 160).

[9] James J. Heckman. 'Sample selection bias as a specification error'. In: *Econometrica* 47.1 (1979), pp. 153–161 (cited on pages 154, 159).

[10] Judea Pearl and Dana Mackenzie. *The Book of Why*. Penguin Books, 2019 (cited on pages 154, 161).

[11] Sonia Hernández-Díaz, Enrique F Schisterman, and Miguel A Hernán. 'The birth weight "paradox" uncovered?' In: *American Journal of Epidemiology* 164.11 (2006), pp. 1115–1120 (cited on page 155).

[12] 'Carson v. Bethlehem Steel Corp.' In: 82 F.3d 157, 158, 7th Cir. (1996) (cited on page 156).

[13] Francine D. Blau and Lawrence M. Kahn. 'The gender wage gap: Extent, trends, and explanations'. In: *Journal of Economic Literature* 55.3 (2017), pp. 789–865 (cited on page 156).

[14] Sonja C. Kassenboehmer and Mathias G. Sinning. 'Distributional changes in the gender wage gap'. In: *ILR Review* 67.2 (2014), pp. 335–361 (cited on page 156).

[15] Elise Gould, Jessica Schieder, and Kathleen Geier. 'What is the gender pay gap and is it real'. In: *Economic Policy Institute* (2016) (cited on page 156).

[16] Gary S. Becker. *The Economics of Discrimination*. University of Chicago Press, 2010 (cited on page 158).

[17] Casey B. Mulligan and Yona Rubinstein. 'Selection, Investment, and Women's Relative Wages Over Time'. In: *Quarterly Journal of Economics* 123.3 (2008), pp. 1061–1110. DOI: 10.1162/qjec.2008.123.3.1061 (cited on page 159).

[18] Peter J. Bickel, Eugene A. Hammel, and J. William O'Connell. 'Sex Bias in Graduate Admissions: Data from Berkeley: Measuring bias is harder than is usually assumed, and the evidence is sometimes contrary to expectation.' In: *Science* 187.4175 (1975), pp. 398–404 (cited on page 161).

[19] Nina Roussille. 'The central role of the ask gap in gender pay inequality'. In: *URL: https://ninaroussille. github. io/files/Roussille_ askgap. pdf* 34 (2020), p. 35 (cited on page 161).

Causal Inference via Directed Acyclical Graphs and Nonlinear Structural Equation Models

7

"you are smarter than your data. Data do not understand causes and effects; humans do."

– Judea Pearl [1].

Here we explore a fully nonlinear, nonparametric formulation of causal diagrams and associated structural equation models. These provide a useful tool for thinking about structures underlying causal identification.

7.1 Introduction

The purpose of this module is to provide a more formal and general treatment of acyclic nonlinear (and nonparametric) structural equation models (SEMs) and corresponding causal directed acyclic graphs (DAGs). We discuss the concepts and identification results provided by Judea Pearl and his collaborators and by James H. Robins and his collaborators.

In 2011, J. Pearl was awarded the A.M. Turing award, the highest award in the field of Computer Science and Artificial Intelligence: "For fundamental contributions to artificial intelligence through the development of a calculus for probabilistic and causal reasoning." In the Biometrika 1995 article [2], J. Pearl presents his work as a generalization of the SEMs put forward by T.Haavelmo [3] in 1944 and others.

These models and concepts allow us to rigorously define structural causal effects in fully nonlinear models and obtain conditional independence relationships that can be used as inputs to establishing nonparametric identification from the structure of the causal DAGs alone.[1] Structural causal effects are defined as hypothetical effects of interventions in systems of equations. We discuss identification of effects of *do interventions* introduced by Pearl [2] and *fix interventions* introduced by Heckman and Pinto [4] and Robins and Richardson [5].[2] fix interventions induce counterfactual DAGs called SWIGs (Single World Intervention Graphs) and can recover the causal graphs we've seen in previous chapters.

1: We abstract away from rank-type conditions. See Remark 6.2.1.

2: Fix interventions also had appeared as part of do calculus in Pearl [2].

Whether causal effects derived from SEMs approximate policy or treatment effects in the real world depends to a large extent on the degree to which the posited SEM approximates real phenomena. In thinking about the approximation quality of a model, it is important to keep in mind that we will never be able to establish that a model is fully correct using statistical criteria. However, we may be able to reject a given model using formal falsifiability criteria – though not all models are statistically falsifiable – or contextual knowledge. Further, evidence for some causal effects inferred from SEMs can be provided by further use of explicit randomized controlled trials, though the use of experiments is not an option in many cases. Ultimately, contextual knowledge is often crucial for making the case that a given structural model represents real phenomena sufficiently well to produce credible estimates of causal effects when using observational data.

Notation

Consider a pair of random variables (or equivalently, random vectors) U and V with joint distribution probability (mass) function $\mathsf{p}_{UV}(u, v)$ at generic evaluation points (u, v). We will simply denote $\mathsf{p}_{UV}(u, v)$ by $\mathsf{p}(u, v)$ whenever there is no ambiguity. We will denote the marginal probability (mass) functions

by $\mathsf{p}_U(u)$ and $\mathsf{p}_V(v)$, or simply by $\mathsf{p}(u)$ and $\mathsf{p}(v)$. The random variables U and V are independent, which we denote as

$$U \perp\!\!\!\perp V,$$

if and only if the joint probability density (or mass) function $\mathsf{p}(u, v)$ can be factorized as

$$\mathsf{p}(u, v) = \mathsf{p}(u)\,\mathsf{p}(v)$$

or equivalently if and only if

$$\mathrm{E}[g(U)\ell(V)] = \mathrm{E}[g(U)]\mathrm{E}[\ell(V)]$$

for any bounded functions g and ℓ. This definition of independence implies the ignorability or exclusion results,

$$\mathsf{p}(u \mid v) = \mathsf{p}(u), \quad \mathsf{p}(v \mid u) = \mathsf{p}(v),$$

which follow from Bayes' law:

$$\mathsf{p}(u \mid v) = \frac{\mathsf{p}(u)\,\mathsf{p}(v)}{\mathsf{p}(v)}.$$

Conditional independence is defined similarly by replacing distributions and expectations with their conditional analogs. Appendix 7.B reviews some useful results on conditional independence.

7.2 From Causal Diagrams to Causal DAGs: TSEM Example

Formal causal nonlinear DAGs generalize linear parametric models to general nonparametric forms. Recall our previous discussion of a model for a household's log-demand for gasoline (Y), which is a function of log-price (p) and household characteristics (X). We can generalize the simple TSEM to a nonlinear DAG as follows.

Example 7.2.1 (TSEM) We have a system of triangular structural equations:

$$\begin{aligned} Y &:= f_Y(P, X, \epsilon_Y), \\ P &:= f_P(X, \epsilon_P), \\ X &:= \epsilon_X, \end{aligned} \tag{7.2.1}$$

where f's are said to be deterministic structural functions and $\epsilon_Y, \epsilon_P, \epsilon_X$ are structural shocks that are independent of each other. The dimension of structural shocks is not restricted. Also, note the independences:

$$\epsilon_Y \perp\!\!\!\perp (P, X), \quad \epsilon_P \perp\!\!\!\perp X.$$

A causal diagram depicting the algebraic relationship defining the TSEM in Example 7.2.1 is shown in Figure 7.1. The absence of edges between nodes encodes the model's independence restrictions. Thus, as before, we can see that we can view graphs as representations of independence relations in statistical models. The graph visually depicts independence restrictions and the propagation of information or structural shocks from root nodes to their children, grandchildren, and so forth.

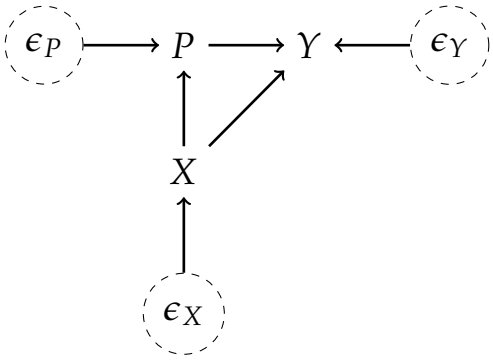


Figure 7.1: The causal DAG equivalent to the TSEM in Example 7.2.1.

It is also common to draw graphs based on only observed variables. We can erase the latent root nodes from Figure 7.1 to produce the equivalent diagram illustrated in Figure 7.2.

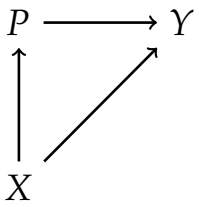


Figure 7.2: The causal DAG corresponding to the TSEM in Example 7.2.1 with latent root nodes erased.

The TSEM is purely a statistical model. We can view this model as structural under invariance restriction, following Haavelmo [3].

Definition 7.2.1 (Structural Form) *When we say that the TSEM is structural, we mean that it is defined by a structure made up of a set of stochastic processes:*

$$\begin{aligned} Y(p, x) &:= f_Y(p, x, \epsilon_Y), \\ P(x) &:= f_P(x, \epsilon_P), \\ X &:= \epsilon_X, \end{aligned}$$

indexed by $(p, x) \in \mathcal{P} \times \mathcal{X}$, called structural functions or structural potential outcome processes. Moreover,

- *(Exogeneity) Stochastic shocks ϵ_P, ϵ_X, and ϵ_Y are generated as independent variables outside of the model;*
- *(Consistency) The endogenous variables are generated by recursive substituations:*

$$Y := Y(P, X), \quad P := P(X), \quad X := \epsilon_X;$$

- *(Invariance) The structure remains invariant to changes of the distribution of stochastic shocks ϵ.*

The structure will be assumed to be preserved under various

interventions as defined below.

While SEMs are statistical models, assumptions akin to those in Definition 7.2.1 endow them with a structural meaning. Structural meaning may be generated by economic or other scientific reasoning. For example, structural functions may correspond to demand functions, supply functions, and expenditure functions, with these notions going back at least to Marshall [6] in the 19th century.

Remark 7.2.1 (Link to Potential Outcomes) Consider binary $p \in \{0, 1\}$ for simplicity. Consider potential outcomes, given by the structure:

$$Y(p, X) := g(p, X, \epsilon_Y(p)).$$

We can view potential outcomes through a SEM framework as follows. Let $\epsilon_Y := \{\epsilon_Y(p) : p \in \{0, 1\}\}$, then we have that

$$Y(p, X) = g(p, X, \epsilon_Y(p)) = f_Y(p, X, \epsilon_Y),$$

for

$$f_Y(p, x, e) := 1(p = 0)g(p, x, e(0)) + 1(p = 1)g(p, x, e(1))$$

for the argument $e = \{e(p) : p \in \{0, 1\}\}$. This example emphasizes that the dimensionality of ϵ's is not restricted in the general framework.

Identification by Regression

By conditioning on $X = x$ in the graph in Figure 7.1, we obtain the graph shown in Figure 7.3. We can equivalently express the relationship shown in Figure 7.3 in terms of equations as

$$Y(x) = f_Y(P(x), x, \epsilon_Y), \quad \epsilon_Y \perp\!\!\!\perp P(x).$$

If $P(x)$ is non-degenerate, we can further condition on $P(x) = p$ to learn the average structural function

$$\mathrm{E}[f_Y(p, x, \epsilon_Y)]$$

via regressions. We formally record this result as follows.

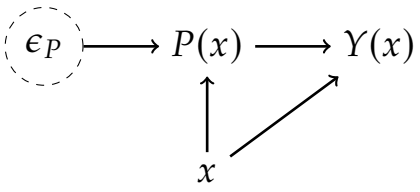


Figure 7.3: The graph produced from Figure 7.1 by conditioning on $X = x$. Here X is a parent to both P and Y. After conditioning, the remaining source of variation in $P(x)$ is ϵ_P. ϵ_P is determined exogenously – as if by an experiment – which allows measurement of the causal effect $P(x) \to Y$.

In the TSEM, the conditional average structural function

$$\mathrm{E}[f_Y(p, x, \epsilon_Y)]$$

can be identified by conditioning on P and X:

$$\begin{aligned}&\mathrm{E}[Y|P = p, X = x]\\ &\quad = \mathrm{E}[f_Y(P, X, \epsilon_Y)|P = p, X = x]\\ &\quad = \mathrm{E}[f_Y(p, x, \epsilon_Y)|P = p, X = x]\\ &\quad = \mathrm{E}[f_Y(p, x, \epsilon_Y)]\end{aligned}$$

provided the event $\{P = p, X = x\}$ is assigned positive density.

This average structural function has the interpretation as the expected outcome when P and X are exogenously set (set outside of the model as if by a policy maker or experiment) to $P = p$ and $X = x$.

Hence, we can use the average structural function to provide counterfactual predictions – predictions for the outcome under exogenous assignment of the policy variable P at fixed values for X. Within the TSEM, these counterfactual predictions align with the usual prediction rule $\mathrm{E}[Y|P = p, X = x]$.

If the confounder X is not observed, the causal relationship $P(x) \rightarrow Y$ is not identified.

If we can identify the conditional average structural function, we can also identify the conditional average structural causal effect:

$$\begin{aligned}&\mathrm{E}[f_Y(p_1,x, \epsilon_Y)] - \mathrm{E}[f_Y(p_0, x, \epsilon_Y)]\\ &\qquad = \mathrm{E}[Y|P = p_1, X = x] - \mathrm{E}[Y|P = p_0, X = x].\end{aligned} \tag{7.2.2}$$

The right hand side of (7.2.2) is a statistical quantity that can clearly be learned from data on Y, P, and X under reasonable assumptions. The left hand side of (7.2.2) defines a structural quantity of interest: the average effect of exogenously changing P from p_0 to p_1 at $X = x$.

Interventions

Do Interventions. The do operation $do(P = p)$ or do intervention corresponds to creating the counterfactual graph shown in Figure 7.4. On the graph, we remove P and replace it with a deterministic node p. In terms of equations (7.2.1) defining the TSEM, we replace the equation for P with p and then set P equal to p in the first equation. The corresponding counterfactual SEM is

$$\left(\begin{bmatrix} Y \\ P \\ X \end{bmatrix} : \text{do}(P = p)\right) := \begin{bmatrix} f_Y(p, X, \epsilon_Y) \\ p \\ X \end{bmatrix} = \begin{bmatrix} Y(p) \\ p \\ X \end{bmatrix}.$$

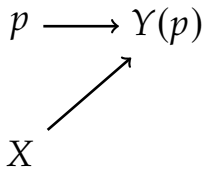


Figure 7.4: Causal DAG describing the counterfactual SEM induced by doing $P = p$.

The variables $Y(p)$ and X are the counterfactuals generated by the intervention $do(P = p)$. Note that the intervention keeps X and stochastic shocks ϵ_Y invariant.

The do operation has been extended to generate other types counterfactuals. For instance, another class of interventions are soft interventions[3] where the intervening variable is set to a value that is a function of its natural value (e.g., increasing a price by 10%). We could represent such interventions by the modified counterfactual SEM:

$$\left(\begin{bmatrix} Y \\ P \\ X \end{bmatrix} : \text{soft}_Y(P, \alpha)\right) := \begin{bmatrix} f_Y(\alpha(P), X, \epsilon_Y) \\ f_P(X, \epsilon_P) \\ X \end{bmatrix} = \begin{bmatrix} Y(\alpha(P)) \\ P \\ X \end{bmatrix}.$$

As an additional general example, we now consider *fix interventions* that induce single-world intervention graphs (SWIGs).[4]

Fix Interventions and SWIGs. Instead of removing P from the graph in Figure 7.2, we can split it into two nodes – P and a deterministic node p – where all the outgoing arrows from P are removed. The fixed node p then inherits the outgoing arrows from the original P.

The corresponding counterfactual SEM is

$$\left(\begin{bmatrix} Y \\ P \\ X \end{bmatrix} : \text{fix}_Y(P = p)\right) := \begin{bmatrix} f_Y(p, X, \epsilon_Y) \\ P \\ X \end{bmatrix} = \begin{bmatrix} Y(p) \\ P \\ X \end{bmatrix}.$$

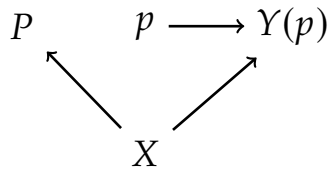


Figure 7.5: Causal DAG describing the counterfactual SEM induced by setting $P = p$ in the Y equation in (7.2.1) (formally a SWIG).

3: The ideas of constructing counterfactuals go back at least to P. Wright's work in 1928 [7], which involved replacing one structural equation with a different equation to define a counterfactual SEM. Specifically, Wright replaced the supply equation with another one reflecting a multiplicative tariff on the price that producers receive. This intervention is a (multiplicative) soft intervention. Building on P. Wright's work, soft interventions have been widely used in empirical economics (e.g., decomposition analysis of wages to study discrimination, carbon and emission taxes in environmental economics and industrial organization). See also [8, 9] for recent theoretical research in the computer science literature, framed in terms of DAGs and nonlinear ASEMs.

4: The fix intervention was introduced in Heckman and Pinto [4], as an extension of the do operation, and SWIGs were developed by Richardson and Robins [5].

The fix intervention merely says that we are setting $P = p$ in the Y equation. Figuratively speaking, it is a "localized do" operation. The variables $Y(p), P$, and X are the counterfactuals generated by this intervention. The intervention does not affect the P and X equations, nor does it affect ϵ_Y in the Y equation.

The SWIG allows us to immediately see that conditional exogeneity (ignorability) holds:

$$Y(p) \perp\!\!\!\perp P \mid X.$$

Therefore we can identify the counterfactual regression $\mathrm{E}[Y(p) \mid X]$ by the "factual" regression $\mathrm{E}[Y \mid P = p, X]$,

$$\mathrm{E}[Y(p) \mid X] = \mathrm{E}[Y(p) \mid P = p, X] = \mathrm{E}[Y \mid P = p, X],$$

invoking conditional independence and consistency arguments.

The do and fix interventions generate the same counterfactual distribution for $(Y(p), X)$, so the average causal effects of simple interventions coincide in the two approaches. However, the fix intervention creates a triple $(Y(p), X, P)$, which is useful for answering more complicated counterfactual questions.

For example, the counterfactual prediction $\mathrm{E}[Y(0) \mid P = 1]$ tells us what trainees ($P = 1$) would have earned on average, had they not gone through the training program ($p = 0$). In treatment effect analysis, this quantity is crucial for defining the average treatment effects for the treated:

$$\mathrm{E}[Y(1) \mid P = 1] - \mathrm{E}[Y(0) \mid P = 1].$$

Thus, the fix intervention allows us to seamlessly talk about conditional on P counterfactuals:[5]

$$\mathrm{E}\Big[Y(p) \mid P = \bar{p}\Big] := \mathrm{E}\Big[(Y \mid P = \bar{p}) : \mathrm{fix}_Y(P = p)\Big].$$

5: The same statement is formally not true with the do operation in place of the fix operation. Of course, one can also define these conditional counterfactuals by reverting to potential outcomes notation within causal DAGs; see [10].

7.3 General Acyclic SEMs and Causal DAGs

We will now turn to generalizing the concepts of the previous section from the TSEM case to general Directed Acyclic Graphs (DAGs) and the corresponding acyclic structural equation models (ASEMs).

DAGs and Acyclic SEMs via Examples

We now give a sequence of formal definitions, which can be easily understood by looking at just a single example.

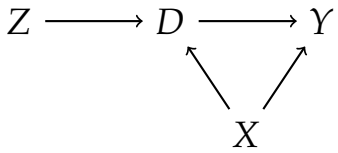


Figure 7.6: LS-DAG Example

Example 7.3.1 (Less Simple DAG (LS-DAG)) A directed acyclic graph (DAG) is a collection of nodes and directed edges with no cycles.

Consider the DAG in Figure 7.6: Here we can *say* that

- X is a parent of its children D and Y;
- D and Y are descendants of Z;
- There is a directed path from Z to Y;
- There are two paths from Z to X, but no directed path;
- D is a collider of the path $Z \to D \leftarrow X$;
- D is a noncollider of the path $Z \to D \to Y$;
- $Y \leftarrow X \to D$ is a backdoor path from Y to D.
- There are no cycles (there is no directed path that returns to the same node).

Example 7.3.2 (ASEM Corresponding to the LS-DAG) A system of triangular structural equations corresponding to Example 7.3.1 is

$$\begin{aligned} Y &:= f_Y(D, X, \epsilon_Y), \\ D &:= f_D(Z, X, \epsilon_D), \\ X &:= \epsilon_X, \\ Z &:= \epsilon_Z, \end{aligned}$$

where ϵ_Y, ϵ_X, ϵ_D, and ϵ_Z are mutually independent.

Factual distributions in DAG models have a beautiful Markov factorization structure, which allows for a simple representation of the joint distribution of all variables.

Example 7.3.3 (Factual Law in LS-DAG) Noting the dependences of each variable in the LS-DAG, we can write the joint distribution (density) p of Y, D, X, Z as

$$\mathsf{p}(y, d, x, z) = \mathsf{p}(y|d, x)\,\mathsf{p}(d|x, z)\,\mathsf{p}(x)\,\mathsf{p}(z).$$

Indeed,

$$\mathsf{p}(y, d, x, z) = \mathsf{p}(y|d, x, z)\,\mathsf{p}(d, x, z),$$

by Bayes' law. Then $\mathsf{p}(y|d, x, z) = \mathsf{p}(y|d, x)$ as the distribution

of Y is independent of Z, given its parents D and X. Further, $\mathsf{p}(d,x,z) = \mathsf{p}(d|x,z)\,\mathsf{p}(x,z)$, by Bayes' law, and $\mathsf{p}(x,z) = \mathsf{p}(z)\,\mathsf{p}(x)$ by independence.

General DAGs

The purpose of the rest of this section is to give concise general definitions.

A graph G is an ordered pair (V,E), where $V = \{1,...,J\}$ is a collection of vertices/nodes and E is a matrix of edges $e_{ij} \in \{0,1\}$ – that is, $E = \{e_{ij} : (i,j) \in V^2\}$.

Given a collection of random variables $X = (X_j)_{j\in V}$, we associate each index j with the name "X_j" whenever convenient. If the edge (i,j) is present, namely $e_{ij} = 1$, we read it as

$$\text{"}X_i \to X_j\text{" or "}X_i \text{ is an immediate cause of } X_j\text{."}$$

Consider a strict partial order $<$ on V induced by E, where $X_j < X_k$ (we read this as "X_j is determined before X_k") means that either $X_j \to X_k$ or $X_j \to X_{v_1} \to ...X_{v_m} \to X_k$ is true for some v_ℓ's in V. A partial ordering of V exists if for each j the statement $X_j < X_j$ is not true. [6] Note that we may interchangeably use random variable names, X_ℓ, or their indices ,ℓ, when referring to nodes in the graph.

6: The latter statement means that there are no *cycles*.

Definition 7.3.1 (DAG) *The graph $\mathsf{G} = (V,E)$ is a DAG if the graph has no cycles, that is, if V is partially ordered by the edge structure E.*

Example 7.3.4 (LS-DAG continued) In our example (Example 7.3.1), we had vertices $V = \{1,2,3,4\}$ identified with Y, D, X, Z, and the edge set

$$E = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

The partial ordering is $X < D$, $X < Y$, $Z < D$, $D < Y$.

Definition 7.3.2 (Parents, Ancestors, Descendants on a DAG) *The parents of X_j are the set $Pa_j := \{X_k : X_k \to X_j\}$. The children*

of X_j are the set $Ch_j := \{X_k : X_j \to X_k\}$. The ancestors of X_j are the set $An_j := \{X_k : X_k < X_j\} \cup \{X_j\}$. The descendants of X_j are the set $Ds_j := \{X_k : X_k > X_j\}$.

Definition 7.3.3 (Paths and Backdoor Paths on DAGs) *A directed path is a sequence $X_{v_1} \to X_{v_2} \to ...X_{v_m}$. A non-directed path is a path, where some arrows (but not all) arrows are replaced by $\leftarrow$. A collider node is a node X_j such that $\to X_j \leftarrow$. A backdoor path from X_l to X_k is an undirected path that starts at X_l and ends with an incoming arrow $\to X_k$.*

From DAGs to ASEMs

Every causal DAG implicitly defines a nonparametric acyclic structural equation model. Thus the two objects are simply different representations or views of the same assumptions on the data generating process and the stochastic potential or counterfactual outcome processes. DAGs are simply a visual depiction of ASEMs and ASEMs are simply a structural equation based expression of DAGs.

Definition 7.3.4 (ASEM) *The ASEM corresponding to the DAG $\mathsf{G} = (V, E)$ is the collection of random variables $\{X_j\}_{j \in V}$ such that*

$$X_j := f_j(Pa_j, \epsilon_j), \quad j \in V,$$

where the disturbances $(\epsilon_j)_{j \in V}$ are jointly independent.

Definition 7.3.5 (Linear ASEM) *The linear ASEM is an ASEM where the equations are linear:*

$$f_j(Pa_j, \epsilon_j) := f_j' Pa_j + \epsilon_j;$$

here we identify functions $\{f_j\}$ with coefficient vectors $\{f_j\}$.

In linear ASEMs we may replace the requirement of independent errors by the weaker requirement of uncorrelated errors.

Definition 7.3.6 (Structural/Potential Response Processes) *The structural potential response processes for the ASEM corresponding to the DAG $\mathsf{G} = (V, E)$ are given by the structure:*

$$X_j(pa_j) := f_j(pa_j, \epsilon_j), \quad j \in V,$$

viewed as stochastic processes indexed by the potential parental values pa_j.

Definition 7.3.7 (Consistency) *The observable variables are generated by drawing $\{\epsilon_j\}_{j\in V}$ and then solving the system of equations for $\{X_j\}_{j\in V}$.*

The stochastic shocks $\{\epsilon_j\}_{j\in V}$ are called exogenous variables, and the variables $\{X_j\}_{j\in V}$ are called endogenous variables. Endogenous variables are determined by the model equations, while exogenous variables are not.

The joint distribution of variables in ASEMs is generally characterized as follows:

Theorem 7.3.1 (Factual Law via Markovian Factorization) *The general ASEM model, given by $(X_j)_{j\in V}$ with an associated DAG $\mathsf{G}(V, E)$, obeys the following equivalent properties:*

- ***Factorization**: The law admits factorization:*

$$\mathsf{p}(\{x_\ell\}_{\ell\in V}) = \prod_{\ell\in V} \mathsf{p}(x_\ell \mid pa_\ell).$$

- ***Local Markov Property**: All variables are independent of their non-descendants given their parents.*

Counterfactuals Induced by Interventions

We next discuss counterfactuals generated by interventions. We first consider counterfactuals in the Less Simple DAG example (Example 7.3.1). Note that we use the abbreviation "CF" to denote "counterfactual."

Example 7.3.5 (CF-ASEM Induced by Do for LS-DAG Example) Consider the ASEM from Example 7.3.1. A counterfactual system induced by $do(D = d)$ is

$$\begin{aligned} Y(d) &:= f_Y(X, d, \epsilon_Y), \\ d&, \\ Z &= \epsilon_Z, \\ X &= \epsilon_X, \end{aligned}$$

where $\epsilon_X, \epsilon_Z, \epsilon_Y$ are mutually independent. The corresponding graph, provided in Figure 7.7, is denoted by $\mathsf{G}(d)$.

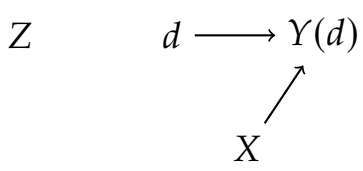


Figure 7.7: CF LS-DAG induced by $do(D = d)$ intervention.

Example 7.3.6 (CF-ASEM Induced by Fix for LSDAG Example) Consider the ASEM from Example 7.3.1. A counterfactual SEM induced by fix$(D = d)$ takes the following form:

$$\begin{aligned} Y(d) &:= f_Y(X, d, \epsilon_Y), \\ d, & \\ D &:= f_D(X, Z, \epsilon_D), \\ Z &:= \epsilon_Z, \\ X &:= \epsilon_X, \end{aligned}$$

where $\epsilon_X, \epsilon_Z, \epsilon_D, \epsilon_Y$ are mutually independent. The corresponding graph, provided in Figure 7.8, is denoted by $\widetilde{\mathsf{G}}(d)$.

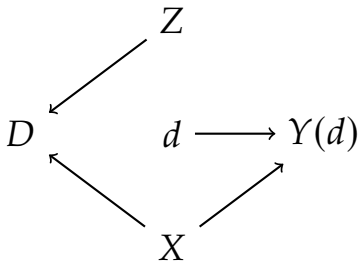


Figure 7.8: CF LS-DAG (SWIG) induced by the $fix_Y(D = d)$ intervention.

We now give a more general definition.

Definition 7.3.8 (Counterfactual ASEM induced by Do Intervention) *The intervention $do(X_j = x_j)$ on an ASEM is said to create the CF-ASEM defined by the modified graph*

$$\mathsf{G}(x_j) = (V, E^*)$$

and collection of counterfactual variables

$$(X_k^*)_{k \in V}$$

where

- *the edges incoming to the node j are set to zero, namely $e_{ij}^* = 0$ for all i,*
- *the remaining edges are preserved, namely $e_{ik}^* = e_{ik}$, for all i and $k \neq j$, and*
- *the counterfactual random variables are defined as*

$$\begin{aligned} X_k^* &:= f_k(Pa_k^*, \epsilon_k), \text{ for } k \neq j. \\ X_j^* &:= x_j \end{aligned}$$

where Pa_k^* are parents of X_k^* $(k \neq j)$ under E^*.

The do intervention modifies the graph G to $\mathsf{G}(x_j)$ by removing edges. Pearl [10] has described this process as "surgery."[7] We next define the do notation to mean

$$\Big((X_\ell)_{\ell \in V} : \mathrm{do}(x_j)\Big) := (X_\ell^*)_{\ell \in V}.$$

7: This sounds a bit painful.

Definition 7.3.9 (Counterfactual ASEM induced by Fix Intervention) *The intervention* $\text{fix}(X_j = x_j)$ *on an ASEM is said to create the CF-ASEM defined by the modified SWIG*

$$\tilde{\mathsf{G}}(x_j) := (\tilde{V}, \tilde{E}),$$

and collection of counterfactual variables

$$(X_k^*)_{k \in V} \cup (X_a^*)$$

where we split the node X_j *into* $X_j^* := X_j$ *and the new deterministic node* a

$$X_a^* := x_j,$$

where

- *the node* X_a *inherits only outgoing edges from* X_j *and no incoming edges; namely* $\tilde{e}_{ai} = e_{ji}$ *for all* i *and* $\tilde{e}_{ia} = 0$ *for all* i*;*
- *the node* X_j^* *inherits only incoming edges from* X_j *and no outgoing edges, namely* $\tilde{e}_{ij} = e_{ij}$ *for all* i *and* $\tilde{e}_{ji} = 0$ *for all* i*;*
- *all the remaining edges are preserved, namely* $\tilde{e}_{ik} = e_{ik}$*, for all* i *and* $k \neq j$ *and* $k \neq a$*; and*
- *the counterfactual random variables are assigned according to*

  $$X_k^* := f_k(Pa_k^*, \epsilon_k), \text{ for } k \neq a,$$

  where Pa_k^* *are parents of* X_k^* $(k \neq j)$ *under* $\tilde{E}$*.*

Intervention induces new counterfactual distributions for the endogenous variables; see Appendix 7.A for details.

7.4 Testable Restrictions and d-Separation

Next we examine the constraints on the data generating process that are implied by a given DAG.

For this we turn to a fundamental theorem in DAGs. We will define the concept of d-separation and prove that d-separation implies conditional independence. This property is typically referred to as a global Markov condition that is implied by the DAG. In order to define this property, we need a few more definitions.

The "d" here denotes "directional" as the direction of arrows in a DAG is important for understanding conditional independence relations; see, e.g., Pearl [10] Chapter 11.

Definition 7.4.1 (Blocked Paths) *A path π is is said to be blocked by a subset of nodes S if and only if*

(a) π contains a chain $i \to m \to j$ or a fork $i \leftarrow m \to j$ such that m is in S;
(b) Or, π contains a collider $i \to m \leftarrow j$, where neither m nor any descendant of m is in S.

A path that is not blocked is called open.

In Figure 7.9 the (backdoor) path $Y \leftarrow X \to D$ is blocked by $S = X$.

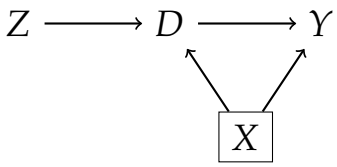


Figure 7.9: The path $Y \leftarrow X \to D$ is blocked by conditioning on X.

The following definition allows empty sets as conditioning sets.

Definition 7.4.2 (Opening a Path by Conditioning) *A path containing a collider is opened by conditioning on it or its descendant.*

In Figure 7.10 the path $Y \to C \leftarrow D$ is blocked, but becomes open by conditioning on the collider $S = C$.

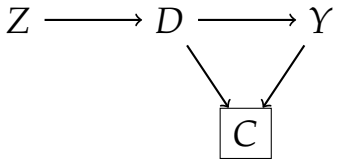


Figure 7.10: The path $Y \to C \leftarrow D$ is blocked, but becomes open by conditioning on C.

The following defines a key graphical property of DAG, which can be used to deduce key statistical independence restrictions.

Definition 7.4.3 (d-Separation) *Given a DAG G, a set of nodes S d-separates nodes X and Y if nodes in S block all paths between X and Y. d-separation is denoted as*

$$(Y \perp\!\!\!\perp_d X \mid S)_{\mathsf{G}}.$$

The following is a fundamental result concerning the conditional independence relations encoded in the graphs.

Theorem 7.4.1 (Verma and Pearl [11]; Conditional Independence from d-Separation) *d-Separation implies conditional independence:*

- ***Global Markov:*** $(Y \perp\!\!\!\perp_d X \mid S)_{\mathsf{G}} \implies Y \perp\!\!\!\perp X \mid S$.

Figuratively speaking, conditioning on S breaks the information flow between Y and X, meaning that Y can't be predicted by X, conditional on S, and vice versa.

This fundamental result is very intuitive and can be verified directly in simple examples. However, the formal proof is

difficult. The reverse implication is not true in general, but is argued to hold "generically" as we discuss in Section 7.5.

Example 7.4.1 We show a couple of examples illustrating that d-separation implies conditional independence:

1. In Figure 7.11, the variables X and Y are d-separated by $S = (Z, U)$, because S blocks all paths between X and Y. We also have Y is independent of X conditional on S: By the Markov factorization property, $\mathsf{p}(y, x \mid u, z) = \mathsf{p}(y \mid x, z, u)\,\mathsf{p}(x \mid z, u) = \mathsf{p}(y \mid u, z)\,\mathsf{p}(x \mid z, u)$. This equality provides a testable restriction.

2. In Figure 7.12, the variables X and Y are d-separated by $S = Z$, because S blocks all paths between X and Y. We also have Y is independent of X conditional on S: By the Markov factorization property, $\mathsf{p}(y, x \mid z) = \mathsf{p}(y \mid z)\,\mathsf{p}(x \mid z)$. This equality provides a testable restriction.

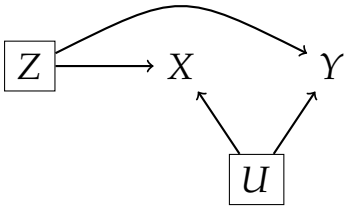


Figure 7.11: Example of d-separation.

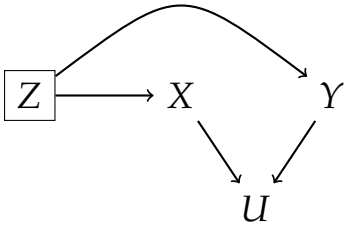


Figure 7.12: Example of d-separation.

These testable restrictions are called exclusion restrictions in econometrics because

$$Y \perp\!\!\!\perp X \mid Z \text{ is equivalent to } \mathsf{p}(y \mid x, z) = p(y \mid z), \tag{7.4.1}$$

where the equivalence follows from Bayes' law. In particular,

$$\mathrm{E}[g(Y) \mid X, Z] = \mathrm{E}[g(Y) \mid Z] \tag{7.4.2}$$

for any bounded function g of Y. (7.4.2) means that X is excluded from the best predictor of $g(Y)$ using X and Z. There are many tests of such restrictions available in the literature.[8] Perhaps one of the reasons for which there are many such tests is that conditional independence testing is formally impossible; see [12]. In practice, the formal impossibility means that any test must be carefully crafted to target specific features within a statistical model as no generic, uniformly valid testing procedure exists.

8: E.g, the reader can search Google Scholar for conditional independence tests, exclusion restrictions tests, or conditional moment tests.

With specific structure provided, conditional independence testing can be relatively straightforward. For example, it reduces to testing hypotheses about linear regression coefficients within a linear ASEM.

Implementation of Tests in Linear ASEMs. Consider the hypothesis that Y is independent of X, given Z. In linear ASEMs, we can test this hypothesis by testing whether the

coefficient $\alpha = 0$ in the projection equation

$$Y = \alpha' X + \beta' Z + \epsilon, \epsilon \perp Z.$$

We can perform this test easily with the tools we've developed so far. See R: Dagitty Notebook and Python: Pgmpy Notebook for an example.

Such tests are of course available under structures that are more general than in linear model. For example, [12] exploits debiased ML ideas (introduced in Chapter 4 and further developed in Chapter 10 in this text) to set up testing of exclusion restrictions in some nonlinear models.

Remark 7.4.1 (Equivalence of Local and Global Markov Properties) The local Markov property, the Markov factorization, and the global Markov property are equivalent (Pearl [10]). Therefore, one can use any of these properties to set up tests of the validity of the Markov structure.

7.5 Falsifiability and Causal Discovery*

Here, we provide a brief discussion of whether it is possible to falsify (reject) a causal structure encoded by a DAG with data.

Equivalence Classes and Falsifiability

Definition 7.5.1 (Equivalence Classes) *The class of DAGs that induce the same joint distribution of variables is called an equivalence class, and members of an equivalence class may be described as Markov equivalent. DAGs that produce the same joint distribution variables cannot be distinguished from each other.*

Pearl [10] shows that the equivalence class of a DAG is given by reversing any edges such that any such reversal does not destroy existing or create new *v-structures*: converging arrows whose tails are not connected by an edge.

The equivalence classes of a DAG are called PDAGs (partially directed acyclic graphs). We plot them by erasing arrowheads that can be oriented in the opposite direction without adding or removing v-structures. We illustrate PDAGs in Figures 7.13 and 7.14.

Figure 7.13 starts with the triangular structural equation model from Example 7.2.1. Figure (a) is the original DAG implied by the model. To produce the PDAG, shown in (b), we consider reversing each of the arrows from X to Y, X to P, and P to Y. Because each of the nodes is connected, there are no v-structures in the original DAG, and there is similarly no possible reversal that could add a v-structure. As such, the PDAG is simply the original DAG with all arrows removed. In this case, the DAG structure produces no testable implications.

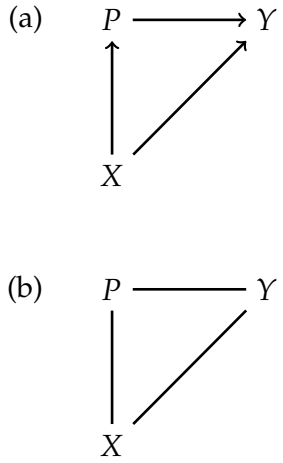


Figure 7.13: The original DAG, (a), and the equivalence class or PDAG, (b), for the TSEM example, Example 7.2.1. The undirected edges in the PDAG mean that they can be directed in any direction as long as this does not create a cycle. In empirical analysis directionality must therefore be deduced and assumed from the context.

Figure 7.14 starts from a more elaborate DAG than the simple TSEM. We refer to this DAG as "Pearl's Example" because it shows up repeatedly as an illustration in Pearl's work; see, e.g., [2]. Figure (a) is the original DAG defining the model. We produce the PDAG in (b) by considering the reversal of all combinations of arrows connecting the eight nodes. Here, there are only two reversals, changing $Z_2 \to X_3$ to $Z_2 \leftarrow X_3$ and changing $Z_1 \to X_1$ to $Z_1 \leftarrow X_1$, that do not destroy any existing v-structures or create new v-structures. For example, reversing the arrow $Z_2 \to X_2$ would destroy the v-structure $Z_2 \to X_2$ and $Z_1 \to X_2$. As such, the PDAG in (b) is almost identical to the DAG in (a) with the exception that the arrows between Z_2 and X_3 and between Z_1 and X_1 have been removed. In this case, the DAG encodes a model which includes exclusion restrictions or testable implications and is potentially falsifiable.

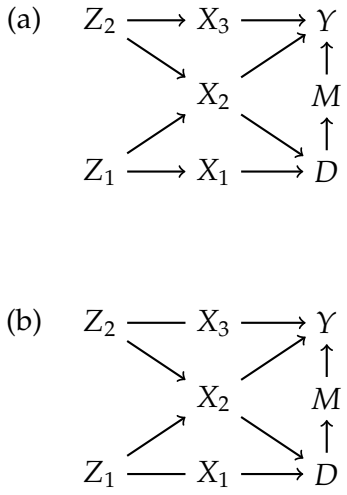


Figure 7.14: The original DAG, (a), and the equivalence class or PDAG, (b), for the Pearl's Example. The undirected edges in the PDAG mean that they can be directed in any direction as long as this does not create a cycle. Only two edges can be reoriented here.

Remark 7.5.1 (Falsifiability) The edge matrix E of a graph is *triangular* if rows of E can be rearranged to have only 1's below the diagonal. In the absence of any further restrictions, an ASEM with graph $\mathsf{G} = (V, E)$ has testable implications if E is not triangular. If E is triangular, then any law p of any arbitrary collection of random variables $(X_j)_{j \in V}$ indexed by V can be factorized as

$$\mathsf{p}(\{x\}_{j \in V}) = \prod_{j \in V} \mathsf{p}(x_j \mid pa_j).$$

With population data we have p and can check if it factorizes according to V. If matrix E is triangular, p always obeys the factorization property. This is to say that there are no exclusion restrictions in the model.

Example 7.5.1 (TSEM continued) In the TSEM example (Example 7.2.1, we have vertices $V = \{1, 2, 3\}$ identified with

Y, P, X and the "triangular" edge set

$$E = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}.$$

In the absence of other assumptions, the corresponding TSEM implies no falsifiable restrictions. The equivalence class of the DAG model for this case is generated by rearranging the rows of E in 3! ways, which is equivalent to rearranging the names (Y, P, X) for the nodes.

Example 7.5.2 (Pearl's Example) The DAG given in Figure 7.14 has vertices $V = \{1, \ldots, 8\}$ identified with Y, M, D, X_1, X_2, X_3, Z_1 ,Z_2 and the edge set

$$E = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

This edge set cannot be rearranged to have only ones below the diagonal. The DAG in this case has testable implications, and the equivalence class of the DAG model can only involve changing edges between Z_1 and X_2 and between Z_2 and X_3.

Faithfulness and Causal Discovery

Given that DAGs effectively encode conditional independence relations, it is tempting to try to infer conditional independence directly from the data. *Causal discovery* refers to methods that indeed attempt to learn conditional independence relationships from data with one application being attempting to recover causal structures. The possibility of recovering causal structures perfectly from the population data critically relies on the concept of faithfulness.

Recall that d-separation implies conditional independence, but the reverse implication

$$Y \perp\!\!\!\perp X|S \implies (Y \perp\!\!\!\perp_d X|S)_{\mathsf{G}} \tag{7.5.1}$$

is not true in general. If we restrict attention to the set of distributions p of random variables associated with graph G such that implication (7.5.1) holds, we are said to impose the *faithfulness* assumption on p.

Example 7.5.3 (Unfaithfulness) A trivial example is the DAG

$$X \to Y$$

where

$$Y := \alpha X + \epsilon_Y; \quad X := \epsilon_X;$$

with ϵ_X and ϵ_Y independent standard normal variables. Consider S to be the empty set. In this model we have that $Y \perp\!\!\!\perp X$ when $\alpha = 0$, but Y and X are not d-separated in the DAG $X \to Y$. The distribution p of (Y, X) corresponding to $\alpha = 0$ is said to be unfaithful. However, the exceptional point $\alpha = 0$ has a measure 0 on the real line, so this exception is said to be non-generic.

The observation about the simple example above generalizes: If probabilities p themselves are viewed as generated by Nature as a draw from a continuum P, where each $\mathsf{p} \in \mathsf{P}$ factorizes according to G, then the set of models where the reverse implication (7.5.1) does not hold has measure zero. This observation motivates the argument that the faithfulness assumption is a weak requirement; that is, a given p is "very unlikely" to be unfaithful.

Remark 7.5.2 (Causal Discovery) The use of the faithfulness assumption should allow us to discover the equivalence class of the true DAG from the population distribution p: We can compute all valid conditional independence relations and then discover the equivalence class of DAGs. See, for example, the PC algorithm [13] for an explicit causal discovery algorithm and the review provided in [14]. We can then apply contextual knowledge to further orient the edges of the graph.

Even though the set of unfaithful distributions has measure zero, the neighborhood of this set may not be small in high-dimensional graphs, which creates difficulty in inferring the DAG structure from an estimated version $\hat{\mathsf{p}}$.

Example 7.5.4 (Unfaithfulness Continued) In the trivial example above, suppose that we have that $\hat{\alpha} = .1$ and $\hat{\alpha} \sim N(\alpha, \sigma^2)$ where $\sigma = .1$. Then we can't be sure whether $\alpha = 0$, $\alpha = .1$,

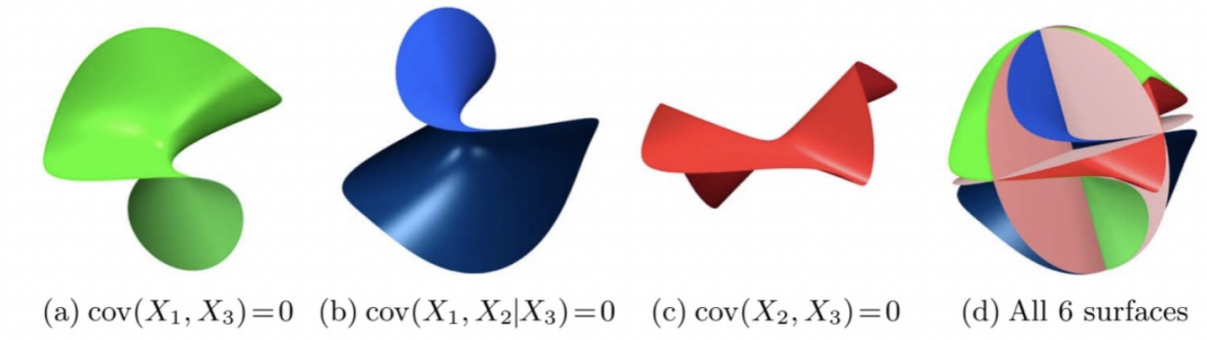


Figure 7.15: Uhler et. al [15]: A set of "unfaithful" distributions p in the simple triangular Gaussian SEM/DAG: $X_1 \to X_2, (X_1, X_2) \to X_3$. The set is parameterized in terms of the covariance of (X_1, X_2, X_3). The right panel shows the set of unfaithful distributions, and the three other panels show 3 of 6 components of the set. Each of the cases corresponds to the non-generic case which would make faithfulness fail, leading to discovery of the wrong DAG structure. While the exact setting where faithfulness would fail is non-generic, there are many distributions that are "close" to these unfaithful distributions. This observation means that, in finite samples, we are not able distinguish models that are close to the set of unfaithful distributions from unfaithful distributions and may thus also discover the wrong DAG structure and correspondingly draw incorrect causal conclusions.

or α equals any other number, though say a 95% confidence interval would have α between $-.1$ and .3. Therefore, we can't be sure whether the true model is

$$X \to Y \text{ or } X \quad Y.$$

Informally speaking, it is impossible to discover the true graph structure in this example when $\alpha \approx 0$. In econometrics jargon, this statement amounts to saying that we can't distinguish exact exclusion restrictions from "approximate" exclusion restrictions.

Thus, it is hard to distinguish exact independence from approximate independence with finite data. In high-dimensional graphs, the possibility that $\hat{p}$ lands in the "near-unfaithful" regions can be substantial, as Uhler et. al.[15]'s analysis shows.

See Uhler et al's [15] figure; reproduced in Figure 7.15.

The observations above motivate a form of sensitivity analysis – e.g., Conley et al. [16] – where one replaces exact exclusion restrictions by approximate exclusion restrictions that can't be distinguished from exact exclusion restrictions and examines the sensitivity of causal effect estimates.

Notebooks

- R: Dagitty Notebook employs the R package "dagitty" to analyze Pearl's Example (introduced in Figure 7.14) as well as simpler ones. Python: Pgmpy Notebook employs the analogue with Python package "pgmpy" and conducts

the same analysis. Both packages automatically list all conditional independence in a DAG; these are obtained by using the graphical d-separation criterion. We then go ahead and test those restrictions assuming a linear ASEM structure. The notebook also illustrates the analysis from the next chapter.

- R: Dosearch Notebook employs the R-package "dosearch" to analyze Pearl's Example (introduced in Figure 7.14). This package automatically finds identification answers to causal queries, allowing us to also answer these types of queries under different data sources, sample selection, and other deviations from the standard framework. Python: Dosearch Notebook does the same thing by loading the R "dosearch" package into Python.

Additional resources

- Dagitty.Net is an excellent online resource where you can plot and analyze causal DAG models online. It contains many interesting examples of DAGs used in empirical analysis in various fields.

- Causalfusion.Net is another excellent online resource where you can plot and analyze causal DAG models. This resource covers many different deviations from the standard framework.

Study Problems

The study problems ask learners to analyze Pearl's Example (introduced in Figure 7.14). The provided notebooks are a useful starting point for answering these questions.

Recall that Pearl's Example is structured as follows:

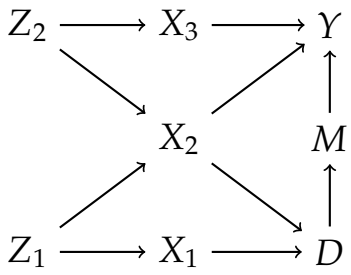


Figure 7.16: Pearl's Example

1. Consider Pearl's Example and answer the following questions. The best way to answer this question is to use computational packages (but please explain the principles the package is using).
   a) What are the testable implications of the assumptions embedded in the model? Hint: The testable implications are derived from the d-separation criterion.
   b) Assume that only variables D, Y, X_2 and M are measured, are there any testable implications?
   c) Now assume only D, Y, and X_2 are measured. Are there any testable implications?
   d) Now assume that all of the variables but X_2 (7 in total) are measured. Are there any testable restrictions?
   e) Assume that an alternative model, competing with Model 1, has the same structure, but with the $X_2 \rightarrow D$ arrow reversed. What statistical test would distinguish between the two models?
2. Work through the proof that d-separation implies conditional independence in Section 7.C. Supply the steps of the proof that were left as a homework or reading exercise.

7.A Counterfactual Distributions*

Interventions induce new counterfactual distributions for endogenous variables. We can readily compute these distributions from the definitions of interventions, as illustrated in the following for the do intervention.

Example 7.A.1 (Counterfactual Law for Do Intervention in LS-DAG (Example 7.3.1)) We can write the counterfactual distribution of $Y(d), Z, X$ in terms of the factual distribution as

$$\mathsf{p}(y, z, x : \mathrm{do}(d)) = \mathsf{p}(y|d, x)\,\mathsf{p}(z)\,\mathsf{p}(x).$$

Indeed,

$$\mathsf{p}(y, z, x : \mathrm{do}(d)) = \mathsf{p}(y|z, x : do(d))\,\mathsf{p}(z, x : do(d)),$$

by definition and Bayes' law. We also have $\mathsf{p}(y|z, x : do(d)) = \mathsf{p}(y|d, x)$ and $\mathsf{p}(z, x : do(d)) = \mathsf{p}(z, x)$ by the definition of the counterfactual ASEM, and $\mathsf{p}(z, x) = \mathsf{p}(z)\,\mathsf{p}(x)$ by indepen-

dence of Z and X.

Theorem 7.A.1 (Counterfactual Law Induced by the Do Intervention) *The induced law p_{X^*} of the counterfactual variables $X^* = (X^*_\ell)_{\ell \in V \setminus j}$ induced by $do(X_j = x_j)$ can be stated in terms of the factual law as follows:*

$$\mathsf{p}(\{x_\ell\}_{\ell \in V \setminus j} : \mathrm{do}(x_j)) := \mathsf{p}_{X^*}(\{x\}_{\ell \in V \setminus j}) = \prod_{\ell \in V \setminus j} \mathsf{p}(x_\ell \mid pa^*_\ell),$$

where $\{x\}_{\ell \in V \setminus j}$ denotes the point where the density function is evaluated, pa^*_j denotes the parental values under the new edge structure, and p denotes the factual law.

The result follows immediately from the Markov factorization property and the definition of counterfactuals under the do intervention. This characterization is interesting in its own right, because it can be used for identification and inference on the counterfactual laws directly, provided that we are willing to model the distribution of the variables. The use of Bayesian methods can be fruitful for this purpose.

These type of formulas are often called "g-formulas" and first appeared in the work [17] of James Robins in 1986 (using another "tree-based" form of causal graphs).

7.B Review of Conditional Independence

The following lemma reviews various ways in which conditional independence can be established.

Lemma 7.B.1 (Equivalent Forms of Conditional Independence) *Variables X and Y are conditionally independent given Z if and only if one of the following conditions is met:*

1. $\mathrm{p}(x \mid y, z) = \mathrm{p}(x \mid z)$ *if* $\mathrm{p}(y, z) > 0$.
2. $\mathrm{p}(x \mid y, z) = f(x, z)$ *for some function* f.
3. $\mathrm{p}(x, y \mid z) = \mathrm{p}(x \mid z)\mathrm{p}(y \mid z)$ *if* $\mathrm{p}(z) > 0$.
4. $\mathrm{p}(x, y \mid z) = f(x, z)g(y, z)$ *for some functions* f *and* g.
5. $\mathrm{p}(x, y, z) = \mathrm{p}(x \mid z)\mathrm{p}(y \mid z)\mathrm{p}(z)$ *if* $\mathrm{p}(z) > 0$.
6. $\mathrm{p}(x, y, z) = \mathrm{p}(x, z)\mathrm{p}(y, z)/\mathrm{p}(z)$ *if* $\mathrm{p}(z) > 0$.
7. $\mathrm{p}(x, y, z) = f(x, z)g(y, z)$ *for some functions* f *and* g.

As a reading exercise prove the equivalence of (1) and (2), of (1) and (7), and of any other pair.

7.C Theoretical Details of d-Separation*

Here we explain why d-separation implies conditional independence.[9]

Lemma 7.C.1 (Easy Form of d-Separation) *Let* X, Y, *and* Z *be three disjoint sets of variables in an ASEM such that their union is an ancestral set, that is, for any* $X \in \mathsf{X} \cup \mathsf{Y} \cup \mathsf{Z}$ *and* $X' < X$ *we have* $X' \in \mathsf{X} \cup \mathsf{Y} \cup \mathsf{Z}$. *If* Z *d-separates* X *and* Y, *then*

$$\mathsf{X} \perp\!\!\!\perp \mathsf{Y} \mid \mathsf{Z}.$$

Proof. Let Z_1 be the set of nodes in Z that have parents in X. And let $\mathsf{Z}_2 = \mathsf{Z} \backslash \mathsf{Z}_1$.

Because Z d-separates X and Y, we have that (see Figure 7.17):

- For any $W \in \mathsf{X} \cup \mathsf{Z}_1$, $Pa_W \subseteq \mathsf{X} \cup \mathsf{Z}$;[10]
- For any $W \in \mathsf{Y} \cup \mathsf{Z}_2$, $Pa_W \subseteq \mathsf{Y} \cup \mathsf{Z}$.[11]

Let U denote the set of variables not included in X, Y, or Z. We then obtain a factorization

$$\begin{aligned} \mathrm{p}(\mathsf{x}, \mathsf{z}, \mathsf{y}) &= \int \prod_{W \in \mathsf{U} \cup \mathsf{X} \cup \mathsf{Y} \cup \mathsf{Z}} \mathrm{p}(w \mid Pa_W = pa_W) d\mathsf{u} \\ &= \int \prod_{W \in \mathsf{U}} \mathrm{p}(w \mid Pa_W = pa_W) d\mathsf{u} \\ &\quad \times \prod_{W \in \mathsf{X} \cup \mathsf{Z}_1} \mathrm{p}(w \mid Pa_W = pa_W) \\ &\quad \times \prod_{W \in \mathsf{Z}_2 \cup \mathsf{Y}} \mathrm{p}(w \mid Pa_W = pa_W), \end{aligned}$$

where in the last equality we used the fact that u does not appear at all in the second and third factors, since $\mathsf{X} \cup \mathsf{Y} \cup \mathsf{Z}$ is ancestral. Moreover, the second factor is a function of x and z alone and the third factor is a function of y and z alone. The integral is 1 by total probability.[12] It follows that $\mathsf{X} \perp\!\!\!\perp \mathsf{Y} \mid \mathsf{Z}$.[13] □

Now we restate the main claim we'd like to demonstrate, which is that d-separation implies conditional independence.

Global Markov. Let X and Y be two variables and Z be a set of variables that does not contain X or Y. If Z d-separates X and Y, then

$$X \perp\!\!\!\perp Y \mid \mathsf{Z}$$

Proof of Theorem 7.4.1.

9: We follow the proof sketch presented in Nevin L. Zhang's lecture notes, but rely on ASEMs to simplify some arguments and supply a proof for a key claim.

10: Suppose that any such node has a parent in Y. If it were a node in X, then we get a violation of d-separation. If it were a node in Z_1, then we have that Z_1 has one parent in X and one parent in Y and therefore it is a collider that was included in Z, violating d-separation.

11: Suppose that any such node has a parent in X. By the definition of Z_1 it has to be a node in Y. But then we have that a node in Y has a parent in X, violating d-separation.

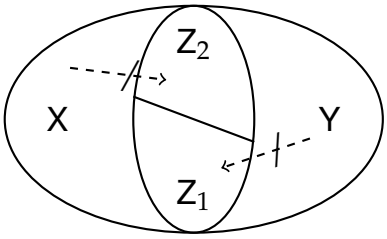


Figure 7.17: Pictorial representation of key argument in Lemma 7.C.1.

12: Prove this as a reading exercise by integrating over the variables in U in reverse order with respect to the DAG ordering.

13: Prove this as a reading exercise, i.e., prove bullet (7) of Lemma 7.B.1.

Let X be the set of all ancestors of $\{X, Y\} \cup \mathsf{Z}$ that are *not* d-separated from X by Z. Let Y be the set of all ancestors of $\{X, Y\} \cup \mathsf{Z}$ that are neither in X nor in Z.

Key Claim: The set Z d-separates the sets X and Y.

The claim follows from the careful use of the definition of d-separation, and is proven below.

Given the key claim, Lemma 7.C.1 implies that $\mathsf{X} \perp\!\!\!\perp \mathsf{Y} \mid \mathsf{Z}$, since $\mathsf{X} \cup \mathsf{Y} \cup \mathsf{Z}$ is ancestral by its exhaustive construction. This implies that there must exist functions $f(\mathsf{x}, \mathsf{z})$ and $g(\mathsf{z}, \mathsf{y})$ such that

$$\mathrm{p}(\mathsf{x}, \mathsf{z}, \mathsf{y}) = f(\mathsf{x}, \mathsf{z}) g(\mathsf{z}, \mathsf{y}).$$

Since X is in X and Y in Y, the conclusion is reached.[14] □

14: Prove this explicitly, as a reading exercise, by integrating over all variables in $\mathsf{X} \backslash \{X\}$ and $\mathsf{Y} \backslash \{Y\}$ and invoking Lemma 7.B.1.

Proof of the Key Claim. Suppose that Z does not d-separate the sets X and Y and that there exists a node $X' \in \mathsf{X}$ which is not d-separated from some node $Y' \in \mathsf{Y}$. Thus, there is an open path X - - X',[15] and an open path X' - - Y'. Consider the concatenation of these two paths. If X' is not a collider on this concatenated path, then the path X - - X' - - Y' is also open, and therefore X is not d-separated from Y', which is in contradiction with the definition of X and Y. Thus X' has to be a collider on this concatenated path. Moreover, note that since we are only restricting our analysis to the ancestral set $An_{\{X,Y\} \cup \mathsf{Z}}$, we have that X' must be an ancestor of either Z or Y or X:

15: In this proof, we denote with U - - V a path from a node U to a node V and with $U \dashrightarrow V$ a directed path from U to V.

If X' is an ancestor of some node in Z then the path X - - X' - - Y' is again open, leading to a contradiction with the definition of X and Y.

If X' is an ancestor of Y, then there is a directed path $X' \dashrightarrow Y$. If that path is open, then there is an open path X - - $X' \dashrightarrow Y$, violating the fact that Z was d-separating X from Y. For the path to be closed, it must be that some node $Z \in \mathsf{Z}$ is on the path. However, in this case X' is an ancestor of a node in Z, which has already been excluded.

Finally, if X' is an ancestor of X, then there exists a directed path $X' \dashrightarrow X$. This path also has to be open, as if a node in Z existed on that path, then X' would be an ancestor of a node in Z, which has been excluded. However, in this case, we have an open path Y' - - $X' \dashrightarrow X$, from Y' to X, which violates the definition of X and Y. □

Bibliography

[1] Judea Pearl and Dana Mackenzie. *The Book of Why*. Penguin Books, 2019 (cited on page 166).

[2] Judea Pearl. 'Causal diagrams for empirical research'. In: *Biometrika* 82.4 (1995), pp. 669–688 (cited on pages 167, 183).

[3] Trygve Haavelmo. 'The probability approach in econometrics'. In: *Econometrica* 12 (1944), pp. iii–vi+1–115 (cited on pages 167, 169).

[4] James Heckman and Rodrigo Pinto. 'Causal analysis after Haavelmo'. In: *Econometric Theory* 31.1 (2015 (NBER 2013)), pp. 115–151 (cited on pages 167, 172).

[5] Thomas S. Richardson and James M. Robins. *Single world intervention graphs (SWIGs): A unification of the counterfactual and graphical approaches to causality*. Working Paper No. 128, Center for the Statistics and the Social Sciences, University of Washington. 2013. URL: https://csss.uw.edu/files/working-papers/2013/wp128.pdf (cited on pages 167, 172).

[6] Alfred Marshall. *Principles of Economics: Unabridged Eighth Edition*. Cosimo, Inc., 2009 (cited on page 170).

[7] Philip G. Wright. *The Tariff on Animal and Vegetable Oils*. New York: The Macmillan company, 1928 (cited on page 172).

[8] Frederick Eberhardt and Richard Scheines. 'Interventions and causal inference'. In: *Philosophy of Science* 74.5 (2007), pp. 981–995 (cited on page 172).

[9] Juan Correa and Elias Bareinboim. 'General Transportability of Soft Interventions: Completeness Results'. In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., 2020, pp. 10902–10912 (cited on page 172).

[10] Judea Pearl. *Causality*. Cambridge University Press, 2009 (cited on pages 173, 178, 179, 182).

[11] Thomas Verma and Judea Pearl. *Influence diagrams and d-separation*. Tech. rep. Cognitive Systems Laboratory, Computer Science Department, UCLA, 1988 (cited on page 180).

[12] Rajen D. Shah and Jonas Peters. 'The hardness of conditional independence testing and the generalised covariance measure'. In: *Annals of Statistics* 48.3 (2020), pp. 1514–1538 (cited on pages 181, 182).

[13] Peter Spirtes, Clark N. Glymour, Richard Scheines, and David Heckerman. *Causation, Prediction, and Search*. MIT Press, 2000 (cited on page 185).

[14] Clark Glymour, Kun Zhang, and Peter Spirtes. 'Review of causal discovery methods based on graphical models'. In: *Frontiers in Genetics* 10 (2019), p. 524 (cited on page 185).

[15] Caroline Uhler, Garvesh Raskutti, Peter Bühlmann, and Bin Yu. 'Geometry of the faithfulness assumption in causal inference'. In: *Annals of Statistics* 41.2 (2013), pp. 436–463 (cited on page 186).

[16] Timothy G. Conley, Christian B. Hansen, and Peter E. Rossi. 'Plausibly exogenous'. In: *Review of Economics and Statistics* 94.1 (2012), pp. 260–272 (cited on page 186).

[17] James Robins. 'A new approach to causal inference in mortality studies with a sustained exposure period—application to control of the healthy worker survivor effect'. In: *Mathematical Modelling* 7.9-12 (1986), pp. 1393–1512 (cited on page 189).

8 Valid Adjustment Sets from DAGs

"if 'good' is taken to mean 'best' fit, it is tempting to include anything in x that helps predict [treatment]"

– Jeffrey Wooldridge [1].

DAGs give us an intuitive approach to take domain knowledge and turn it into an identification strategy. In this section, we focus on identification by conditioning and discuss graphical criteria that lead to the construction of valid adjustment sets for the identification of average causal effects via regression adjustment. We also discuss how graphical criteria can help us differentiate between "good" and "bad" controls.

8.1 Valid Adjustment Sets

Consider any variable D of an ASEM as a treatment of interest and any of its descendants Y as an outcome of interest. An adjustment set S is said to be valid for identification of the causal effect of D on Y if the conditional exogeneity/ignorability condition holds

$$Y(d) \perp\!\!\!\perp D \mid S.$$

In what follows, we present an exhaustive (complete) approach for finding valid adjustment sets by using SWIGs.

We write down the counterfactual SWIG induced by the

$$\text{fix}(D = d)$$

intervention, which operates on all structural equations defining the descendants of D by setting $D = d$ in these equations.

Then, if we have that the potential outcome $Y(d)$ is d-separated from the (policy) variable D by a set of variables S, conditional exogeneity/ignorability holds:

$$Y(d) \perp\!\!\!\perp D \mid S.$$

Given that conditional exogeneity/ignorability holds, we can identify counterfactual expectations,

$$\text{E}[Y|S = s : \text{do}(d)] := \text{E}[Y(d)|S = s],$$

from expectations of observed variables,

$$\text{E}[Y|S = s, D = d],$$

provided that the positivity condition $p(s, d) > 0$ holds. The agreement between counterfactual and conditional expectations follows because

$$\text{E}[Y(d)|S = s] = \text{E}[Y(d)|D = d, S = s]$$

by exogeneity and

$$\text{E}[Y(d)|D = d, S = s] = \text{E}[Y|D = d, S = s]$$

by consistency.

We can recover unconditional counterfactual means by integration:

$$\mathrm{E}[Y : \mathrm{do}(d)] := \mathrm{E}[Y(d)] = \mathrm{E}[\mathrm{E}[Y|S, D = d]],$$

provided that the positivity condition $p(s, d) > 0$ for each s in the support of $S \mid D = d$ holds.

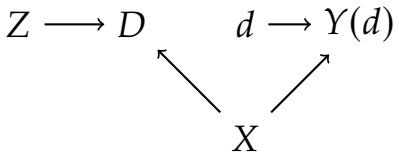


Figure 8.1: CF LS-DAG induced by $\mathrm{fix}(D = d)$ intervention.

Example 8.1.1 (Identification in LS-DAG.) In the SWIG graph in Figure 8.1 corresponding to the LS-DAG model from Example 7.3.1, we see that either $S = X$ or $S = (X, Z)$ d-separates $Y(d)$ from D. Therefore, either choice of S provides a valid adjustment set for identifying counterfactual predictions. Here conditioning on Z is not necessary, though we maintain robustness with respect to the presence of a directed edge from Z to Y by including Z in the conditioning set.

We can identify the entire conditional distribution

$$\mathrm{P}(Y(d) \leq t \mid S = s)$$

from the conditional distribution

$$\mathrm{P}(Y \leq t \mid D = d, S = s).$$

We achieve identification of the distribution by replacing Y with $1(Y < t)$ in all previous statements and applying the same arguments for each $t \in \mathbb{R}$. The unconditional distribution of potential outcomes is retrieved by integrating out S:

$$\mathrm{P}(Y(d) \leq t) := \mathrm{E}[\mathrm{P}(Y(d) \leq t \mid S)].$$

The following theorem, essentially due to [2], records the discussion formally.

Theorem 8.1.1 (A Complete Criterion for Identification by Conditioning) *Consider any ASEM with DAG* G*. Let us re-label a policy node* X_j *as* D*, and let* Y*, an outcome of interest, be any other descendant of* D*.*

Consider a SWIG DAG $\tilde{\mathsf{G}}(d)$ *which is induced by the* $\mathrm{fix}(D = d)$ *intervention. Consider any other subset of nodes* S *that appears in both* G *and* $\tilde{\mathsf{G}}(d)$*, such that*

$$Y(d) \text{ is d-separated from } D \text{ by } S \text{ in } \tilde{\mathsf{G}}(d).$$

- *Then the following conditional exogeneity/ignorability holds:*

$$Y(d) \perp\!\!\!\perp D \mid S.$$

- *Then*

$$\mathrm{E}[Y(d)|S = s] = \mathrm{E}[Y \mid D = d, S = s]$$

holds for all s such that $p(d, s) > 0$.

Example 8.1.2 (Pearl's Example) Consider the DAG in Figure 8.2, which we introduced as Pearl's Example in Figure 7.14, and the corresponding ASEM, which we don't write out. Here, we are interested in the causal effect $D \to Y$, that is, the effect $d \mapsto Y(d)$. The corresponding SWIG-intervention DAG is shown in Figure 8.3. In this DAG, valid adjustment sets S include

$$\{X_1, X_2\}, \{X_2, X_3\}, \{X_2, Z_2\}, \{X_2, Z_1\},$$

because each d-separates $Y(d)$ and D by blocking all open paths. Conditioning on just X_2 won't work, because it blocks the inner backdoor paths from $Y(d)$ to D, but opens the outer path on which X_2 is a collider. To close this opened path, it suffices to also condition on one of X_1, X_3, Z_1 or Z_2.

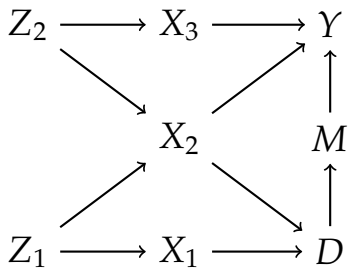


Figure 8.2: A DAG in Pearl's Example

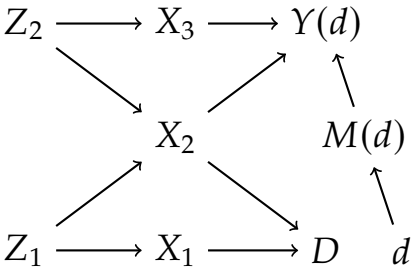


Figure 8.3: The DAG induced by the Fix/SWIG intervention fix($D = d$) in Pearl's Example.

8.2 Useful Adjustment Strategies

Theorem 8.1.1 provides an exhaustive criterion for finding valid adjustment sets. We now discuss other frequently used strategies for obtaining valid adjustment sets which are strictly less general. Some of these strategies are quite helpful because they are either very simple to apply or can also be used under partial knowledge of the DAG.[1]

We consider three approaches that allow us to identify the causal effect of D on Y:

- Conditioning on one of **all parents** of Y (that are not descendants of D), **all parents** of D, or **all parents** of both D and Y is sufficient. This approach provides a valid adjustment set irrespective of the remaining structure of the problem.
- Conditioning using the **backdoor criterion** enables us to find all minimal adjustment sets.

1: See [3] for a more detailed discussion of identification by conditioning under limited knowledge of DAGs.

- Conditioning on **all common causes** of D and Y is also sufficient.

Conditioning on Parents

A very simple strategy is conditioning on one of the parents of D, the parents of Y, or the parents of both D and Y.

Example 8.2.1 (Pearl's Example Continued) One simple principle is that conditioning on parents of D, namely X_1 and X_2, is sufficient. Alternatively, conditioning on all parents of Y that are non-descendants of D, namely X_2 and X_3, is also sufficient. We should not condition on M, because it is a descendant of D.

Corollary 8.2.1 (Adjustment for Parents) *Consider any ASEM. Re-label a policy node X_j as D, and let Y, an outcome of interest, be any other descendant of D.*

- *Let Z be all parents of D, and let A be any other set of nodes that are not descendants of D. Then $S = (A, Z)$ is a valid adjustment set.*
- *Let Z be the set of all parents of Y that are non-descendants of D and let A be any other set that are not descendants of D. Then $S = (A, Z)$ is a valid adjustment set.*

Note that A is allowed to be an empty set. Also note that, in the second case, the additional adjustment set A is redundant, since $\mathsf{p}(y \mid a, z, d) = \mathsf{p}(y \mid z, d)$ in this case.

Adjusting for parents is a very useful strategy, because it only requires knowledge of parents in a DAG without precise knowledge of the remaining graph structure. Conditioning on parents is also behind the propensity score strategies used in many experimental or quasi-experimental empirical analyses. If the propensity score is known, it can be used as a parent of D itself. Finally, conditioning on parents of Y is most useful for attaining maximal statistical efficiency, but may be less robust than conditioning on *both* sets of parents under unforeseen deviations from the given graph structure. See [3] for further detailed discussion of robustness of adjusting for both sets of parents.

Conditioning by Backdoor Blocking

Pearl [4] developed the following powerful criterion.

Corollary 8.2.2 (Backdoor Criterion) *Consider any ASEM. Relabel a policy node X_j as D, and let Y, an outcome of interest, be any other descendant of D. The adjustment set S is valid if the backdoor criterion is satisfied: No element of S is a descendant of D, and all backdoor paths from Y to D are blocked by S.*

In other words, if a collection of random variables S satisfies the backdoor criterion with respect to (D, Y), then conditioning on S identifies the causal effect of D on Y. The basic idea is that if we block the backdoor path, we remove all channels of non-causal association between D and Y.

Example 8.2.2 (Pearl's Example Again, using the Backdoor Criterion) The graph in Figure 8.2 has two backdoor paths from D to Y: the inner path $D \leftarrow X_2 \rightarrow Y$ and the outer path $D \leftarrow X_1 \leftarrow Z_1 \rightarrow X_2 \leftarrow Z_2 \rightarrow X_3 \rightarrow Y$. Conditioning on just X_2 does not allow us to identify the causal effect of D on Y because X_2 blocks the inner backdoor path from Y to D but opens the outer path on which X_2 is a collider. To close this opened path, it suffices to condition on X_1, X_3, Z_1, or Z_2. For example, conditioning sets $S_1 = \{X_1, X_2\}$ or $S_2 = \{X_2, X_3\}$ are valid. Figuring out other valid conditioning sets is left as an exercise. (You can find the answers using the notebook R: Dagitty Notebook or Python: Pgmpy Notebook.) Conditioning on M is obviously not valid – it is a descendant of D, an intermediate outcome.

Application of the backdoor criterion can produce all minimal adjustment sets. Relative to the complete strategy formalized in Theorem 8.1.1, we exclude the descendants of D from valid adjustment sets when we focus on backdoor paths. A simple example of a graph where the backdoor criterion does not find all valid adjustment sets is

$$Z \leftarrow D \rightarrow Y.$$

Here conditioning on Z is valid but unnecessary. Conditioning on Z may thus decrease statistical efficiency.[2]

2: We may think that conditioning on Z here could be useful to uncover heterogeneity. However, $Y(d)$ does not depend on Z, so conditioning on Z is not useful for describing heterogeneity and can decrease the efficiency of the estimator.

Conditioning on All Common Causes of D and Y

Another simple and widely used adjustment strategy is conditioning on all common causes of the outcome variable of interest and the treatment variable.

Example 8.2.3 (Pearl's Example Again, using the All Common Causes Criterion) The set of common causes of D and Y is $\{Z_1, Z_2, X_2\}$. This set is a valid adjustment set that differs from the sets found using the parental strategy. We can push the All Common Causes criterion further. For example, we can omit Z_1 and Z_2 from the DAG, and we can create a new node $X = (X_1, X_2, X_3)$ producing the DAG shown in Figure 8.4. This DAG corresponds to a valid ASEM model where X now represents all common causes of D and Y, making it a sufficient adjustment set. This set is bigger than some of the sets found by the previous criteria. It is also tempting to see if the "root common" causes Z_1 and Z_2 in the original DAG, Figure 8.2, form a valid adjustment set – and they actually do not (why?).

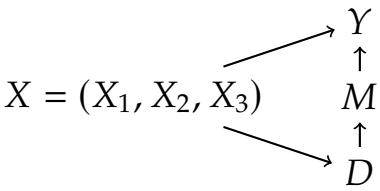


Figure 8.4: Reduced DAG for Pearl's Example

Let $\underline{An}_X$ denote the set of strict ancestors of node X, where strict means that X is excluded. That is,

$$\underline{An}_X = An_X \setminus X.$$

Corollary 8.2.3 (Adjustment for All Common Causes) *Consider any ASEM. Re-label a policy node X_j as D, and let Y, an outcome of interest, be any other descendant of D. Let S be the intersection of the strict ancestors of D and Y, called the common causes:*

$$S = (\underline{An}_D \cap \underline{An}_Y).$$

Then S is a valid adjustment set. Furthermore, the set of variables S' that completely mediates the effects of S on Y and D also constitutes a valid adjustment set.

The strategy above is commonly used in empirical work. However, [3] recommend adjusting for the union S of causes of Y or D (excluding descendants of D) in practice as they formally quantify this strategy as the maximally robust strategy under perturbations of a specified DAG structure that preserves S. This strategy is useful when we don't know the parents of Y or D, but only know that S are their ancestors.

Corollary 8.2.4 (Adjustment for the Union of Causes) *Consider any ASEM. Re-label a policy node X_j as D, and let Y, an outcome of interest, be any other descendant of D. Let S be the union of the ancestors of D and Y that excludes descendants of D other than Y:*

$$S = \underline{An}_D \cup \underline{An}_Y \setminus Ds_D.$$

Then S is a valid adjustment set.

Example 8.2.4 (Pearl's Example Continued) Application of the Union of Causes criterion gives $\{Z_1, Z_2, X_1, X_2, X_3\}$ as a valid adjustment set.

8.3 Examples of Good and Bad Controls

We now present a series of simple example DAGs that might arise in empirical research. Within these examples, we discuss what would be good and bad variables to adjust for in each case (aka good and bad controls), when one is interested in estimating the average treatment effect of a treatment D on an outcome Y.[3] Similar to the collider bias examples we presented in Section 6.3, we will see how adjusting for some of the observed variables can introduce bias and lead to estimating a parameter that is far from the causal effect of interest. In each case, we will denote the candidate control of interest with Z and will denote unobserved variables with U. We depict unobserved variables with a dashed circle in the figures.

3: The content in this section draws heavily from the excellent research paper of Cinelli, Forney and Pearl [5].

We start by analyzing a group of potential control variables that in most empirical applications would correspond to *pre-treatment variables*, i.e. variables whose value was determined prior to the treatment assignment. It is common empirical practice to adjust for as many pre-treatment variables as available in an attempt to ensure that conditional ignorability holds. However, we will see that bias can be introduced by controlling even for pre-treatment variables if one is not careful. Rather than always control for all pre-treatment variables, a better approach is to adjust only for pre-treatment variables that are ancestors of either the treatment, the outcome, or both. If one is willing to believe that identification by conditioning is feasible, then following this approach is a safe strategy.

We then consider the use of *post-treatment variables*, i.e. variables that correspond to quantities whose value is determined after the treatment assignment. We will see that in this case there

are relatively few good control cases. In some cases, controlling for post-treatment variables might not hurt and may even improve precision (reduce variance). However, such settings seem unlikely to be common in empirical practice. Hence, as a high-level rule, controlling for post-treatment variables should be avoided when one is interested in estimating causal effects.

Finally, we provide a separate discussion of post-treatment but *pre-outcome variables*, i.e. variables whose value is determined prior to the determination of the value of the outcome of interest. Pre-outcome variables should be included if one is interested in estimating direct effects of the treatment on the outcome while excluding indirect effects. This type of direct effect is referred to as a *controlled direct effect* to distinguish it from other forms of direct effects appearing in mediation analysis. We will see again that one should be careful that the mediation variables that one conditions on are not themselves confounded through unobserved factors even in this case.

Pre-Treatment Variables or Proxies of Pre-Treatment Variables

Observed common causes or proxies of common causes. A common example of a good control that we have discussed so far is an observed common cause, Z, of D and Y (Figure 8.5a). Even if the common cause is unobserved, it suffices that we have a proxy control variable that controls all the information flow to either the treatment (complete treatment proxy; Figure 8.5b) or to the outcome (complete outcome proxy; Figure 8.5c). Controlling for such a proxy also blocks the backdoor path $D \leftarrow U \rightarrow Y$. Of course, the proxy blocking the backdoor path only holds if the proxy variable captures *all* the information flow from the unobserved confounder. If, for instance, there are also direct paths from the unobserved variable to the treatment (in the case of a treatment proxy), then controlling for a proxy does not remove confounding bias. In this case, we will see that one can follow more advanced approaches related to proxy controls under additional structure in Chapter 12.

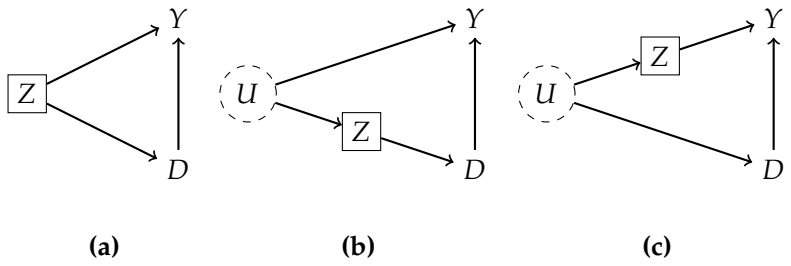


Figure 8.5: Good controls: **(a)** observed common cause, **(b)** complete treatment proxy control of unobserved common cause, **(c)** complete outcome proxy control of unobserved common cause.

Example 8.3.1 (Effect of Multivitamin Consumption on Birth Defects [6]) Suppose we want to estimate the effect of prenatal multivitamin consumption D on birth defects Y. One factor that can potentially influence a mother's decision on multivitamin consumption is prior history of birth defects in the family (Z); see e.g. [7]. Such prior history is possibly due to unobserved genetic factors U that also have a direct effect on the risk of malformation Y; see e.g. [8]. In this case, family medical history Z provides a complete treatment proxy of the unobserved confounder (as in Figure 8.5b) as long as the behavior of a mother is solely driven by the family medical history. Controlling for medical history would thus remove the confounding bias in this scenario.

Confounded mediators with observed common cause or proxies of unobserved common cause. It is important to note that confounding occurs even when there exists a common cause Z of the treatment D and some mediator M in a path from D to Y (Figure 8.6a). In such cases, if we don't condition on the common cause of D and M, there is an open backdoor path $D \leftarrow Z \rightarrow M \rightarrow Y$. In such cases, Z is a good control as it blocks this backdoor path. Similarly, if a common cause U of D and M is unobserved, but some complete treatment proxy control Z (Figure 8.6b) or some complete outcome proxy control Z (Figure 8.6c) is observed, then it suffices to adjust for this proxy Z.

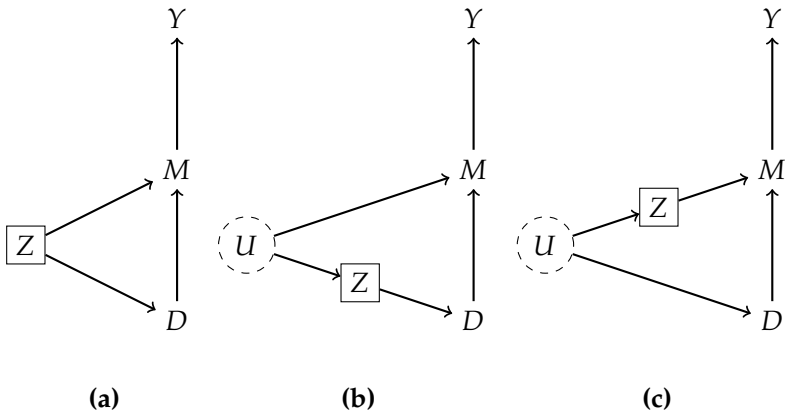


Figure 8.6: Good controls: **(a)** confounded mediator with observed common cause, **(b)** confounded mediator, with observed complete treatment proxy control of unobserved common cause, **(c)** confounded mediator with observed complete outcome proxy control of unobserved common cause.

Causes of only treatment or only outcome. As stated in Corollary 8.2.4, a conservative empirical practice is to include the union of parents of D and Y in the adjustment set. Including variables that are parents of the outcome (Figure 8.7a) can lead to reduced variance during estimation as explained in Chapter 2 where we discuss including pre-treatment covariates in RCTs. Including variables Z that affect the treatment D but have no causal path to the outcome (Figure 8.7b) is potentially more controversial. Including these variables does not introduce bias. However, their inclusion can be detrimental for precision, as such variables can potentially explain away all of the useful variation in the treatment, leaving little variation for the identification of causal effects.

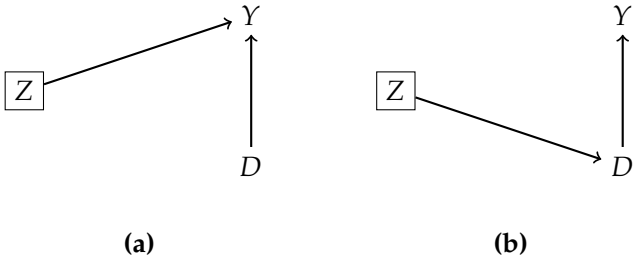


Figure 8.7: Neutral controls: **(a)** Outcome-only cause. Can improve precision; decrease variance. **(b)** Treatment-only cause. Can decrease precision; introduce variance.

Even more importantly, when there are unobserved common causes of D and Y as illustrated in Figure 8.8, adjusting for a treatment-only cause, Z, can exacerbate the bias stemming from unobserved confounding. Essentially, controlling for Z removes exogenous variation in the treatment D that is useful for identifying the causal effect but leaves the confounded variation - as Z is not related directly to the unobserved confounder U. As such, the resulting estimated effect may be essentially driven by the unobserved confounder and thus be heavily biased. For this reason, one should avoid controlling for variables that are

known to have no causal path to the outcome that does not pass through the treatment. As we will see in Chapter 12, such variables are actually what are referred to as *instruments*. These variables can be thought as useful natural experiments that can be leveraged for causal identification even in the presence of unobserved confounding. However, we will need to use alternative identification arguments and estimation strategies to make use of instruments. We introduced these *instrumental variable* approaches in Chapter 12 and Chapter 13. Importantly, instruments *should not* be used in an identification by adjustment strategy.

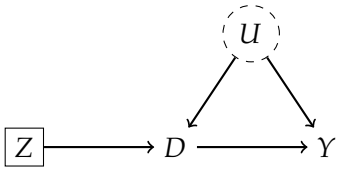


Figure 8.8: Bad control. Bias amplification by adjusting for an *instrument*. Treatment-only cause (*instrument*) that can amplify unobserved confounding bias.

M-bias The DAG in Figure 8.9, typically referred to in the literature as the M structure, is the source of much debate; see e.g. [9, 10]. If such cases were impossible, the high-level strategy of controlling for all pre-treatment variables when attempting to identify causal effects by conditioning would be an unambiguously safe empirical route resulting in no harm other than potentially increasing variance by including an *instrument*. However, this structure shows that there exist settings where adjusting for a pre-treatment covariate Z can lead to a wrong causal effect, while not adjusting for Z would have yielded the correct causal effect. A better high-level strategy is the one highlighted in the prior sections: If we are willing to assume that identification by conditioning is possible, then we should adjust only for pre-treatment variables that are either an ancestor of the treatment, of the outcome, or of both treatment and outcome.

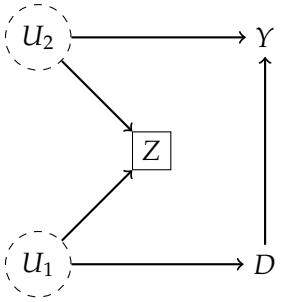


Figure 8.9: Bad control. M-Bias. Pre-treatment variable that introduces Heckman selection bias between two uncorrelated unobserved causes.

More concretely, in the M structure graph (Figure 8.9), D and Y are driven by two independent unobserved causal factors U_1, U_2. The variable Z is a common outcome of these two unobserved causal factors. When conditioning on Z, we introduce collider bias between U_1, U_2, making them correlated factors. Conditioning on Z can thus lead to a causal effect estimate that is solely driven by this spurious correlation between U_1 and U_2, introduced by the collider bias. In graphical terms, adjusting for Z closes the path $D \leftarrow U_1 \rightarrow Z \leftarrow U_2 \rightarrow Y(d)$ in the SWIG DAG $\tilde{\mathsf{G}}(d)$ produced by the fix$(D = d)$ operation. However,

there is no open path connecting D to $Y(d)$ when we do not condition on Z. Hence, the effect identified by not adjusting for any variable is the correct causal effect within this example structure.

Example 8.3.2 (Homophily bias in estimating peer effects) A classical example where M-bias arises in empirical work in social sciences is in the estimation of peer effects on social networks [11, 12]. As a concrete example, suppose that we want to understand the spread of civic engagement among friends. Suppose that we look at data that consist of friendship pairs and let D be the level of civic engagement level of one friend at time t and Y the level of civic engagement of the other friend at time $t + 1$. Note that when we are estimating the correlation of these two variables, we are implicitly conditioning on the friendship variable Z, since we only have data from friendship pairs. Due to homophily, friendship could be driven by the unobserved intrinsic characteristics of each of the two individuals (U_1 and U_2 in Figure 8.9). It is reasonable to assume that these characteristics are independent as they are determined well before any friendship is formed. Moreover, these qualitative characteristics (e.g. levels of altruism) could very well have a direct effect on each individual's civic engagement. Thus, the estimation of peer effects can be heavily biased due to exactly M-bias.

Homophily refers to the tendency of associate with similar individuals - i.e. similar people tend to become friends.

Finally, note that the M-bias argument is very sensitive to the exact independence of the unobserved factors U_1, U_2. In most empirical applications, we expect these unobserved factors that drive the treatment and outcome of interest to be correlated with each other as in Figure 8.10a. In this case, note that even if we don't adjust for Z, the calculated effect is biased due to the backdoor path $D \leftarrow U_1 \rightarrow U_2 \rightarrow Y$. Thus, neither adjusting nor not adjusting for Z gives the correct answer.

Moreover, it is not clear whether adjusting for Z increases or decreases the correlation between U_1 and U_2 and hence exacerbates or ameliorates the confounding bias. Similarly, if Z itself has a direct effect on the outcome (as in Figure 8.10b), on the treatment, or on both (as in Figure 8.10c), then not adjusting for Z opens the backdoor paths $D \leftarrow U_1 \rightarrow Z \rightarrow Y$ and $D \leftarrow Z \rightarrow Y$, correspondingly. Hence, it is not clear that removing the bias induced by these open backdoor paths, by adjusting for Z, is more beneficial than the extra M-bias incurred by closing the path $D \leftarrow U_1 \rightarrow Z \leftarrow U_2 \rightarrow Y$. Work of [9, 13] argues that M-bias in many realistic data generating processes is of lower order than confounding bias and therefore argues

that one should err on the side of adjusting for pre-treatment covariates even in the potential presence of M-bias. [10] provides a counterpoint, arguing that the strength of the different biases will differ in general and thus careful consideration of the strength of each of the causal paths at play should be done on a case-by-case basis.

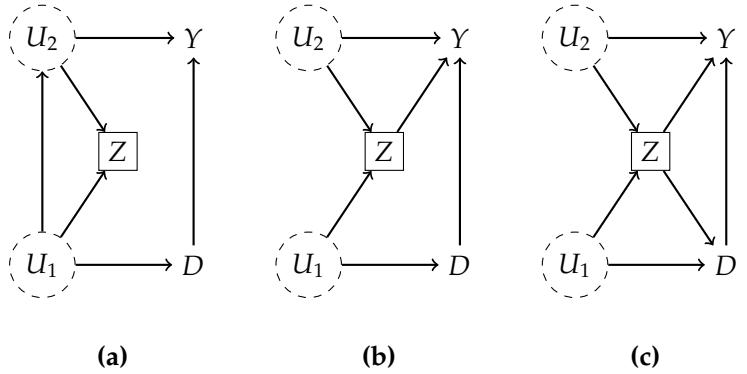


Figure 8.10: No perfect control solutions: **(a)** M-bias with correlated unobserved factors. **(b)** M-Bias with confounding. Pre-treatment variable that introduces Heckman selection between two uncorrelated unobserved causes and is a confounder itself. **(c)** Butterfly Bias. M-bias with direct confounding.

Post-Treatment Variables

Now we turn to adjustment for post-treatment variables. The general message of this section is that explicitly adjusting for post-treatment variables is almost always a bad idea. Importantly, the general message implies that researchers should be careful to avoid implicitly adjusting for post-treatment variables through the way they have structured their observational analysis, data collection, and variable definitions – see e.g. [6] for examples from epidemiology. For instance, when estimating the effect of education on wages using data on *employed* individuals, we are implicitly conditioning on "employment" which is a post-treatment variable and can lead to selection bias.

Mediation. A common way a post-treatment variable can lead to bias in identifying the full causal effect of D on Y is if it lies on a causal path from the treatment to the outcome (Figure 8.11a). In this case, the causal influence that flows through that path is blocked and we are only measuring a partial effect. It is important to note, that the causal influence of such a path can be partially blocked even if one conditions on a descendant of the mediator (Figure 8.11b).

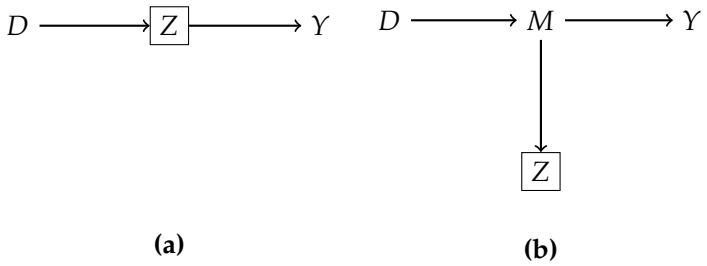


Figure 8.11: Bad controls for learning the full direct effect of D on Y: **(a)** over-control bias, by controlling for a mediator. **(b)** over-control bias, by controlling for an outcome caused by a mediator.

Interestingly, controlling for an ancestor of a mediator (Figure 8.12) does not impede us from learning the full direct effect of D on Y. In this case, the flow through the causal path $D \to M \to Y$ is not blocked by Z. For example, d-separation can be easily checked in the SWIG $\tilde{\mathsf{G}}(d)$ produced by fix$(D = d)$.

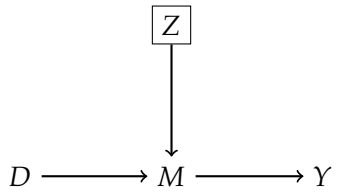


Figure 8.12: Neutral control. Cause of a mediator. Can potentially improve precision.

When we are controlling for a post-treatment variable that mediates the effect of the treatment as in Figure 8.11a, we are only capturing direct effects from the treatment to the outcome that do not work through this mediator. This type of direct effect after controlling for mediators is typically referred to as a *controlled direct effect*. Identifying the controlled direct effect is many times a relevant empirical question, in which case controlling for Z is not problematic. However, even when we are interested in the controlled direct effect, we should pay attention to cases where the mediators are themselves confounded through unobserved factors as illustrated in Figure 8.13. In such settings, by controlling for the mediator, we are opening a collider path $D \to Z \leftarrow U \to Y$ which can lead to severe bias, such as calculating non-zero direct effects even when they are zero.

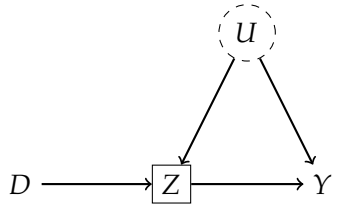
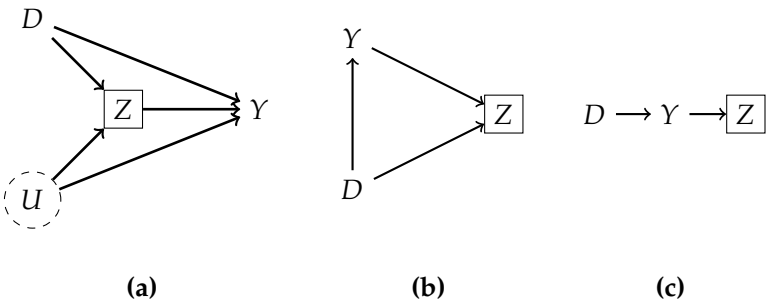


Figure 8.13: Bad control even for the *controlled direct effect*. Confounded mediator bias.

Heckman selection bias Another common way that post-treatment variables can lead to bias is due to collider bias or Heckman selection, as described in Section 6.3. In this case, conditioning on the post-treatment variable introduces spurious correlations between the treatment variable and some other variable which opens new paths of non-causal influence from the treatment to the outcome. For instance, Figure 8.14a corresponds to the low birthweight paradox we presented in Example 6.3.2. Similarly, Figure 8.14b corresponds to the Hollywood Example Example, Example 6.3.1. Finally, Figure 8.14c arises when we are controlling for an outcome of the outcome as might be produced by recall bias in a case-control study.

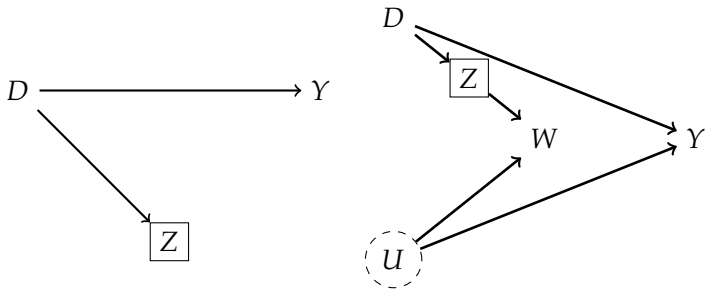


(a) **(b)** **(c)**

Figure 8.14: Bad controls: **(a)** collider stratification bias (e.g. low birth-weight "paradox" example), **(b)** collider stratification bias, **(c)** controlling for an outcome of the outcome of interest.

Example 8.3.3 (The Industrial Growth Puzzle [14]) In a study conducted during the nineteenth century in the US and Britain, it was found that despite nutrition quality D having improved, the height of men Y decreased. One possible explanation of the results of this study is that the subjects of the study were people who were enlisted in the army or in prison. Both of these variables, enlisted in the army and being in prison, are plausibly determined *after* the outcome variable of height is realized. It might, for example, be that taller men had more civilian opportunities growing up and did not end up enlisting in the army. In this case, looking at a sample of enlistees is implicitly controlling for an outcome of the outcome of interest which could lead to a biased estimate of the effect of nutrition on height.

There are of course some edge cases where controlling for a post-treatment variable Z does not lead to selection bias – e.g. Figure 8.15a and Figure 8.15b. In each of these two cases, the post-treatment variable is not a collider on a path from D to Y. However, it is not clear that adjusting for Z improves the analysis in any respect even in these cases, and adjusting for Z could potentially hurt precision.



(a) **(b)**

Figure 8.15: Neutral controls: **(a)** outcome of the treatment that is unrelated to the outcome of interest, **(b)** outcome of the treatment that does not introduce Heckman selection.

Notes

Any empirical study that tries to learn the causal effect of D on Y by conditioning on S must have a thought process that justifies this approach. The DAG/ASEM framework is a rigorous representation of such a thought process which enables explicit incorporation of domain knowledge, automatic checking of identifiability, and automatic deduction of testable restrictions. Graphs also provide an effective way of visualizing and communicating models.

Notebooks

- R: Dagitty Notebook employs the R package "dagitty" to analyze some simple DAGs as well as Pearl's Example. This package automatically finds adjustment sets and also lists testable restrictions in a DAG. Python: Pgmpy Notebook employs the analogue with Python package "pgmpy" and conducts the same analysis.

Study Problems

The study problems ask learners to continue the analysis of Pearl's Example DAG that we started in the Study Problems to Chapter 7. The provided notebooks are a useful starting point. Recall that Pearl's Example is structured as follows:

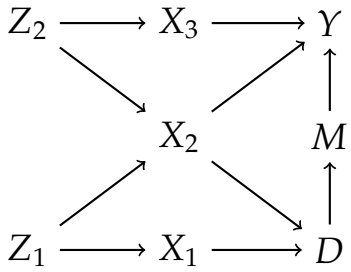


Figure 8.16: Pearl's Example

1. For Pearl's Example, write out the parents, non-parents, descendants, and non-descendants of nodes X_2 and M. List all the backdoor paths between Y and X_2. Can you identify the effect of X_2 on Y by conditioning?

2. (Front-Door-Criterion) For Pearl's Example, show that we can identify the effect $D \to M$ by conditioning on an empty set and the effect $M \to Y$ by conditioning on D. Combining the two results, we can identify the total

effect of D on Y. Solving this exercise analytically is a nice exercise; you can compare your results against causal identification packages. (Identification via this strategy is known as the Front-Door criterion; see Appendix 8.A.

3. Add an arrow $Z_2 \to Z_1$ in Pearl's Example and figure out how to identify the effect of $D \to Y$ by conditioning, of $D \to M$ by conditioning, and of $M \to Y$ by conditioning. (Note that valid conditioning sets may be empty.) Can you identify the effect of $X_2 \to Y$? If so, how? You may solve this analytically or using a causal identification package.

4. Add an arrow $X_1 \to M$ in Pearl's Example and figure out how to identify the effect of $D \to Y$ by conditioning, of $D \to M$ by conditioning, and of $M \to Y$ by conditioning. Can you identify the effect of $X_2 \to Y$? If so, how? You may solve this analytically or using a causal identification package.

5. Try to ask an instruction-following LLM (such as ChatGPT) about identification and valid adjustment sets, both for the original Pearl's Example as well as the variations in the latter two problems. Can you verify or find mistakes in the response? If you find mistakes, how might they be corrected? When mistakes are pointed out to the LLM, is it able to correct them? For example, you can try starting with the following prompt and make variations on it: "I have a causal graph with nodes Z1, Z2, X1, X2, X3, D, M, Y and edges Z1->X1, Z1->X2, Z2->X2, Z2->X3, X1->D, X2->D, X2->Y, X3->Y, D->M, M->Y. Is the effect of D on Y identified? What are the valid adjustment sets?"

8.A Front-Door Criterion via Example

We examine identification in Pearl's Example (Figure 8.2), via the front-door criterion. First note that we can write the potential outcome of interest $Y(d)$ as $Y(M(d))$, since in the SWIG $\tilde{G}(d)$ there is no other path from d to $Y(d)$ other than through $M(d)$.

$$\begin{aligned}
\mathrm{E}[Y(d)] &= \mathrm{E}[Y(M(d))] \\
&= \int \mathrm{E}[Y(M(d)) \mid M(d) = m]\mathrm{P}(M(d) = m)dm \\
&= \int \mathrm{E}[Y(m) \mid M(d) = m]\mathrm{P}(M(d) = m)dm
\end{aligned}$$

Suppose that we make a further surgery to the SWIG graph in Figure 8.3 by adding an intervention on the variable $M(d)$, i.e. take the modified SWIG graph induced by intervention $\text{fix}(D = d)$ and on that graph make a further intervention $\text{fix}(M(d) = m)$. This leads to the new SWIG:

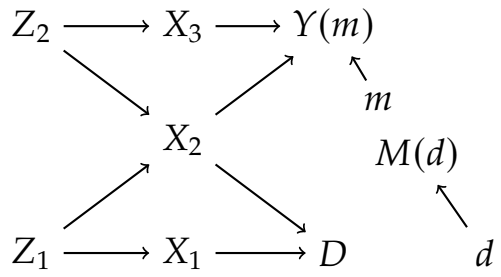


Figure 8.17: The DAG induced by a recursive Fix/SWIG intervention $\text{fix}(M(d) = m)$ on the SWIG in Figure 8.3.

Note that in this SWIG, we have $Y(m) \perp\!\!\!\perp M(d)$. Thus we have:

$$\mathrm{E}[Y(m) \mid M(d) = m] = \mathrm{E}[Y(m)],$$

leading to the front-door formula:

$$\mathrm{E}[Y(d)] = \int \mathrm{E}[Y(m)]\mathrm{P}(M(d) = m)dm$$

The term $\mathrm{E}[Y(m)]$ is the mean counterfactual response of Y when we intervene on M and $\mathrm{P}(M(d) = m)$ is the probability law of the counterfactual response of M when we intervene on D. Both of these interventional quantities can be separately identified via backdoor adjustment. More concretely, $\mathrm{E}[Y(m)] = \mathrm{E}[\mathrm{E}[Y \mid M = m, D]]$, and $\mathrm{P}(M(d) = m) = \mathrm{P}(M = m \mid D = d)$.[4] Note that under linearity assumptions on the CEFs – i.e. $\mathrm{E}[Y \mid M = m, D] = \alpha m + \beta D + c$ and $\mathrm{E}[M \mid D = d] = \gamma d + \delta$ – we get $\mathrm{E}[Y(1) - Y(0)] = \alpha\gamma$.[5] Thus, the average treatment effect $\alpha\gamma$, can be estimated by estimating α via OLS of Y on M, D and γ via OLS of M on D.

4: See Exercise 2.

5: Prove this as a reading exercise.

Bibliography

[1] Jeffrey M Wooldridge. 'Violating ignorability of treatment by controlling for too many factors'. In: *Econometric Theory* 21.5 (2005), pp. 1026–1028 (cited on page 194).

[2] Thomas S. Richardson and James M. Robins. *Single world intervention graphs (SWIGs): A unification of the counterfactual and graphical approaches to causality*. Working Paper No. 128, Center for the Statistics and the Social Sciences, University of Washington. 2013. URL: https://csss.uw.edu/files/working-papers/2013/wp128.pdf (cited on page 196).

[3] Tyler J. VanderWeele and Ilya Shpitser. 'A new criterion for confounder selection'. In: *Biometrics* 67.4 (2011), pp. 1406–1413 (cited on pages 197, 198, 200).

[4] Judea Pearl. *Causality*. Cambridge University Press, 2009 (cited on page 199).

[5] Carlos Cinelli, Andrew Forney, and Judea Pearl. 'A Crash Course in Good and Bad Controls'. In: *Sociological Methods & Research* (2022) (cited on page 201).

[6] Miguel A Hernán, Sonia Hernández-Díaz, Martha M Werler, and Allen A Mitchell. 'Causal knowledge as a prerequisite for confounding evaluation: an application to birth defects epidemiology'. In: *American Journal of Epidemiology* 155.2 (2002), pp. 176–184 (cited on pages 203, 207).

[7] A Pastuszak, D Bhatia, B Okotore, and G Koren. 'Preconception counseling and women's compliance with folic acid supplementation.' In: *Canadian Family Physician* 45 (1999), p. 2053 (cited on page 203).

[8] Rolv Terje Lie, Allen J Wilcox, and Rolv Skjærven. 'A population-based study of the risk of recurrence of birth defects'. In: *New England Journal of Medicine* 331.1 (1994), pp. 1–4 (cited on page 203).

[9] Peng Ding and Luke W. Miratrix. 'To Adjust or Not to Adjust? Sensitivity Analysis of M-Bias and Butterfly-Bias'. In: *Journal of Causal Inference* 3.1 (2015), pp. 41–57 (cited on pages 205, 206).

[10] Judea Pearl. 'Comment on Ding and Miratrix: "To Adjust or Not to Adjust?"' In: *Journal of Causal Inference* 3.1 (2015), pp. 59–60. DOI: doi:10.1515/jci-2015-0004 (cited on pages 205, 207).

[11] Cosma Rohilla Shalizi and Andrew C Thomas. 'Homophily and contagion are generically confounded in observational social network studies'. In: *Sociological Methods & Research* 40.2 (2011), pp. 211–239 (cited on page 206).

[12] Felix Elwert and Christopher Winship. 'Endogenous selection bias: The problem of conditioning on a collider variable'. In: *Annual Review of Sociology* 40 (2014), pp. 31–53 (cited on page 206).

[13] Wei Liu, M Alan Brookhart, Sebastian Schneeweiss, Xiaojuan Mi, and Soko Setoguchi. 'Implications of M bias in epidemiologic studies: a simulation study'. In: *American Journal of Epidemiology* 176.10 (2012), pp. 938–948 (cited on page 206).

[14] Eric B Schneider. 'Collider bias in economic history research'. In: *Explorations in Economic History* 78 (2020), p. 101356 (cited on page 209).

Predictive Inference via Modern Nonlinear Regression

9

"Nowhere is it written on a stone tablet what kind of model should be used to solve problems involving data."

– Leo Breiman [1].

Here we discuss nonlinear regression methods based on tree models and (deep) neural network models. Tree-based methods include regression trees, random forests, and boosted trees. Regression trees are great for exploration and explainable analytics, while random forests and boosted trees are great predictive tools for structured data and data sets of intermediate size (say, up to several million observations). Neural networks are extremely flexible nonlinear regression methods and are particularly successful for data sets of larger size.

9.1 Introduction

We are interested in predicting an outcome Y using raw regressors Z, which are k-dimensional. The best prediction rule $g(Z)$ under square loss is the conditional expectation function (CEF) of Y given Z:

$$g(Z) = \mathrm{E}(Y \mid Z).$$

In previous chapters, we used best linear prediction rules to approximate $g(Z)$ and linear regression or Lasso regression for estimation. Now we consider nonlinear prediction rules to approximate $g(Z)$, focusing on tree-based methods and neural networks.

The use of best prediction rules (CEFs) is not just important for generating good predictions but is crucial for causal inference. Identification of causal parameters such as ATEs via conditioning strategies requires us to work with CEFs rather than with best linear prediction rules. Previously we tried to make best linear prediction rules flexible to try to approximate best prediction rules. Here we explore fully nonlinear strategies.

9.2 Regression Trees and Random Forests

Introduction to Regression Trees

Regression trees are based on partitioning the regressor space (the space where Z takes on values) into a set of rectangles. A simple model is then fit within each rectangle.

The most common approach fits a simple constant model within each rectangle, which corresponds to approximating the unknown function by a "step function." Given a partition into M regions, denoted $R_1, \ldots, R_M$ the approximating function when a constant is fit within each rectangle is given by

$$f(z) = \sum_{m=1}^{M} \beta_m 1(z \in R_m),$$

where $\beta_m, m = 1, \ldots, M$ denotes a constant for each region and $1(\cdot)$ denotes the indicator function.

Suppose we have n observations (Z_i, Y_i) for $i = 1, \ldots, n$. The estimated coefficients for a given partition are obtained by

minimizing the in-sample MSE:

$$\hat{\beta} = \arg \min_{b_1,\ldots,b_M} \mathbb{E}_n \left(Y - \sum_{m=1}^{M} b_m 1(Z \in R_m) \right)^2,$$

which results in

$$\hat{\beta}_m = \text{ average of } Y_i \text{ where } Z_i \in R_m.$$

The regions $R_1, \ldots, R_M$ are called nodes, and each node R_m has a predicted value $\hat{\beta}_m$ associated with it.

A nice feature of regression trees is that you get to draw cool pictures, so let's explore their usage graphically in the context of our wage example. In this example, the outcome variable Y is (log) hourly wage; and Z includes experience, geographic, and educational characteristics.

Figure 9.1 illustrates a simple regression tree for the wage data. This tree has a depth of two, meaning that predictions are produced as a sequence of two binary decisions (or partitions of the data). Starting at the top of the tree and working down provides a simple prediction rule for any observation. For example, the predicted wage for a worker without a college degree (`college` = 0) and with less than 14 years of experience (`exper` < 14) is 12 dollars an hour. We obtain this prediction by starting at the top of the tree and taking the left branch because `college` = 0 < .5. We then go left again at the second step because `exper` < 14 and arrive at the predicted value of 12.

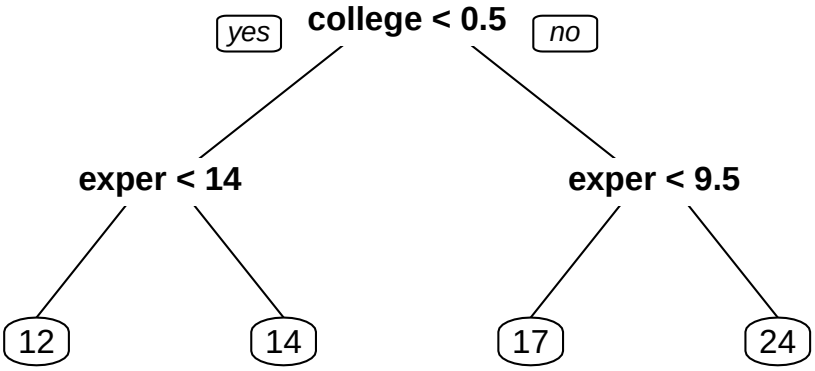


Figure 9.1: Regression tree based on wage data. The bottom nodes on the tree provide prediction rules for different subsets of observations. For example, the predicted hourly wage for a college educated worker with 9.5 or more years of experience (a worker with `college` = 1 and `exper` ≥ 9.5) is 24 dollars.

The key feature of trees is that the cut points for the partitions are adaptively chosen based on the data. That is, the splits are not pre-specified but are purely data dependent. So, how did we use the data to grow the tree in Figure 9.1?

To make computation tractable, we use recursive binary partitioning or splitting of the regressor space:

- **Growing the Tree: Level 1.** First, we cut the regressor space into two regions by choosing the regressor and splitting point such that using the prediction rule fit within each region produces the best improvement in the in-sample MSE.[1]

  Applying this procedure in the wage data gives us the depth 1 tree shown Figure 9.2. In this case, the best regressor to split on is the indicator of college degree, that takes values 0 or 1. Here splitting at any point between 0 and 1 provides the same rule, and an often used convention for binary variables is to use the "natural" split point of .5. Applying this split point yields the initial prediction rule: an hourly wage of $20 for college graduates and $13 for others.

1: To be clear, note that, in principle, finding this split point requires trying the partition produced by splitting the data along every possible value of every observed variable. That is, we are neither pre-specifying which variables nor which split points are important in providing a good prediction rule.

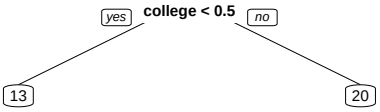


Figure 9.2: Depth 1 tree in the wage example

- **Growing the Tree: Level 2.** To grow the tree to depth 2, we then repeat the procedure for choosing the first partition rule within the two regions resulting from the first step. This step will result in a partition of the covariate space into four new regions. It is important to note that the two splits produced at this point may use different variables/splitting points than before. This feature means that the tree algorithm can create "interactions" and "nonlinearities" without requiring input from the user.

  In our example, the regions resulting from applying the first splitting rule correspond to college graduates and non-college graduates). For college graduates, the partitioning rule that minimizes in-sample MSE is to split this group into those with less than 9.5 years of experience and those with 9.5 years or more of experience. We have thus refined the prediction rule for graduates to be $24 an hour if experience is greater than or equal to 9.5 years, and $17 an hour otherwise. For non-graduates the procedure works similarly, though here the in-sample MSE minimizing split is produced by dividing non-graduates into those with less than 14 years of experience and those with 14 years of experience or more.

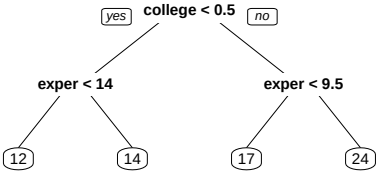


Figure 9.3: Depth 2 tree in the wage example

- **Growing the Tree: Higher Levels and Stopping Rule.** To grow deeper trees corresponding to more complex prediction rules, we simply keep repeating. We stop when the desired depth of the tree is reached,[2] or when a prespecified minimal number of observations per region, called minimal node size, is reached.

  In the wage example, we can grow a depth 3 tree by

2: One practical choice of the depth of a tree is to stop just before we get a headache from looking at a complicated tree. This rule is indeed useful if we want to present the tree as a communication device.

repeating the basic procedure within each of the four nodes of the depth 2 tree. The resulting tree is illustrated in Figure 9.4. Here, we see that the indicator for self-reported sex (`female`), high-school graduate indicator (`hsg`), and Southern region indicator (`so`) are the splitting variables chosen in the third level.

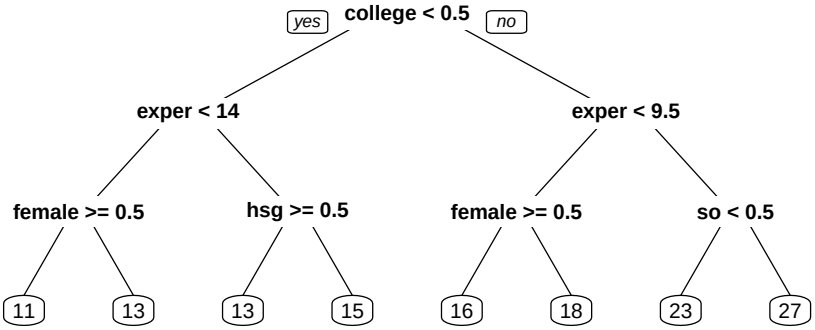


Figure 9.4: Depth 3 tree in the wage example. The depth of three was chosen to avoid getting headaches from looking at a more complicated tree.

Pruning Regression Trees. We now make several observations.

First, the deeper we grow the tree, the better is our approximation to the regression function $g(Z)$. However, the deeper the tree, the noisier our estimate $\hat{g}(Z)$ becomes, since there are fewer observations per terminal node to estimate the predicted value for this node. From a prediction point of view, we can try to find the right depth or the structure of the tree by a validation exercise such as using a single train/test split or cross-validation. For example, in the wage example, the tree of depth 2 performs better in terms of cross-validated MSE than the tree of depth 3 or 1. The process of cutting down the branches of the tree to improve predictive performance is called "Pruning the Tree."

Often for business analytics and explainability, simple trees like the ones shown are used. If we only care about building good prediction rules, we may build complicated trees and apply pruning to improve predictive performance. A simple penalty for the complexity of the tree is the number of leaves (terminal nodes) times a penalty level, where the penalty level is chosen heuristically; see, e.g, [2]. For example, we can always use a train/test split or cross-validation to settle on a penalty level. There is not a rigorously justified plug-in penalty level for trees like there is for Lasso. Figuring out such a plug-in rule is actually a good research problem.

Figure 9.5: "To prune a tree." Source: Wikipedia

Random Forests

In practice, regression trees often do not provide the best predictive performance, because a single regression tree provides a relatively crude approximation to a smooth regression function g(Z). We illustrate the potential poor approximation of regression trees in Figures 9.6 and 9.7. These figures simply illustrate that step functions, which are the outputs of typical regression tree implementations, struggle in approximating smooth functions.

A powerful and widely used approach that aims to improve upon simple regression trees is to build a random forest, as proposed by Leo Breiman [3]. The idea of a random forest is to grow many different deep trees that have low approximation error and then average the prediction rules across trees.

To produce different trees using only the observed data, the trees going into a random forest are grown from artificial data generated by sampling randomly with replacement from the original data; that is, each tree in a random forest is fit to a *bootstrap sample*.[3] Within the bootstrap samples, trees are grown deep to keep approximation error low. Averaging across the trees produced in the bootstrap samples is then meant to reduce the noisiness of the individual trees. The procedure of averaging noisy prediction rules over bootstrap samples is called Bootstrap Aggregation or Bagging. When the data set is large, we can also rely on fitting trees within *subsamples*[4] instead of using the bootstrap. Using subsamples offers some computational advantages and also simplifies theoretical analysis.

The idea seems very unusual, so let us explain again.

Each bootstrap sample is created by sampling from our data on pairs (Y_i, Z_i) randomly, with replacement. Hence, some observations are drawn multiple times and some aren't redrawn at all. Given a bootstrap sample, indexed by b, we build a tree-based prediction rule $\hat{g}_b(Z)$. We repeat the procedure B times in total, and then average the prediction rules that result from each of the bootstrap samples:

$$\hat{g}_{\text{random forest}}(Z) = \frac{1}{B}\sum_{b=1}^{B} \hat{g}_b(Z).$$

The use of the bootstrap here is unusual, yet corresponds to an intuitive idea: If we could have many independent copies of

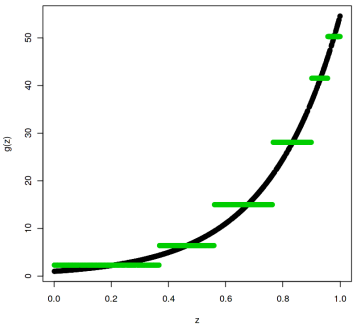


Figure 9.6: Approximation of $g(Z) = \exp(4Z)$ by a shallow regression tree in the noiseless case.

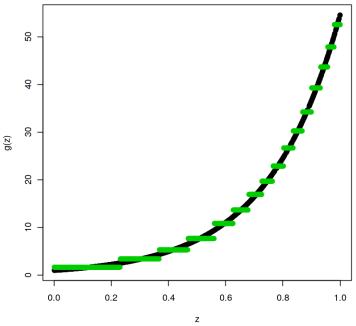


Figure 9.7: Approximation of $g(Z) = \exp(4Z)$ by a deep regression tree in the noiseless case.

3: *bootstrap sample*: typically a sample of the same or similar size to the size of the original dataset produced by sampling uniformly from the original data with replacement. Other sampling schemes may also be used, e.g. to accommodate dependence.

4: *subsample*: typically a sample of size much smaller than the original dataset produced by sampling uniformly from the original data without replacement. Other sampling schemes may also be used, e.g. to accomodate dependence.

the data, we could obtain low-bias but potentially very noisy prediction rules in each copy of the data and then average the prediction rules obtained over these copies to reduce the noise. Since we don't have many copies in reality, we rely on the bootstrap to create many quasi-copies of the data. Another feature of this idea is that the cut-points defining partitions for the tree obtained within each bootstrap sample will be different, producing a different step function approximation. Averaging over many step functions with steps at different locations will potentially produce a much smoother approximation to the underlying function. The improved approximation relative to simple trees is illustrated in Figure 9.8.

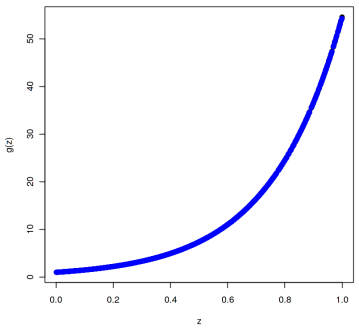


Figure 9.8: Approximation of $g(Z) = \exp(4Z)$ by a random forest in the noiseless case.

There are many modifications of the simple version of bootstrap aggregation that we have discussed. The most important modification is the use of additional randomization to "decorrelate" the trees: When we build trees over different bootstrap samples, we also randomize over the variables that trees are allowed to use in forming partitions. This additional layer of randomization encourages trees in different bootstrap samples to have different structure throughout the tree – both near the top and at the bottom – by forcing consideration of distinct sets of variables.

In summary, a random forest is an average of tree based prediction rules (a forest) produced from bootstrap or subsample data (generated randomly).

Boosted Trees

The idea of boosting is that of recursive fitting: We estimate a simple prediction rule, then take the *residuals*[5] and estimate another simple prediction rule for these residuals. We then take the residuals produced from this new prediction rules and build yet another simple model to predict them. We keep repeating this process until we reach some stopping criterion. The sum of these prediction rules fitted at each step then gives us the overall prediction rule for the outcome.

5: *residuals*: the unexplained part of an outcome we want to predict, after subtracting the prediction from the observed outcome.

Boosting can be applied with any type of base prediction rule. A common use of boosting is with regression trees which leads to *boosted trees*. Boosted trees are built up using shallow trees as the simple prediction rule. Shallow trees are trees with very few levels of depth. By keeping depth low, shallow trees produce low noise prediction rules. However, shallow trees also tend to have high approximation error because they rely on step functions with very few steps to approximate the

target regression function. That is, a single shallow regression tree tends to produce a high bias, low variance prediction rule. Boosting then helps alleviate the bias of shallow regression trees. At each step, fitting a model to the residuals from the previous step reduces the approximation error from the previous step. The improved approximation of boosted trees relative to simple trees is illustrated in Figure 9.9.

The boosting algorithm

1. Initialize the residuals: $R_i := Y_i, i = 1, \ldots, n$.
2. For $j = 1, \ldots, J$
   a) fit a tree-based prediction rule $\hat{g}_j(Z)$ to the data $(Z_i, R_i)_{i=1}^n$;
   b) update the residuals $R_i := R_i - \lambda \hat{g}_j(Z_i)$, where λ is called the learning rate.
3. Output the boosted prediction rule:

$$\hat{g}(Z) := \sum_{j=1}^{J} \lambda \hat{g}_j(Z).$$

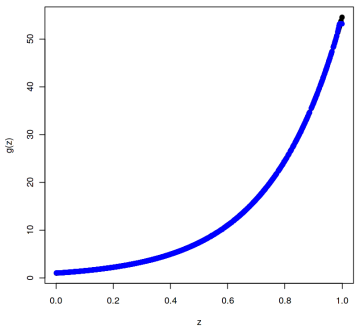


Figure 9.9: Approximation of $g(Z) = \exp(4Z)$ by boosted trees in the noiseless case with a sufficient number of steps J.

In practice, using boosted trees requires making several choices. One needs to define the tree-based prediction rule used at each step and also choose the number of learning steps, J, and the learning rate, λ. These tuning parameters are typically chosen by cross-validation.[6]

Note that the boosting algorithm is quite general and can be applied to non-tree uses. Note that the number of learning steps for boosting is important across any implementation. Because each step is building a model to predict the unexplained part of the outcome from the previous step, the in-sample prediction errors – the fit to the outcomes used to train the model – must weakly increase with each additional step. If too many iterations are taken, it is thus likely that overfitting will occur, but too few iterations may leave significant bias in the final prediction rule. In practice, the number of iterations is typically chosen by stopping the procedure once there is no marginal improvement to cross-validated MSE. A very popular implementation widely used in industry is xgboost, which has the capability to impose qualitative shape constraints like monotonicity in one or several variables. Other frequently used implementations are lightgbm and catboost.

6: We need $0 < \lambda < 1$, and a common default value for λ is 0.1. The idea of boosting is to fit simple prediction rules, so one will typically specify the prediction rule by setting the depth of the trees to a small number. For example, at each step, the prediction rule may be a regression tree of depth one (so-called stumps) or depth two. Typically, one will try several small values for depth and again choose among them by cross-validation.

9.3 Neural Nets / Deep Learning

Neural networks are a very powerful tool for modelling nonlinear relationships. They rely on many constructed regressors to approximate $g(Z)$, the conditional expectation given the regressors. The method and the name "neural networks" were loosely inspired by the mode of operation of the human brain, and developed by scientists working in Artificial Intelligence. They can be represented by cool graphs and diagrams.

Basic Ideas

First, we focus on a single layer neural network to introduce the more formal definition of neural nets. The estimated prediction rule will take the form:

$$\hat{g}(Z) := \hat{\beta}' X(\hat{\alpha}) := \sum_{m=1}^{M} \hat{\beta}_m X_m(\hat{\alpha}_m),$$

where the $X_m(\hat{\alpha}_m)$'s are constructed regressors called *neurons*,

$$\alpha = (\alpha_m)_{m=1}^{M}, \quad \beta = (\beta_m)_{m=1}^{M}, \quad X(\alpha) = (X_m(\alpha_m))_{m=1}^{M}.$$

We always take Z to include a constant of 1 as a component and set $X_1(\alpha) = 1$. The remaining neurons are generated as

$$X_m(\alpha_m) = \sigma(\alpha_m' Z), \;\; m = 2, \ldots, M,$$

where α_m's are neuron-specific vectors of parameters called weights, and σ is an activation function chosen by the practitioner. Popular activation functions are

- the sigmoid function,

$$\sigma(v) = \frac{1}{1 + e^{-v}},$$

- the rectified linear unit function (ReLU),

$$\sigma(v) = \max(0, v),$$

- the smoothed rectified linear unit function (SReLU),

$$\sigma(v) = \log(1 + \exp(v)),$$

- the leaky rectified linear unit function (Leaky-ReLU),

$$\sigma(v) = \alpha v 1(v < 0) + v 1(v \geq 0)$$

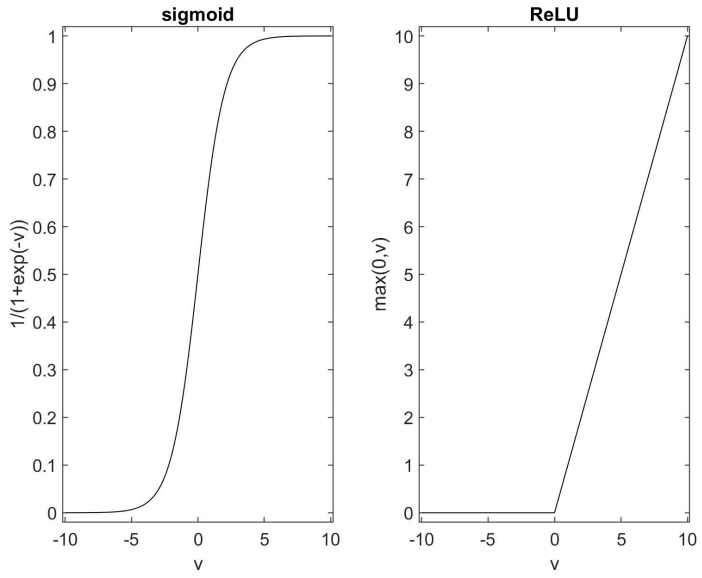


Figure 9.10: The sigmoid (logit) and ReLU activation functions

- or the linear function,

$$\sigma(v) = v.$$

The use of nonlinear activation functions is critical for generating high-quality approximations.

The estimators $\{\hat{\alpha}_m\}$ and $\{\hat{\beta}_m\}$, for $m = 1, \ldots, M$, are obtained as the solution to a penalized nonlinear least squares problem. For example, we could obtain parameter estimates by solving

$$\min_{\{\alpha_m\},\{\beta_m\}} \sum_i \left(Y_i - \sum_{m=1}^{M} \beta'_m X_{im}(\alpha_m) \right)^2 + \text{pen}(\alpha, \beta; \lambda), \quad (9.3.1)$$

where $\text{pen}(\alpha, \beta; \lambda)$ is a penalty function with penalty parameter λ. Common penalty functions are lasso-type ℓ_1 penalties,

$$\lambda \left(\sum_m \sum_j |\alpha_{mj}| + \sum_m |\beta_m| \right),$$

and Ridge-type ℓ_2 penalties,[7]

$$\lambda \left(\sum_m \sum_j (\alpha_{mj})^2 + \sum_m (\beta_m)^2 \right).$$

Neural network estimates are typically computed using stochastic gradient descent (SGD) algorithms. In its simplest version, SGD proceeds as follows: At each step, parameters are updated

7: In many implementations of neural network training the ℓ_2 penalty is referred to as the "weight decay" parameter; inspired by the fact that the ℓ_2 penalty adds an extra $-2\lambda w$ term in the gradient calculated at each gradient step of SGD for each parameter w, with w being the parameter's current value. Thus it always "decays" the parameter towards zero.

based on the update formula

$$(\alpha, \beta) \leftarrow (\alpha, \beta) - \eta \partial_{\alpha,\beta} \text{Loss}(B; \alpha, \beta)$$

where $B \subset \{1, \ldots, n\}$ is a subset of the samples and the loss is the penalized non-linear least squares objective in Equation (9.3.1) calculated on the subset B:

$$\text{Loss}(B; \alpha, \beta) := \sum_{i \in B} \left(Y_i - \sum_{m=1}^{M} \beta'_m X_{im}(\alpha_m) \right)^2 + \text{pen}(\alpha, \beta; \lambda).$$

In other words, every time we take a small step in the direction opposite to an approximate (or stochastic) version of the gradient of the loss that we want to minimize. The gradient designates the direction of parameters towards which the loss increases the most and the opposite is the direction that the loss decreases the most.[8] The magnitude of the step is controlled by the parameter η, which is many times referred to as the *step-size*.

8: This is typically referred to as the direction of *steepest descent*

In SGD, gradients are computed on subsamples of data (often consisting of a single observation) called batches, and a single cycle through all subsamples is termed an "epoch." By only making use of batches of observations, SGD algorithms are able to scale to massive data sets. Using subsamples of data introduces "stochasticity" relative to using the "full" gradient computed on the entire data. This noise in the computation of gradients also seems to have advantages in helping SGD algorithms avoid local saddle points. There are many fine practical details in terms of efficient computation of gradients for deep neural nets, how updating is done in SGD algorithms in general, and in the application of SGD to learning parameters of deep neural nets.[9]

9: These details are outside of the scope of this monograph. Interested readers might refer to *Deep Learning* by Goodfellow, Bengio, and Courville [4] for a textbook treatment of these issues. A popular method for training neural networks is called Adam; see this Towards Data Science blog for a detailed explanation [5].

The optimization methods employed for learning neural network parameters provide avenues for regularization beyond simply penalizing the size of the coefficients. A popular regularization method is *dropout* regularization where each neuron in a given layer can be set to zero with a given probability – for example, .1 – during parameter update steps. Dropout encourages more robust networks: If a particular neuron is important, the dropout regularization encourages creation of very similar neurons that can replicate the properties of the given neuron. Therefore, dropout regularization can be viewed as a penalty that forces similar weights for groups of neurons.

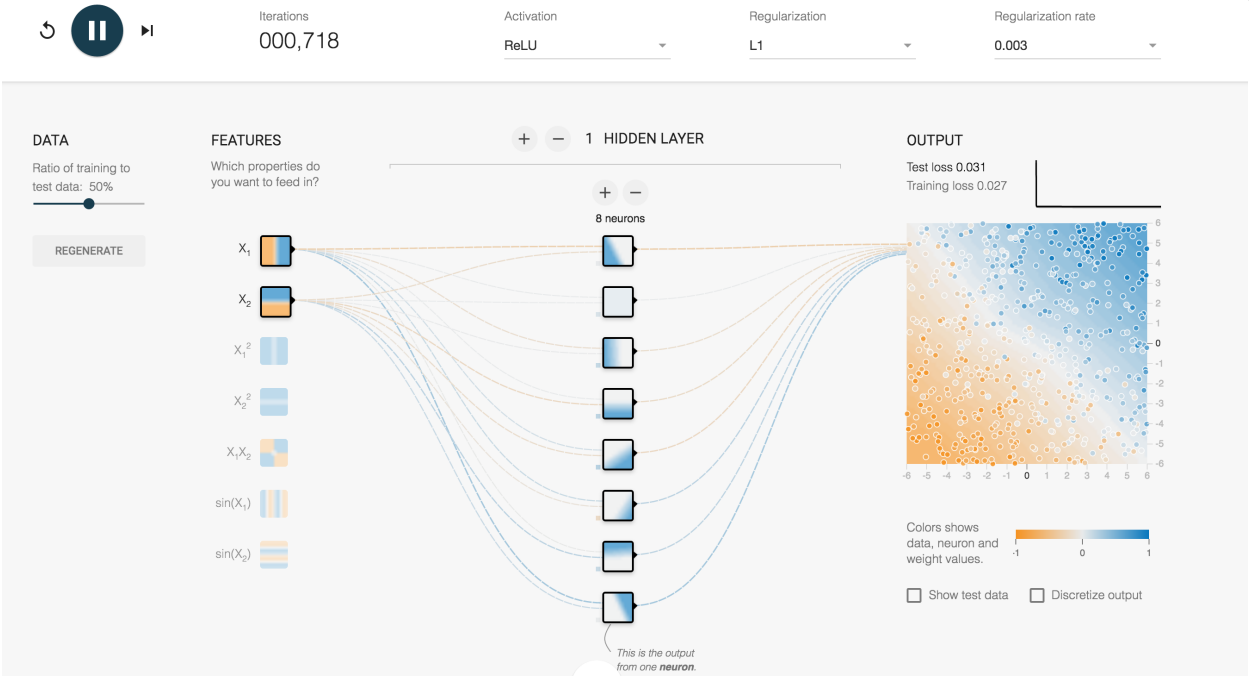
Another commonly used regularization device used with neural networks is *early stopping*. With early stopping, a measure of

out-of-sample prediction accuracy is monitored along with the value of the in-sample objective function (9.3.1). Rather than optimizing until the in-sample objective function is minimized, optimization proceeds until out-of-sample performance appears to start to degrade. By updating parameters based on in-sample fit but stopping based on out-of-sample performance, early stopping helps guard against overfitting.

As can be seen from the preceding paragraphs, using neural networks in practice relies on the choice of many tuning parameters. As there is relatively little theoretical guidance on these choices, tuning parameters are typically chosen using data splitting. An important choice that clearly relates to model flexibility is the number of neurons and neuron layers when considering the deeper networks discussed below. Having more neurons or layers gives us additional flexibility, just like having more constructed regressors provides more flexibility in high-dimensional linear models. Other choices about regularization then interact with the choice of how many neurons and layers to use in preventing overfitting.[10]

To visualize the working of a neural network, we rely on a resource called playground.tensorflow.org [7], with which we produced a prediction regression model using a simple single layer neural network model based on two input variables. A screenshot taken after training the model is shown below.

10: There has been a flurry of recent research considering the use of very large neural networks with many more parameters than the number of observations that may easily overfit the data. These papers find that such highly overparameterized neural networks tend to find solutions that generalize well.



The network depicts the process of taking raw regressors and transforming them into predicted values. In the second column

(labeled "FEATURES"), we see the inputs – our two raw regressors. The third column depicts a "hidden layer" made up of eight neurons.[11] Each neuron is constructed as a (weighted) linear combination of the raw regressors transformed by an activation function. Here we use the ReLU activation function. The neurons are connected to the inputs and the connections represent the $\hat{\alpha}_m$ coefficients. The coloring represents the sign of the coefficients (orange is negative and blue positive) and the width of the connections represents the size of the coefficients.

11: "Hidden" refers to the fact that these layers are typically not reported. However, these layers can be extracted and used as technical regressors for other tasks. We discuss using hidden layers as features in Chapter 11 which deals with feature engineering.

Finally, the neurons are combined linearly to produce the output – the prediction rule. The connections going outwards from the neurons to the output represent the coefficients $\hat{\beta}_m$ of the linear combination of the neurons that produce the final output. The coloring and the width again represent the sign and the size of these coefficients.

The output (prediction) is shown here by the "heat" map in the box on the right. On the horizontal and vertical axes we see the values of the two inputs. The color and its intensity in the "heat" map represent the predicted value.

At the top of the screenshot, we also see that we used "L1" for the type of regularization, which corresponds to using the Lasso type penalty. Here, the penalty level is called the regularization rate and is provided as the last entry in the top line of the screenshot.

In this example, we used a single layer neural network. If we add one or two additional layers of neurons constructed from the previous layer of neurons we get a "deep" network. We illustrate a two-layer network in the following figure.

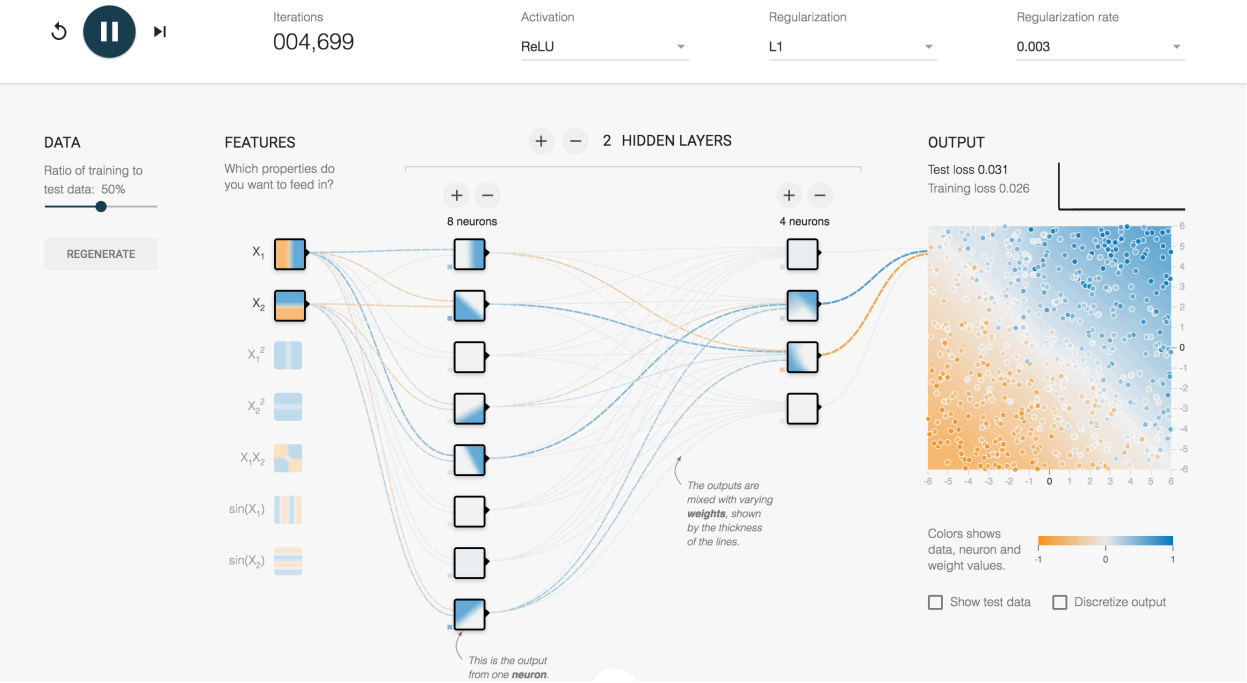
Prediction methods based on neural networks with several layers of neurons are called "deep learning" methods.

Deep Neural Networks

Here, we present the structure of a neural network with general depth. Networks with depth greater than one are called deep neural networks (DNN).

For the sake of generality, we consider networks of the multitask form, where we try to predict multiple outputs Y^t, $t = 1, ..., T$, where t stands for the "task."[12] A typical scenario is to just have one task, $T = 1$, as in all of our preceding discussion. However, there are many cases where we can use a single DNN to solve multiple tasks.

12: For example, we might be interested in predicting the price of a product using product characteristics across multiple markets or time periods, t. In treatment effect analysis, we may build a single neural network to predict both the outcome, Y, and the treatment, D, using other covariates. We could view this as a multitask learning problem where we are interested in two outputs, $Y^1 = Y$ and $Y^2 = D$.



The general nonlinear regression model we work with takes the form

$$Z \overset{f_1}{\longmapsto} H^{(1)} \overset{f_2}{\longmapsto} \quad \ldots \overset{f_m}{\longmapsto} H^{(m)} \overset{f_{m+1}}{\longmapsto} \{X^t\}_{t=1}^T, \tag{9.3.2}$$

where

$$H^{(\ell)} = \{H_k^{(\ell)}\}_{k=1}^{K_\ell}$$

are called neurons, Z is the original input, and the map f_ℓ maps one layer of neurons to the next. The maps f_ℓ are defined as

$$f_\ell : v \longmapsto \{H_k^{(\ell)}(v)\}_{k=1}^{K_\ell} := (1, \{\sigma_{k,\ell}(v'\alpha_{k,\ell})\}_{k=2}^{K_\ell}), \tag{9.3.3}$$

where $\sigma_{k,\ell}$ is the activation function that can vary with the layer ℓ and across neurons k in a given layer. We always include a constant of 1 as a component of Z, and we always designate one of the neurons in each layer up to m to be 1. The final layer, f_{m+1}, does not output the constant of 1 as a component:[13]

$$f_{m+1} : v \longmapsto \{X^t(v)\}_{t=1}^T := (\{\sigma_{t,m+1}(v'\alpha_{t,\ell})\}_{t=1}^T). \tag{9.3.4}$$

13: Common architectures employ activation functions that do not vary with k. However, custom architectures, such as ResNet50 discussed in Figure 9.13, can be viewed as having an activation function that depends on k, with some neurons linearly activated and some non-linearly.

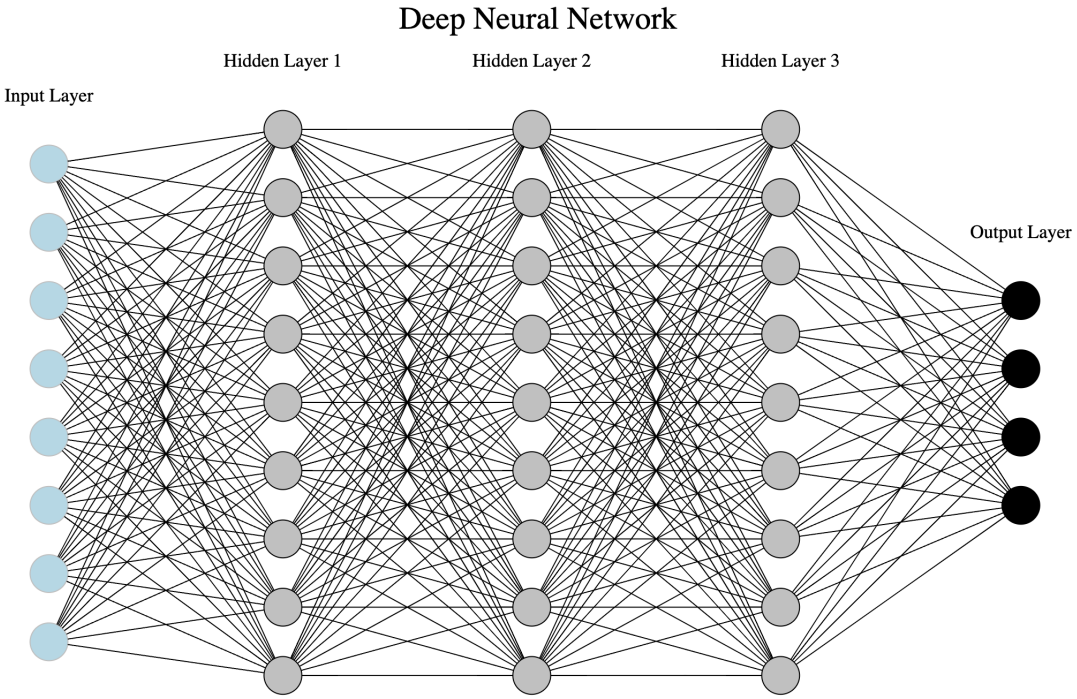


Figure 9.11: Standard Architecture of a Deep Neural Network. The input is mapped nonlinearly into the first hidden layer of the neurons. The output of this first mapping is then mapped nonlinearly into the second layer. This process is then repeated m times. The output of the penultimate layer is finally mapped (linearly or nonlinearly) into the output layer, which can have multiple outputs corresponding to different tasks.

The network mapping (9.3.2) consists of repeated composition of nonlinear mappings. This structure has been shown to be an extremely powerful tool for generating flexible functional forms which yields successful approximations in a wide range of empirical problems and is backed by approximation theory. Good approximations can be achieved by both considering sufficiently many neurons and sufficiently many layers (Yarotsky, 2017 [8]; Schmidt-Hieber, 2020 [9]; Farrell et. al, 2021 [10]; Kidger and Lyons, 2020 [11]). In empirical economic examples, it is common to just use a few hidden layers, while much deeper networks are typically used in image processing and text applications.

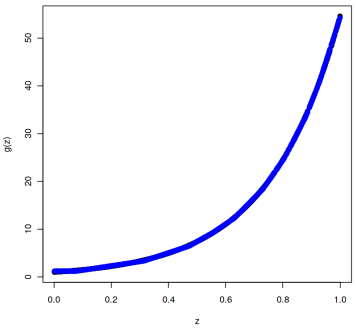


Figure 9.12: Approximation of $g(Z) = \exp(4Z)$ by a Neural Network

Similarly to single layer neural networks, the DNN model can be trained by minimizing the loss function

$$\min_{\eta \in \mathcal{N}} \sum_t w_t \sum_i (Y_i^t - X_i^t(\eta))^2 + \text{pen}(\eta; \lambda), \tag{9.3.5}$$

where η denotes all of the parameters of the mapping

$$Z_i \mapsto X_i^t(\eta),$$

w_t denotes the weight given to a task t, and $\text{pen}(\eta; \lambda)$ is a penalty function with λ denoting the penalty level.

9.4 Prediction Quality of Modern Nonlinear Regression Methods

As we have already mentioned, the best prediction rule for an outcome Y using features/regressors Z is the function $g(Z)$, equal to the conditional expectation of Y using Z:

$$g(Z) = \mathrm{E}[Y \mid Z].$$

Modern nonlinear regression methods, when appropriately tuned and under some regularity conditions, provide estimated prediction rules $\hat{g}(Z)$ that approximate the best prediction rule $g(Z)$ well.

Theoretical work demonstrates that under appropriate regularity conditions and with appropriate choices of tuning parameters, the mean squared approximation error of prediction rules produced by modern nonlinear regression methods is small once the sample size n is sufficiently large, namely,

$$\|\hat{g} - g\|_{L^2(Z)} = \sqrt{\mathrm{E}_Z[(\hat{g}(Z) - g(Z))^2]} \to 0, \quad \text{as } n \to \infty,$$

where E_Z denotes the expectation taken over Z, holding everything else fixed. To deliver these guarantees in high-dimensional settings where the number of features is large, we rely on structured assumptions, such as sparsity in the case of Lasso. Under these conditions we expect that the in-sample MSE and R^2 would agree with the out-of-sample MSE and R^2.

Learning Guarantees of DNNs

We say that a function $g : \mathbb{R}^d \to \mathbb{R}$ is β-smooth if it has $\beta \geq 1$ continuous and uniformly bounded higher-order derivatives.[14] If the regression function g is only known to be β-smooth, then the best estimator of this function has estimation error, in the worst case, that converges at the rate

$$n^{-\beta/(2\beta+d)},$$

as shown by Charles Stone [12]. When d is not small, this rate of convergence is extremely slow, suggesting that learning a function in d variables is difficult if the dimension d is moderate and the target function is only known to be β-smooth.[15]

We can achieve better rates of convergence under some kind of structured sparsity or parsimony assumptions as we saw

14: A more general definition allows β to be non-integer, but we focus on integer β for simplicity.

15: For instance, suppose that $\beta = 1$, i.e. the function is simply assumed to have a uniformly bounded first-order derivative. Moreover, suppose that we have $d = 10$ variables. Then the bound says that if we want an error of $\epsilon = 0.1$, we need n to be such that $n^{-1/12} \approx 0.1$, equivalently we need $n \approx 10^{12} = 1$ trillion samples! If $\beta = 2$, we would only need a petty 10 million samples...

in the rates for high-dimensional linear models in Chapter 3. DNNs are able to take advantage of a nonlinear form of sparsity assumptions that we formulate below following Schmidt-Hieber [9].[16]

16: See also [13] for more recent theoretical developments on provable guarantees for neural networks under sparsity conditions.

Assumption 9.4.1 (Structured Sparsity of Regression Function) *We assume that g is generated as a composition of $q+1$ vector-valued functions:*

$$g = f_q \circ \ldots \circ f_0$$

where the i-th function f_i

$$f_i : \mathbb{R}^{d_i} \to \mathbb{R}^{d_{i+1}},$$

has each of its d_{i+1} components β_i-smooth and depends only on t_i variables, where t_i can be much smaller than d_i.

The rate guarantee will depend on the parsimony/smoothness pairs:

$$(t_i, \beta_i), \quad i = 0, \ldots, q.$$

For example, consider $g : \mathbb{R}^{100} \mapsto \mathbb{R}$,

$$g(x_1, x_2, x_3, x_4, \ldots, x_{100}) = f_1(f_{01}(x_3), f_{02}(x_2)).$$

Then

$$g_0 = f_1 \circ f_0; \quad d_0 = 100, d_1 = 2; \quad t_0 = 1, t_1 = 2.$$

Theorem 9.4.1 (Learning Guarantee for DNNs under Approximate Sparsity) *Suppose that (a) the regression function g obeys the structured sparsity assumption (Assumption 9.4.1); (b) the depth of the DNN model is proportional to* $\log n$*, (c) the width of the DNN model is no less than*

$$s \cdot \log n$$

where s is the effective dimension of the regression function g,

$$s := \max_{i=0,\ldots,q} n^{\frac{t_i}{2\beta_i + t_i}};$$

and (d) other regularity conditions hold as specified in [9]. Then, there exists a sparse DNN estimator $\hat{g}$ with order $s \log n$ non-zero

parameters such that, with probability approaching 1,

$$\|\hat{g} - g\|_{L^2(Z)} \leq \text{const}_P \sigma \sqrt{\frac{s}{n}} \text{polylog}(n),$$

where $\text{polylog}(n)$ *is a polynomial in* $\log(n)$, $\sigma^2 = \mathrm{E}[(Y - g(Z))^2]$, *and* const_P *is a constant that depends on the distribution of the data.*

This fundamental result is due to Schmidt-Hieber [9], where the reader may find the complete statement of regularity conditions and further technical details of the result.

In the example above, despite the high-dimensional setting, $d = 100$, if f_{01}, f_{02}, f_{11} are β-smooth with $\beta \geq 2$, a sparse DNN is able to achieve the rate (ignoring logs):[17]

$$\sqrt{\frac{s}{n}} = n^{-\beta/(2\beta+2)} \leq n^{-1/3}$$

where the effective dimension is

$$s = n^{\frac{2}{2\beta+2}}.$$

17: Comparing to our earlier numerical example that made no sparsity assumptions, here we see that irrespective of the number of input variables, if we want the error to be $\epsilon = 0.1$, then we need $n \approx 1000$ samples, which is more realistic.

Learning Guarantees of Trees and Forests

One important property of adaptively built trees is that they are able to identify the relevant dimensions along which the regression function varies. To isolate this type of behavior of trees and forests, we consider a setting where all the regressors are binary, i.e. $Z \in \{0, 1\}^d$. This is without loss of generality for categorical (discrete-valued) regressors, since each level of the regressor can be coded as a binary indicator.[18]

Without further assumptions on the regression function $g : \{0, 1\}^d \to \mathbb{R}$, the best convergence rates that one could hope for scale at least at a $\sqrt{2^d/n}$ rate. Even for a moderate number of variables d, this rate of convergence can be prohibitively slow.

Adaptively built trees are particularly successful when there is only a small subset S, of size $|S| = r$, among the d variables that is relevant. Using this principle, we can formulate a non-parametric analogue of the sparsity assumption that we analyzed in the case of high-dimensional linear regression with Lasso that allows us to improve on the convergence rate obtained without restrictions.

18: Continuous regressors can also be discretized. However, discretization entails some loss of generality, and approximation properties following discretization have not been formally investigated.

Assumption 9.4.2 (Nonparametric Sparsity of a Regression Function with Binary Regressors) *We assume that there exists a subset S of size $|S| = r$, such that the function g can be written as a function of only the variables in S; i.e. we can write*

$$g(Z) = f(Z_S)$$

where Z_S is the subvector of Z containing only the coordinates in S.

The assumption can probably be relaxed to "approximate" sparsity.[19]

19: This relaxation has not been formally investigated.

Observe that, unlike the sparsity assumption we made in the case of high-dimensional penalized linear regression, Assumption 9.4.2 imposes no restrictions on the form of the function f that takes as input the relevant variables. Here, under the nonparametric sparsity assumption together with several other regularity conditions, we can prove that the mean squared approximation error of shallow regression trees or "honest" and arbitrarily deep regression forests[20] scales at a

$$\sqrt{2^r \log(d) \log(n)/n}$$

rate. Thus, the convergence rate depends strongly on the sparsity level r while the overall number of regressors d enter only logarithmically. Moreover, even if we knew the relevant variables S, we could not hope for a rate faster than $\sqrt{2^r/n}$ since we make no further assumptions on the function f. Thus not knowing the relevant set of regressors S adds an extra multiplicative cost on the achievable rate that only grows logarithmically with the number of regressors and the sample size. See [14] for results of similar flavor for variants of regression trees in settings beyond the binary regressor case.

20: An "honest" training approach makes use of subsampling. See Theorem 9.4.3 and the discussion immediately preceding its statement.

Theorem 9.4.2 (Learning Guarantee for Shallow Regression Trees) *Suppose that (a) the regressors are binary and the outcome variable is bounded; (b) the regression function g obeys Assumption 9.4.2; (c) regularity conditions hold that lower bound the density of the support of the distribution of covariates and upper bound the degree of variance reduction in MSE that can be achieved by features not in S [15]. Then a regression tree estimator $\hat{g}$, where the regression tree is greedily grown based on the MSE criterion up to a depth that is at least r and at most some constant multiple of r,*

A greedy algorithm is any algorithm that follows the problem-solving heuristic of making the locally optimal choice at each stage. In our case, a greedily grown tree optimizes over the name of regressor and splitting point that achieve the best one-step improvement in the in-sample MSE at each node.

satisfies, for $n \geq \text{const}_P 2^r \log(d/\delta)$*, with probability* $1-\delta$*,*

$$\|\hat{g} - g\|_{L^2(Z)} \leq \text{const}_P \sigma \sqrt{\frac{2^r \log(d/\delta)\log(n)}{n}},$$

where $\sigma^2 = \mathrm{E}[(Y - g(Z))^2]$ *and* const_P *is a constant that depends on the distribution of the data.*

Capping the depth of the regression tree as in Theorem 9.4.2 helps avoid overfitting, since otherwise we could potentially construct binary trees that achieve zero error on the training data and have large error out-of-sample.

An alternative to avoiding overfitting is to use an ensemble approach based on sub-sampled data. To implement an ensemble approach, we train multiple regression trees, each on a random sub-sample (without replacement) of the original data-set of size $s < n$ and average the predictions of each of these trees. Moreover, to formally argue about the approximation error of such sub-sampled forests, we will require the forests to be trained in an "honest" manner.

In our setting, an honest training approach is as follows: When we train a tree on a sub-sample, we randomly partition the data in half and we use half of the data to find the best splits in a greedy manner, and the other half of the data to construct the estimates at each node of the tree. Such sub-sampled honest forests have been recently popularized by the work of [16]. Subsequent work of [15] showed that honest forests provably adapt to non-parametric sparsity of the regression function.

Theorem 9.4.3 (Learning Guarantee for Sub-Sampled Honest Forests) *Suppose that (a) the regressors are binary and outcome variable is bounded; (b) the regression function* g *obeys Assumption 9.4.2; (c) regularity conditions hold that lower bound the density of the support of the distribution of covariates and upper bound the degree of variance reduction in MSE that can be achieved by features not in* S *[15]. Then a regression forest estimator* $\hat{g}$*, where each regression tree is built in an honest manner and on a random sub-sample (without replacement) of size* $s = \text{const}_P\, 2^r\, \log(d/\delta)$ *of the original data, satisfies, for* $n \geq \text{const}_P\, 2^r\, \log(d/\delta)$ *with probability* $1-\delta$*,*

$$\|\hat{g} - g\|_{L^2(Z)} \leq \text{const}_P \sigma \sqrt{\frac{2^r\ \log(d/\delta)\ \text{polylog}(n)}{n}}$$

where $\sigma^2 = \mathrm{E}[(Y - g(Z))^2]$ *and* const_P *is a constant that depends on the distribution of the data and* $\text{polylog}(n)$ *is a polynomial factor*

of $\log(n)$.

The rate guarantee for Honest Forests in Theorem 9.4.3 is the same as the rate for shallow trees in Theorem 9.4.2. This theory thus does not shed light on why random forests seem to achieve superior predictive performance over simple trees in many applications. Moreover, practical random forest algorithms tend to work well with default tuning choices, whereas the theory requires a careful alignment of the tuning parameters to get good rate guarantees. The regularity conditions also require the explanatory power of the subset of the covariates that are relevant, S, to dominate the explanatory power of the irrelevant covariates.[21] This condition on signal strength is a sufficient condition, but it may not be necessary for good performance. That is, there seem to remain substantial gaps in our theoretical understanding of the performance of tree-based algorithms. Further exploring these properties may be an interesting area for further study.

21: Irrelevance here only means that, given the set S of relevant covariates, the other variables do not contribute to the best prediction rule. It does not mean that the irrelevant covariates have no predictive power on their own.

Trust but Verify

Both tree-based methods and neural networks provide powerful, flexible models that can deliver high-quality approximations of regression functions. However, the high degree of flexibility can lead to overfitting. Therefore, it is always important to verify the performance on test data to make sure that the predictive model being used is actually a good one.

A simple verification procedure is data splitting, which can be performed in the following way:

1. We use a random subset of data for estimating/training the prediction rule.
2. We use the other part of the data to evaluate the quality of the prediction rule, recording out-of-sample mean squared error, R^2, or some other desired measure of prediction quality.

Recall that the part of the data used for estimation is called the training sample. The part of the data used for evaluation is called the testing or validation sample. We have a data sample containing observations on outcomes Y_i and features Z_i. Suppose we use n observations for training and m for testing/validation. We use the training sample to compute prediction rule $\hat{g}(Z)$. Let V denote the indices of the observations in the

test sample. Then the out-of-sample/test mean squared error is

$$\text{MSE}_{test} = \frac{1}{m}\sum_{k\in V}(Y_k - \hat{g}(Z_k))^2.$$

The out-of-sample/test R^2 is[22]

$$R^2_{test} = 1 - \frac{\text{MSE}_{test}}{\frac{1}{m}\sum_{k\in V} Y_k^2}.$$

22: In typical empirical applications, these quantities are calculated after de-meaning/centering the outcome.

A Simple Case Study using Wage Data

We illustrate ideas using a data set of 5150 observations from the March Current Population Survey Supplement 2015. Y_i's are log wages of never-married workers living in the U.S. Z_i's include experience, education, 23 industry and 22 occupation indicators, and some other characteristics. We consider a variety of linear and nonlinear rules for predicting Y with Z.

For the linear models, we estimate prediction rules of the form $\hat{g}(Z) = \hat{\beta}'X$ using X generated in two ways:

- (basic model) X consists of the 51 raw regressors in Z.
- (flexible model) X consists of 246 variables composed of the 51 raw regressor in Z, a fourth order polynomial in experience, and two-way interactions between the polynomial terms in experience and the non-experience variables in Z.

We estimate $\hat{\beta}$ by linear regression/least squares and by the following penalized regression methods: Lasso and Post-Lasso with plug-in choice of λ, cross-validated Lasso, Ridge, and Elastic Net.

For the nonlinear models, we estimate prediction rules of the form $\hat{g}(Z)$ without imposing that $\hat{g}(Z) = \hat{\beta}'X$. That is, we do not assume prediction rules to be linear. We estimate the prediction models by random forests, regression trees, boosted trees, and Neural Networks. We use an implementation of the random forest where, at the step of growing a regression tree, we choose the best variable to split upon among $\sqrt{p} \ll p$ randomly selected variables.

Table 9.1 displays results based upon a single split of data into training and testing sets. It shows the test MSE in column 1, the standard error of the test MSE in column 2, and the test R^2 in column 3. We see that the best performing prediction rules are provided by OLS using the raw 51 regressors and Lasso using the basic 51 predictors with penalty parameter selected by

	MSE	S.E.	R^2
Least Squares (basic)	0.229	0.016	0.282
Least Squares (flexible)	0.243	0.016	0.238
Lasso	0.234	0.015	0.267
Post-Lasso	0.233	0.015	0.271
Lasso (flexible)	0.235	0.015	0.265
Post-Lasso (flexible)	0.236	0.016	0.261
Cross-Validated Lasso	0.229	0.015	0.282
Cross-Validated Ridge	0.234	0.015	0.267
Cross-Validated Elastic Net	0.230	0.015	0.280
Cross-Validated Lasso (flexible)	0.232	0.015	0.275
Cross-Validated Ridge (flexible)	0.233	0.015	0.271
Cross-Validated Elastic Net (flexible)	0.231	0.015	0.276
Random Forest	0.233	0.015	0.270
Boosted Trees	0.230	0.015	0.279
Pruned Tree	0.248	0.016	0.224
Neural Net	0.276	0.012	0.148

Table 9.1: Prediction Performance for the Test/Validation Sample.

cross-validation. The performance of both Elastic Net with the basic set of regressors and boosted trees are also nearly identical to those of the two best methods. Looking at standard errors, we see that the vast majority of methods have test MSE's that are within one standard error of the best test MSE, suggesting relatively little difference in performance across methods.

The outliers, in terms of performing relatively poorly, are OLS using the flexible set of covariates as well as the regression tree (Pruned Tree) and the neural net. OLS with the flexible set of predictors uses a relatively large number of variables relative to the sample size and seems likely to be overfit. On the other hand, neither the regression tree nor the neural net is fully tuned. Thus, there may be room to improve the performance of these methods.

9.5 Combining Predictions - Aggregation - Ensemble Learning

Given different prediction rules, we can choose either a single method or an aggregation of several methods as our prediction approach. An aggregated prediction is a linear combination of the basic predictors.

In econometrics and statistics, the procedures for combining several methods are called "model averaging" and "aggregation." In machine learning, these terms are relabeled as "ensembles" and "stacking."

Specifically, we consider an aggregated prediction rule of the

form:

$$\tilde{g}(Z) = \sum_{k=1}^{K} \tilde{\alpha}_k \hat{g}_k(Z),$$

where $\hat{g}_k$'s denote basic predictors, potentially including a constant. The basic predictors are computed on the training data.

If the number of prediction rules, K, is small, we can figure out the coefficients of the optimal linear combination of the rules, $\tilde{\alpha}_k$, using test data V by simply running least squares of the outcomes in the test data on their associated predicted values:

$$\min_{(\alpha_k)_{k=1}^K} \sum_{i \in V} \left(Y_i - \sum_{k=1}^{K} \alpha_k \hat{g}_k(Z_i) \right)^2.$$

We wish to emphasize that here we are minimizing the sum of squared prediction errors in the test sample using the prediction rules from the training sample as the regressors. If K is large, we can instead use Lasso for aggregation:

$$\min_{(\alpha_k)_{k=1}^K} \sum_{i \in V} \left(Y_i - \sum_{k=1}^{K} \alpha_k \hat{g}_k(Z_i) \right)^2 + \lambda \sum_{k=1}^{K} |\alpha_k|.$$

Aggregation Results for the Case Study

We consider the prediction rules based on OLS, Post-Lasso, Elastic Net, Pruned Tree, random forest and boosted trees to build an ensemble method.

	Weight OLS	Weight Lasso
Constant	-0.162	-0.147
Least Squares (basic)	0.281	0.293
Post-Lasso (flexible)	0.237	0.223
CV Elastic Net (flexible)	-0.068	-0.056
Pruned Tree	-0.140	0.000
Random Forest	0.377	0.344
Boosted Trees	0.367	0.245

Table 9.2: Weights of the ensemble method.

The estimated weights are shown in Table 9.2. The adjusted R^2 for the test sample gets improved by about 1%.

Auto ML Frameworks

There are a variety of new frameworks emerging that do automated search and aggregation of different prediction methods. These automatic aggregation procedures use approaches like the one we outlined above or other heuristics. Example implementations of automatic aggregation methods include H20, AutoML [17], Auto Gluon [18] (which relies on Neural Nets), Auto-Sklearn, Hyperopt-Sklearn and FLAML.

We've tried H20 on the wage data. It produced a model that beats OLS with the basic predictor set, which gave a test MSE of 0.229, by producing a test MSE of 0.21. (The difference is not statistically significant.) H20 is similar to the ensemble method that we constructed above. The performance was very impressive because we gave H20 a time budget of just 100 seconds!

9.6 When Do Neural Networks Win?

The wage example may give a pessimistic impression on the power of deep learning (and machine learning more generally). A more optimistic impression emerges from examining performance of deep learning in data-rich settings, where large samples and rich features are available.

A recent example comes from Bajari et al. (2021) [19]. Here we are interested in predicting prices of products given their characteristics, which include both text and images. The resulting predictions are called hedonic prices. In this example, neural networks (specifically BERT [20] and ResNet50 [21]) are first used to convert the text and image data into several thousand-dimensional numerical features X (called embeddings). These features extracted from the text and image data are then used as input variables in a deep neural network for predicting product prices. The deep neural network used in the example consists of 3 hidden layers, with the penultimate layer consisting of about 400 neurons.

The features produced in the penultimate layer in a deep neural network are often referred to as embeddings as they encode or "embed" the information from the previous layers that is directly used in producing the final predictions. In the case of hedonic pricing, we may refer to these features as "value embeddings" as the final target is price or value of the product.

The data set used in this example is larger than 10 million observations. The accuracy of prediction for the deep neural network described above, as measured by the R^2 on the test sample, is about 90%. In contrast, random forests applied to predict prices using the text and image embeddings as inputs deliver an R^2 in the test sample that is in the ballpark of 80%, and a linear model estimated via least squares that uses the

text and image embeddings as predictor variables delivers an R^2 in the test sample of only around 70%. Ignoring the neural network embeddings of the text and image data and using only simple catalog features, the R^2 is lower than 40%.

We will discuss further details of generating embeddings in Chapter 11.

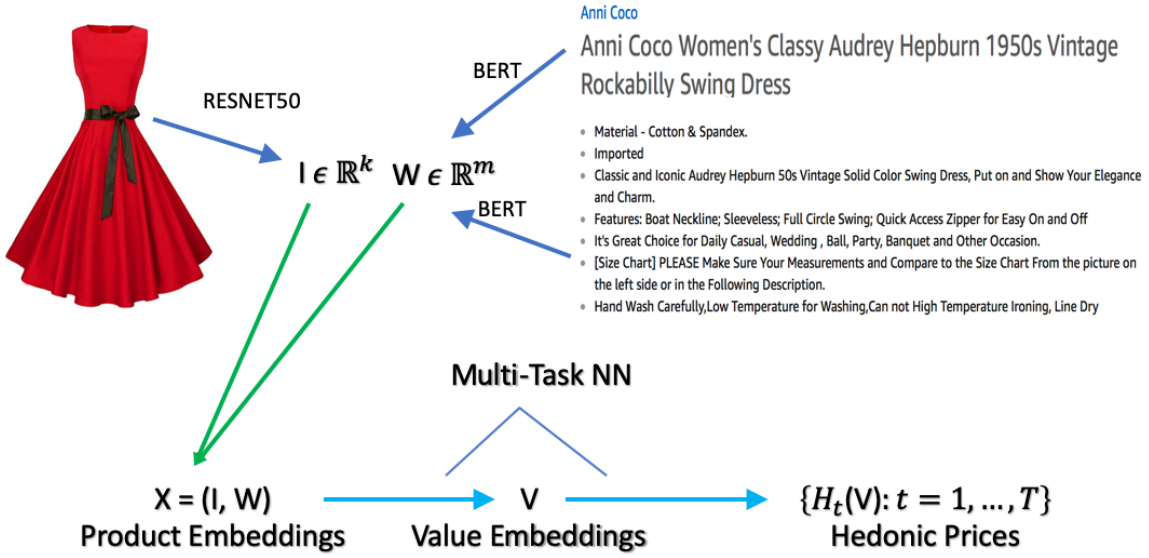


Figure 9.13: The structure of the predictive model in Bajari et al. (2021) [19]. The input consists of images and unstructured text data. The first step of the process creates the moderately high-dimensional numerical embeddings I and W for images and text data via state-of-the art deep learning methods, such as ResNet50 and BERT. The second step of the process takes as input $X = (I, W)$ and creates predictions for hedonic prices $H_t(X)$ using deep learning methods with a multi-task structure. The models of the first step are trained on tasks unrelated to predicting prices (e.g., image classification or word prediction), where embeddings are extracted as hidden layers of the neural networks. The models of the second step are trained by price prediction tasks. The multitask price prediction network creates an intermediate lower dimensional embedding $V = V(X)$, called value embedding and then predicts the final prices in all time periods $\{H_t(V), t = 1, ..., T\}$. Some variations of the method include fine-tuning the embeddings produced by the first step to perform well for price prediction tasks (i.e. optimizing the embedding parameters so as to minimize price prediction loss).

9.7 Closing Notes

To sum up, we have discussed assessment of predictive performance of modern linear and non-linear regression methods using splitting of data into training and testing samples. The results could be used to pick the best prediction rule generated by the classical or modern regression methods or to aggregate prediction rules into an ensemble rule, which can result in some improvements. We illustrated these ideas using the wage data from the 2015 Current Population Survey. We finally introduced Auto ML frameworks and commented that Neural Networks perform best in very data-rich settings.

Notebooks

- Python Notebook on ML-based Prediction of Wages and R Notebook on ML-based Prediction of Wages provide details of implementation of penalized regression, regression trees, random forest, boosted tree and neural network methods, a comparison of various methods and a way to choose the best method or create an ensemble of methods. Moreover, they provide an application of the FLAML (Python) and H2O (R) AutoML framework to the wage prediction problem. With a small time budget, both FLAML and H2O found the model that worked best for predicting wages.

- Python Notebook on Approximation of a Function by Random Forest and Neural Network and R Notebook on Approximation of a Function by Random Forest and Neural Network illustrate the flexibility of these methods in approximating the function $\exp(4x)$.

Additional resources

- Andrej Karpathy [22] 's Recipe for Training Neural Networks provides a good workflow and practical tips for training good neural network models.

- For practical details of tree-based methods, please see Hastie et al. [23] 's book "Introduction to Statistical Learning".

- For an in-depth treatment of deep learning, see Zhang's et al. [24] 's book "Dive Into Deep Learning", Goodfellow et al. [4] "Deep Learning", and Nielsen [25] "Neural Networks and Deep Learning".

Notes

Many of the formative developments in modern nonlinear regression were led by the statistics and artificial intelligence communities. The methods were rebranded as machine learning in the 90s, and learning with neural networks was rebranded as deep learning when it was realized that deep network architectures produced phenomenal results in image classification (and later in natural language processing tasks). The success

of deep neural networks was a breakthrough associated with advances in both computing power and the ability to collect very large data sets. See the textbooks mentioned above for in-depth treatments of deep learning.

In Chapter 10, we will study the use of the machine learning and deep learning for statistical inference on causal and predictive effects in high-dimensional nonlinear regression settings; and in Chapter 11, we'll be using deep learning for engineering features from text and data (e.g. using images and product descriptions as "regressors").

Study Problems

1. Use two paragraphs to explain to a friend how one of the tree-based strategies works.

2. Use two paragraphs to explain to a friend how a basic neural network works.

3. Experiment with one of the empirical notebooks provided and summarize your findings. For example, try to see if you can build a better performing neural network in the wage example. One possibility is to use custom models in Keras, where we can construct a partially linear model that borrows the strength of the basic linear model and corrects it slightly with a nonlinear deviation function.

4. Experiment with the last (non-empirical) notebook. See, for example, if you can find a (much) simpler neural network that provides the same quality of fit as the current example in the notebook.

9.A Variable Importance via Permutations

There are many ways of assessing variable importance in nonlinear models. A very simple one is the following permutation method.

The importance of variable j in any machine learning algorithm (linear or nonlinear) can be defined by computing the loss in predictive performance that results from replacing the observations

of the j-th feature $(Z_{ji})_{i=1}^n$ with their random permutation

$$(Z_{j\pi(i)})_{i=1}^n,$$

where $\pi : \{1, ..., n\} \to \{1, ..., n\}$ is a permutation map, generated at random. The loss is averaged over many random permutations, to obtain an average loss measure L_j. Then the variables are ranked in terms of L_j, from largest to smallest. The top-ranked variables are the most important ones. This idea, that appeared in the original paper by L. Breiman [3], mimics the situation where the permuted regressor is an irrelevant predictor having the same marginal distribution as the observed regressor.

Bibliography

[1] Leo Breiman. 'Statistical modeling: The two cultures'. In: *Statistical science* 16.3 (2001), pp. 199–231 (cited on page 215).

[2] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The Elements of Statistical Learning*. Vol. 1. Springer series in statistics New York, 2001 (cited on page 219).

[3] Leo Breiman. 'Random forests'. In: *Machine learning* 45.1 (2001), pp. 5–32 (cited on pages 220, 243).

[4] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016 (cited on pages 225, 241).

[5] Lili Jiang. *A Visual Explanation of Gradient Descent Methods (Momentum, AdaGrad, RMSProp, Adam)*. 2020. URL: https://towardsdatascience.com/a-visual-explanation-of-gradient-descent-methods-momentum-adagrad-rmsprop-adam-f898b102325c (visited on 04/03/2022) (cited on page 225).

[6] Peter L Bartlett, Andrea Montanari, and Alexander Rakhlin. 'Deep learning: a statistical viewpoint'. In: *Acta Numerica* 30 (2021), pp. 87–201 (cited on page 226).

[7] *playground.tensorflow.org*. https://playground.tensorflow.org/. Accessed: 2022-04-03 (cited on page 226).

[8] Dmitry Yarotsky. 'Error bounds for approximations with deep ReLU networks'. In: *Neural Networks* 94 (2017), pp. 103–114 (cited on page 229).

[9] Johannes Schmidt-Hieber. 'Nonparametric regression using deep neural networks with ReLU activation function'. In: *Annals of Statistics* 48.4 (2020), pp. 1875–1897 (cited on pages 229, 231, 232).

[10] Max H. Farrell, Tengyuan Liang, and Sanjog Misra. 'Deep Neural Networks for Estimation and Inference'. In: *Econometrica* 89.1 (2021), pp. 181–213 (cited on page 229).

[11] Patrick Kidger and Terry Lyons. 'Universal Approximation with Deep Narrow Networks'. In: *Proceedings of Thirty Third Conference on Learning Theory*. Ed. by Jacob Abernethy and Shivani Agarwal. Vol. 125. Proceedings of Machine Learning Research. PMLR, 2020, pp. 2306–2327 (cited on page 229).

[12] Charles J. Stone. 'Optimal global rates of convergence for nonparametric regression'. In: *Annals of statistics* 10.4 (1982), pp. 1040–1053 (cited on page 230).

[13] Rahul Parhi and Robert D Nowak. 'Deep Learning Meets Sparse Regularization: A Signal Processing Perspective'. In: *arXiv preprint arXiv:2301.09554* (2023) (cited on page 231).

[14] Stefan Wager and Guenther Walther. 'Adaptive concentration of regression trees, with application to random forests'. In: *arXiv preprint arXiv:1503.06388* (2015) (cited on page 233).

[15] Vasilis Syrgkanis and Manolis Zampetakis. 'Estimation and Inference with Trees and Forests in High Dimensions'. In: *Proceedings of Thirty Third Conference on Learning Theory*. Ed. by Jacob Abernethy and Shivani Agarwal. Vol. 125. Proceedings of Machine Learning Research. PMLR, 2020, pp. 3453–3454 (cited on pages 233, 234).

[16] Stefan Wager and Susan Athey. 'Estimation and Inference of Heterogeneous Treatment Effects using Random Forests'. In: *Journal of the American Statistical Association* 113.523 (2018), pp. 1228–1242 (cited on page 234).

[17] Erin LeDell and Sebastien Poirier. 'H2o automl: Scalable automatic machine learning'. In: *Proceedings of the AutoML Workshop at ICML*. Vol. 2020. 2020 (cited on page 239).

[18] Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. 'Autogluon-tabular: Robust and accurate automl for structured data'. In: *arXiv preprint arXiv:2003.06505* (2020) (cited on page 239).

[19] Patrick L. Bajari, Zhihao Cen, Victor Chernozhukov, Manoj Manukonda, Jin Wang, Ramon Huerta, Junbo Li, Ling Leng, George Monokroussos, Suhas Vijaykunar, et al. *Hedonic prices and quality adjusted price indices powered by AI*. Tech. rep. cemmap working paper, 2021 (cited on pages 239, 240).

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 'Bert: Pre-training of deep bidirectional transformers for language understanding'. In: *arXiv preprint arXiv:1810.04805* (2018) (cited on page 239).

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 'Deep residual learning for image recognition'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778 (cited on page 239).

[22] Andrej Karpathy. *A Recipe for Training Neural Networks*. 2019. URL: http://karpathy.github.io/2019/04/25/recipe/ (visited on 04/06/2022) (cited on page 241).

[23] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*. Springer, 2013 (cited on page 241).

[24] Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. 'Dive into deep learning'. In: *URL: https://d2l. ai* (2020) (cited on page 241).

[25] Michael A. Nielsen. *Neural networks and deep learning*. Determination Press, 2015 (cited on page 241).

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

10

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

– Jerzy Neyman [1].

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

10.1 Introduction

We recall the predictive effect question:

- How does the predicted value of the outcome,

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

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

This question may have a causal interpretation within any SEM, where conditioning on X is sufficient for identification of the causal effect of D on Y. When this condition holds, the question becomes the causal effect question:

- How does the predicted value of the potential outcome,

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

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

Both questions are interesting and useful to ask, depending on the application. In what follows, we set up double/debiased machine learning (DML) methods for answering these questions with data.[1] These statistical inference methods *do not* distinguish between the two types of questions, so the methods are equally applicable to answering both types.

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

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

The DML method requires a Neyman-orthogonal representation of the target parameters to reduce the spillover of regularization biases inherent in ML methods onto the estimation of the target parameter. The method also makes use of cross-fitting: an efficient form of sample splitting that eliminates biases that may arise from overfitting.

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

10.2 DML Inference in the Partially Linear Regression Model (PLM)

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

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

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

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

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

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

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

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

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

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

In what follows, we will employ the partialling out X operation of the form that inputs a random variable V and outputs the

residualized form:

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

Applying this operation to (10.2.1) we obtain

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

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

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

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

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

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

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

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

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

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

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

Our estimation procedure for β in the sample will mimic the partialling out procedure in the population. We also rely on cross-fitting (outlined below) to make sure our estimated residualized quantities are not overfit.

Double/Orthogonal ML for the Partially Linear Model

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

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

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

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

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

In what follows it will be convenient to use the notation

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

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

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

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

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

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

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

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

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

where

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

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

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

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

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

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

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

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

Therefore, we can select the best ML method for estimating m and the best method for estimating ℓ to minimize the upper bound on the bias. We will be using mean square prediction errors as proxies for MSAEs.

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

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

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

  where the index j reflects the dependency of residuals

on the method.

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

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

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

Two different ML methods may be the best for predicting Y and predicting D. By doing MSPE minimization we in fact minimize MSAEs, since MSPEs approximate MSAEs plus terms that do not depend on j.

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

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

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

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

Discussion of DML Construction

The partialling out operation causes the moment equations defining β to be Neyman-orthogonal. That is, the moment conditions are insensitive to perturbations of the nuisance

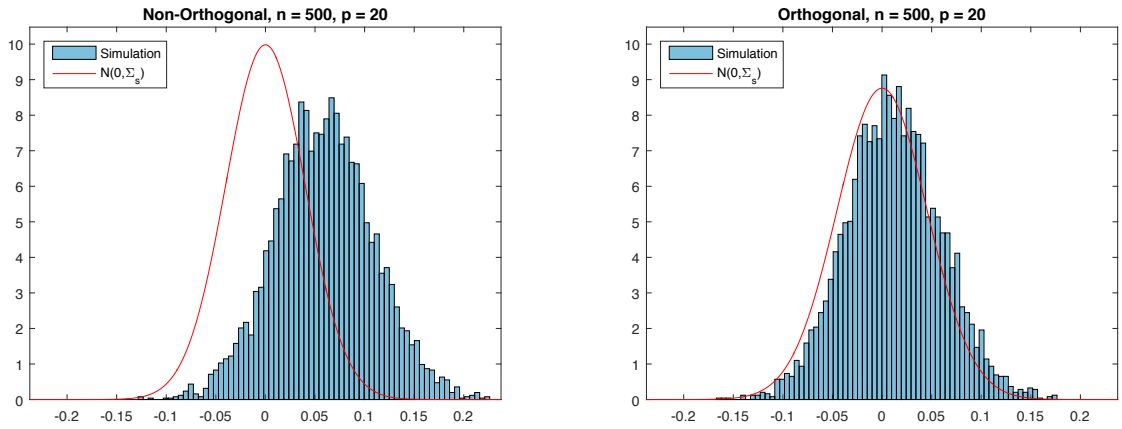


Figure 10.1: Left: Behavior of a conventional (non-orthogonal) ML estimator. Right: Behavior of the orthogonal, DML estimator.

parameters ℓ and m.[2] We discussed Neyman-orthogonality in the context of high-dimensional linear regression models in Chapter 4. We return to and generalize this discussion formally in Section 10.4. This property allows us to get rid of the bias in estimation of m and ℓ that arises when ML estimators are applied in high-dimensional settings.

Naive application of machine learning methods directly to outcome equations may lead to highly biased estimators, because the resulting strategy is not Neyman-orthogonal. The biases in estimation of g, which are unavoidable in high-dimensional estimation, create a non-trivial bias in the estimate of the main effect. This bias is large enough to cause failure of conventional inference.

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

2: Generally we use the term nuisance parameters to name parameters that are not the target parameters. Here the target parameter is β and ℓ and m are nuisance parameters.

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

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

to identify β and uses a biased estimate of g in place of g. This moment strategy is sensitive to deviations away from the true value. Indeed, let us compute the directional derivative in the direction Δ away from the true value:

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

The derivative generally does not vanish, and the biases in estimation of g will transmit to the estimation of β.

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

The DML algorithm uses a form of sample splitting, called cross-fitting, to make sure our estimated residualized quantities are not overfit. Biases arising from overfitting could result from using highly complex fitting methods such as boosting, deep neural networks and random forests. If we don't do sample splitting and the ML estimates overfit, we may end up with very large biases.

Figure 10.2 illustrates how the bias resulting from overfitting in the estimation of nuisance functions can cause the DML (without sample splitting) to be biased and how sample splitting eliminates this problem. In the left panel the histogram shows the finite-sample distribution of the DML estimator in the partially linear model in a simple simulation experiment where nuisance parameters are estimated with overfitting using the full sample, i.e. without sample splitting. The finite-sample distribution is clearly shifted to the left of the true parameter value, demonstrating the substantial bias. In the right panel, the histogram shows the finite-sample distribution of the DML estimator in the same simulation experiment in the partially

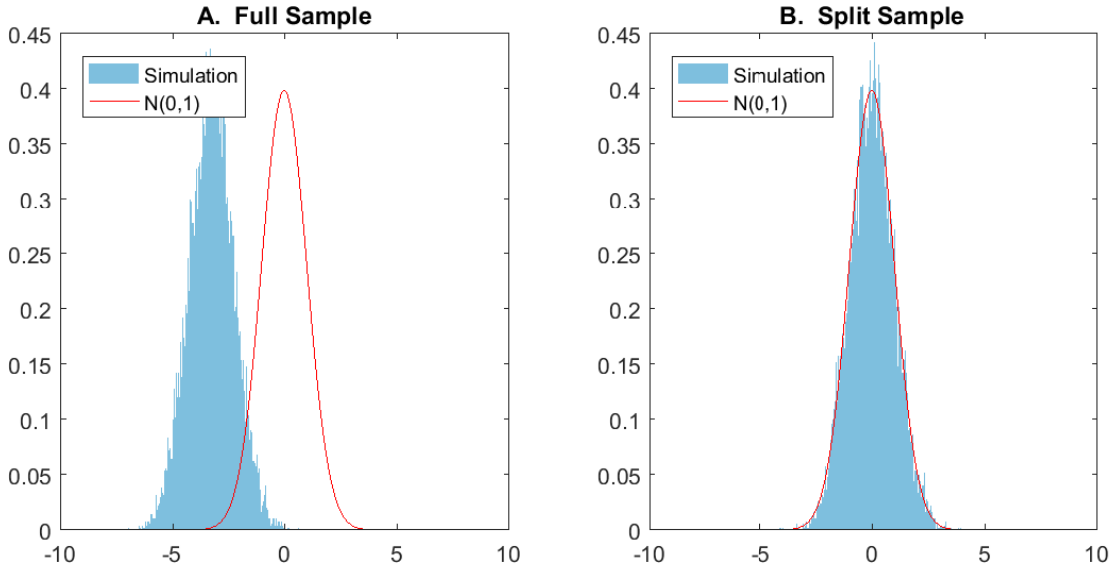


Figure 10.2: Left: DML distribution without sample-splitting. Right: DML distribution with cross-fitting.

linear model where nuisance parameters are estimated with sample-splitting using the cross-fitting estimator. Here, we see that the use of sample-splitting has completely eliminated the bias induced by overfitting.

Remark 10.2.5 (On overfitting) Note that previously in the context of high-dimensional approximately sparse linear models we were using Lasso with the plug-in choice for the penalty level λ which ensures that overfitting is sufficiently well-controlled that we didn't have to use sample splitting. Such refined, theoretically rigorous choices of tuning parameters are not yet available for other machine learning methods. In practice, experienced researchers and machine learning engineers often use intuition, heuristics, and other empirical tools (six packs or witchcraft tables, for example) to set the tuning parameters. While the resulting methods can perform well for prediction purposes, even modest overfitting can result in large biases in DML, as we illustrate in the simulation experiment. Therefore, it is simply safer to rely on sample-splitting in real settings with complicated learners to make sure overfitting of during estimation of our residualized quantities does not contaminate out estimates of the objects of interest.

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

The Effect of Gun Ownership on Gun-Homicide Rates

We consider the problem of estimating the effect of gun ownership on the homicide rate.[3] For this purpose, we estimate the partially linear model:

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

$Y_{i,t}$ is the log homicide rate in county i at time t. $D_{i,t-1}$ is the log fraction of suicides committed with a firearm in county i at time $t-1$, which we use as a proxy for gun ownership $G_{i,t}$, which is not observed. $X_{i,t}$ is a set of demographic and economic characteristics of county i at time t. We use $\bar{X}_i$ to denote the within county average of $X_{i,t}$ and $\bar{X}_t$ to denote the within time period average of $X_{i,t}$. $X_{i,0}$ and $Y_{i,0}$ denote initial conditions in county j. We use $Z_{i,t}$ to denote the set of observed control variables $\{X_{i,t}, \bar{X}_i, \bar{X}_t, X_{i,0}, Y_{i,0}, t\}$. The sample covers 195 large United States counties between the years 1980 through 1999, giving us 3900 observations.[4]

The intent here is that parameter β is an approximation of the causal effect of gun ownership $G_{i,t}$ on homicide rates $Y_{i,t}$, controlling for county-level demographic and economic characteristics. We provide further detail about the use of proxy treatments in Section 10.A. To attempt to flexibly account for fixed heterogeneity across counties, common time factors, and deterministic time trends, we include county-level averages, time period averages, initial conditions, and the time index as additional control variables. This strategy is related to strategies for addressing latent sources of heterogeneity via conditioning as in [4]. Finally, for simplicity in this illustration, we assume that all sources of dependence are accounted for by observed variables such that we may take $\epsilon_{i,t}$ as independent across counties, j, and over time, t.

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

As a summary statistic we first look at a simple regression of $Y_{i,t}$ on $D_{i,t-1}$ without controls. The point estimate is 0.302 with 95% confidence interval based on the assumption that $\epsilon_{i,t}$ is independent over time and space ranging from 0.277 to 0.333. These results suggest that increases in gun ownership

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

4: Python Notebook on DML for Impact of Gun Ownership on Homicide Rates and R Notebook on DML for Impact of Gun Ownership on Homicide Rates

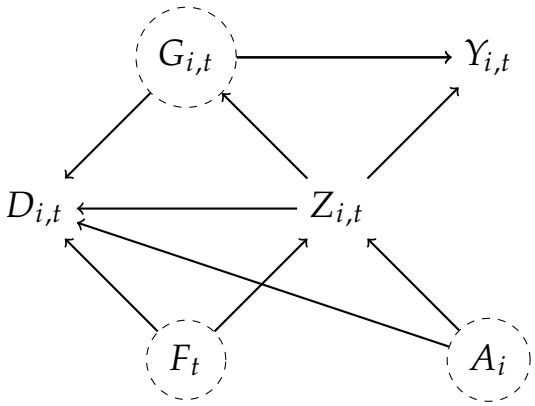


Figure 10.4: A Possible DAG Structure for the Gun Ownership Example. Here we approximate the average causal effect $G_{i,t} \rightarrow Y_{i,t}$ only if $G_{i,t} \approx D_{i,t}$. Under the assumption that $D_{i,t}$ is equal to $G_{i,t}$ plus an additive, independent measurement error, the target parameter β will be attenuated relative to the true causal effect; see Section 10.A. We also include nodes for latent county specific and time period specific shocks. Often such shocks are accounted for with so-called "fixed effects" which typically leverage strong functional form assumptions. Here, we instead leverage the different, though still strong assumption that flexibly conditioning on observables, including time- and county- specific variables, is sufficient to account for all relevant sources of confounding.

rates are associated with (predict) gun homicide rates – if gun ownership increases by 1% the predicted gun homicide rate goes up by around 0.3% – without controlling for any time factors or county characteristics.

Since our goal is to estimate the effect of gun ownership after controlling for a rich set characteristics, we next include the controls and estimate the model by an array of the modern regression methods that we've learned. Specifically, we consider ten candidate learners for predicting the outcome and for predicting the target variable. We consider linear models estimated with OLS using no control variables (OLS - No Controls), using only the raw control variables (OLS - Basic), and using the raw control variables plus the constructed cross-sectional and time series averages and initial conditions (OLS - All). The remaining methods always take as inputs the complete set of candidate control variables. We use cross-validation to choose tuning parameters for Lasso, Ridge, and Elastic Net. We consider a random forest with default choices and boosted trees constrained to have depth four. Finally, we consider neural nets with two hidden layers of 16 nodes each with early stopping. See R Notebook on DML for Impact of Gun Ownership on Homicide Rates for other training details.

Before turning to estimation results for β, we look out estimated out-of-sample predictive performance in Table 10.1 which reports cross-fitted root mean square error (RMSE) for the different procedures we consider. The column RMSE Y gives the RMSE

	RMSE Y	RMSE D
OLS - No Controls	1.0964	1.2109
OLS - Basic	0.9540	0.4990
OLS - All	1550360633.3904	70320592.0157
Lasso (CV)	0.4617	0.1360
Ridge (CV)	0.5303	0.1450
Elastic Net (.5,CV)	0.4654	0.1346
Random Forest	0.4021	0.1253
Boosted trees - depth 4	0.4021	0.1224
DNN dropout	0.6659	0.8214
DNN early stopping	0.5171	0.1802

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

.

for predicting the outcome (log gun homicide rate), and the column RMSE D gives the RMSE for predicting our gun prevalence variable (log of the lagged firearm suicide rate). Here we evidence of the potential relevance of trying several learners rather than just relying on a single, pre-specified choice. There are noticeable differences between performance of most of the learners, with Boosted Trees and Random Forests providing the best performance for predicting both the outcome and policy variable.

Table 10.2 presents the estimated effects of the lagged gun ownership rate on the gun homicide rate as well as the corresponding standard errors. Looking across the results, we see relatively large differences in estimates. These differences suggest that the choice of learner has a material impact in this example. Looking at the measures of predictive performance in Table 10.1, we see that Random Forest and Boosted trees performed best among the considered learners, and we also see that their performance is relatively similar in terms of point estimates of the effect of the lagged gun ownership rate on the gun homicide rate and standard errors. Focusing on the Boosted trees row, the point estimate suggests a 1% increase in the gun proxy is associated with around .1% increase in the gun homicide rate, though the 95% confidence interval is relatively wide: (-0.012,0.211).

The last two rows of the table provide estimates based on using the cross-fit estimates of predictive accuracy of the considered procedures (provided in Table 10.1. The row "Best" uses the method with the lowest MSE as the estimator for $\hat{\ell}(X)$ and $\hat{m}(X)$. In this example, Boosted trees give the best performances in predicting both $Y_{i,t}$ and $D_{i,t-1}$, so the results for "Best" and Boosted trees are identical. The row "Ensemble" uses the linear combination of all ten of the predictors the produces the lowest

	Estimate	Standard Error
OLS - No Controls	0.3020	0.0126
OLS - Basic	-0.2568	0.0963
OLS - All	-4.1832	0.8028
Lasso (CV)	0.2856	0.0568
Ridge (CV)	0.4624	0.0600
Elastic Net (.5,CV)	0.2888	0.0580
Random Forest	0.0363	0.0532
Boosted trees - depth 4	0.0997	0.0568
DNN dropout	0.2646	0.0110
DNN early stopping	0.5731	0.0496
Best	0.0997	0.0568
Ensemble	0.0864	0.0560

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

.

MSE for predicting $Y_{i,t}$ or $D_{i,t-1}$ as the estimator $\hat{\ell}(X)$ or $\hat{m}(X)$ respectively. Here the results are similar to the results using only Boosted trees, but differ somewhat due to non-zero linear combination coefficients on the other learners. We also note that the standard error for the ensemble is (slightly) smaller than that of "Best."

Revisiting the Price Elasticity for Toy Cars

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

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

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

In Table 10.3, we report the cross-validated R^2 for predicting D and Y with each of the learners along with the resulting DML

point esimate, standard error estimate, and 95% confidence interval. The first thing we note is that all confidence intervals indicate a substantial negative effect, with a clear indication not only of the direction of the effect but also of its overall magnitude.

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

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

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

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

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

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

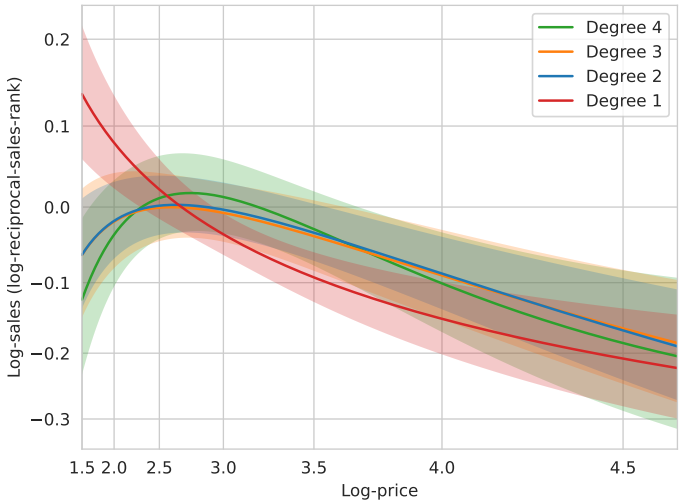


Figure 10.5: DML estimates of the price-sales relationship using PLM with higher-order transformations of price. Note the exponential scaling in the axes, which transforms the overall scale back to (non-log) price and sales (reciprocal sales rank).

We plot the resulting estimated functions for $r = 1, \ldots, 4$ in Figure 10.5. As can be seen, the price-sales relationship seems to not be exactly linear, as it stabilizes around a flat-then-decreasing shape for degrees 2, 3, and 4. This shape either suggests that indeed there is less elasticity at lower price points (the mean log-price is 3.06) or that we simply failed to account well for confounding effects at lower price points, which may be idiosyncratic compared to higher-priced toy trucks.

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

10.3 DML Inference in the Interactive Regression Model (IRM)

DML Inference on APEs and ATEs

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

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

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

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

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

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

This quantity is the average predictive effect of switching $D = 0$ to $D = 1$. Under conditional exogeneity discussed in Chapter 5 and Chapter 6, the APE coincides with the average treatment effect (ATE) of the intervention that moves $D = 0$ to $D = 1$.

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

Our construction of the efficient estimator for ATE will be based upon the relation[5]

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

where

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

and

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

is the Horvitz-Thompson transformation.

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

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

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

and the propensity score reweighting representation,

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

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

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

The construction provided in (10.3.1) is equally applicable in cases where the propensity score $\mathrm{P}(D = 1 \mid X)$ is known, as

in stratified randomized experiments, and in cases where the propensity score is unknown. When the propensity score is known, the role of regression adjustment in (10.3.1) is to reduce estimation noise.

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

DML for APEs/ATEs in IRM

1. Partition sample indices into random folds of approximately equal size: $\{1, ..., n\} = \cup_{k=1}^{K} I_k$. For each $k = 1, ..., K$, compute estimators $\hat{g}_{[k]}$ and $\hat{m}_{[k]}$ of the conditional expectation functions g_0 and m_0, leaving out the k-th block of data, such that $\epsilon \leq \hat{m}_{[k]} \leq 1 - \epsilon$, and for each $i \in I_k$ compute

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

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

2. Compute the estimator

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

3. Construct standard errors via

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

   and use standard normal critical values for inference.

Remark 10.3.2 (Trimming) An important practical issue is trimming $|\hat{H}_i|$ from taking explosively large values. Large values can occur when estimated propensity scores are near 0 or 1, which may indicate failure of the overlap condition – Assumption 5.2.2 in Chapter 5 and restated in Theorem 10.3.1 below. In the algorithm above, $\hat{H}_i$ can take on the largest absolute value of $\bar{H} = 1/\epsilon$. Therefore, setting $\epsilon = .01$ corresponds to $\bar{H} = 100$. There does not seem to be a good theoretical or practical resolution on how to do trimming.

Theorem 10.3.1 (Adaptive Inference on ATE with DML) *Suppose conditions specified in [2] hold. In particular, suppose that the*

overlap condition holds, namely for some $\epsilon > 0$ with probability 1

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

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

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

then the estimation error in these nuisance parameter has no first order effect on $\hat{\theta}$:

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

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

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

where

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

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

DML Inference for GATEs and ATETs

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

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

where G is a group indicator. For example, we might be interested in the impact of a vaccine on teenagers, in which case we could set $G = 1(13 \leq \text{Age} \leq 19)$, or on older individuals, in which case we might set $G = 1(65 \leq \text{Age})$. DML estimation and inference for GATEs can be carried out similarly to estimation and inference for the ATE by exploiting the relation

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

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

Another common target parameter ATET:

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

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

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

Remark 10.3.3 (Misspecification of PLM as inference on an overlap-weighted APE) In the case of binary treatment $D \in \{0, 1\}$, the IRM (Eqs. 10.3.1 and 10.3.2) generalizes the PLM of Section 10.2 (Eq. 10.2.1) by permitting interaction between the treatment and controls. The PLM, nonetheless, admits a very simple estimator for the treatment coefficient via partialling out: simply regress cross-fitted outcome residuals on cross-fitted treatment residuals, never dividing by propensity scores. What does this get at, however, when the PLM fails to hold? Per Remark 10.2.2, we need only consider the BLP of $\tilde{Y}$ in terms of $\tilde{D}$ in the more general IRM. Writing $g_0(D, X) = g_0(0, X) + D(g_0(0, X) - g_0(1, X))$, we see that $\tilde{Y} = \tilde{D}(g_0(1, X) - g_0(0, X)) + \epsilon$. Since $\mathrm{E}[\tilde{D}^2 \mid X] = m_0(X)(1 - m_0(X))$, we find that the estimand is $\beta = \mathrm{E}[m_0(X)(1 - m_0(X))(g_0(1, X) - g_0(0, X))]/\mathrm{E}[m_0(X)(1 - m_0(X))]$, that is, the APE on the population reweighted by $m_0(X)(1 - m_0(X))/\mathrm{E}[m_0(X)(1 - m_0(X))]$, known as overlap weights as they upweight when $m_0(X)$ is close to $1/2$ and downweight when $m_0(X)$ is close to 0 or 1.

In the case of a continuous univariate treatment on $[0, 1]$, we can leverage the same idea of writing $g_0(D, X)$ as a baseline plus the effect of D using the fundamental theorem of calculus: $g_0(D, X) = g_0(0, X) + \int_0^1 \mathbb{I}[D > t] g_0'(t, X) dt$, where g_0' is the derivative in the first argument. We can then find that β identifies some average derivative $\mathrm{E}[w(D, X) g_0'(D, X)]/\mathrm{E}[w(D, X)]$ for some nonnegative weights $w(d, x) = \mathrm{E}[\tilde{D}\mathbb{I}[D > d] \mid X = x]/f(d \mid x) \geq 0$, where $f(d \mid x)$ is the conditional density of

D given $X = x$ (see, e.g., Sec. 2.3.1 of [5]). That is, we estimate *some* average causal effect of increasing every value of D by an infinitesimal amount. However, the population over which we average may be highly uninterpertable.

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

R Notebook on DML for Impact of 401(K) Eligibility on Financial Wealth

Python Notebook on DML for Impact of 401(K) Eligibility on Financial Wealth

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

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

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

Compare this argument to the one given below using DAGs.

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

In what follows, we use net financial assets[6] as the outcome variable, Y, in the analysis. The treatment variable, D, is an indicator for being eligible to enroll in a 401(k) plan. The vector

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

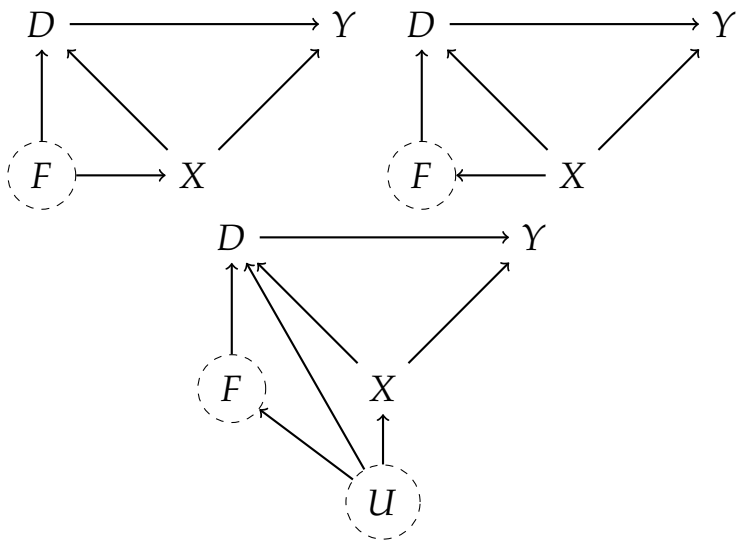


Figure 10.6: Three Causal DAGs for analysis of the 401(K) example in which adjusting for X is a valid identification strategy. The bottom figure encompasses the other two as special cases.

of raw covariates, X, consists of age, gender, income, family size, years of education, a married indicator, a two-earner status indicator, a defined benefit pension status indicator, an IRA participation indicator, and a home ownership indicator.

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

R Notebook on Dagitty-Based Identification in 401(K) Example

It is also useful to consider structures that would break down the identification strategy. We illustrate two such structures in Figures 10.7 and 10.8. In these figures, we introduce a node for the employer match amount, M,[7] which could mediate the effect of 401(k) eligibility and have an important effect on financial wealth.

7: Employers often offer a benefit where they will match a proportion of an employee's contribution to their 401k, up to a limit. The limit is referred to as the employer match amount, and averages between 4 and 5% of employee's salaries.

In Figure 10.7, we suppose that M is determined by unobserved firm characteristics, F, and worker characteristics, X. In this case, adjustment for X is not sufficient as there is a path from latent firm characteristics, which are related to the treatment, to the outcome that is not closed by X. However, if M is determined solely by D and X so the red arrow is erased, adjustment for X is sufficient. Therefore, interpreting the target parameter of our

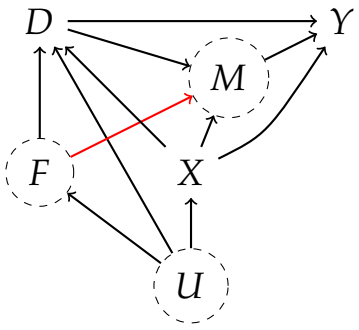


Figure 10.7: A DAG Structure where adjusting for X is not sufficient. If there is no arrow from F to M, adjusting for X is sufficient.

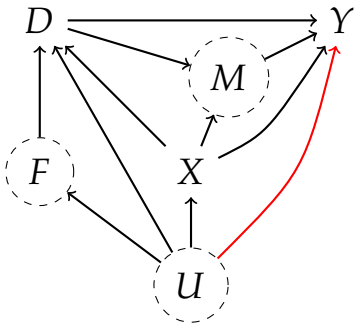


Figure 10.8: Another DAG Structure where adjusting for X is not sufficient. Here the latent confounder U affects all variables, so even in the absence of an arrow connecting F to M, causal effects cannot be determined after adjusting for X. The presence of such latent confounders is always a threat to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

estimation strategy as a causal effect is only valid if the match amount is independent of F given D and X, that is, if there is no arrow from F to M in the graph. Otherwise, the default interpretation is that we are estimating predictive effects of 401(k) eligibility.

In the second example, Figure 10.8, we maintain the assumption that M is independent of F given D and X by eliminating the arrow between nodes F and M. However, we now allow for the possibility that latent variables U have a direct effect on Y; that is, we have an unobserved confounder or omitted variable. In this example, such a counfounder may be unobserved risk preferences that relate to an individual's preference over jobs, an individual's characteristics, but also have direct effects on savings decisions not channeled purely through observed individual or job characteristics. In general, the possibility of latent confounders always poses a challenge to obtaining estimates of causal effects in non-experimental data. The presence or absence of latent confounders cannot be determined solely from the data in general, and thus their presence must be argued against based on scientific and institutional knowledge in different contexts. See, e.g., discussion in the original papers, [6] and [7], underlying this example. As in the previous example, we must interpret our estimates as predictive effects of 401(k) eligibility if we believe the connection from U to Y exists.

In Table 10.4, we report DML estimates of ATE of 401(k) eligibility on net financial assets both in the partially linear model and

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

	Lasso	Forest	Boost	NNet	Ens	Best
A. Interactive Regression Model						
ATE	7993	8105	7713	7788	7839	7753
	[1201]	[1242]	[1155]	[1238]	[1134]	[1237]
B. Partially Linear Regression Model						
ATE	8871	9247	9110	9038	9166	9215
	[1298]	[1295]	[1314]	[1322]	[1299]	[1294]

Note: Estimated ATE and standard errors from a partially linear model (Panel B) and heterogeneous effect model (Panel A) based on orthogonal estimating equations. Column labels denote the method used to estimate nuisance functions.

the interactive regression model allowing for heterogeneous treatment effects. To reduce the disproportionate impact of extreme propensity score weights in the interactive model, we trim the propensity scores at 0.01 and 0.99.

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

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

Finally, we can conclude the discussion with a more sobering note that there are credible deviations in the graph structure (e.g. unobserved firm characteristics may affect the match amount)

that challenges causal interpretation of the estimates. One approach to dealing with such deviations would be to conduct thorough sensitivity analysis.*

10.4 Generic Debiased (or Double) Machine Learning

Key Ingredients

A general construction upon which DML estimation and inference can be built relies on a method-of-moments estimator for some low-dimensional target parameter θ_0 based upon the empirical analog of the moment condition

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

where we call ψ the score function, W denotes a data vector, θ_0 denotes the true value of a low-dimensional parameter of interest, and η denotes nuisance parameters with true value η_0.

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

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

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

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

and the Neyman orthogonality condition is satisfied:

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

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

* We have done some informal simulations to assess the impact of this threat (using the observation that firms match up to 5% of income), and we estimated the size of the bias to be in the ball park of 10%. Given this, we believe the results reported here are reasonable approximations to the causal effects.

Remark 10.4.1 The orthogonality condition is named after Neyman [8], because he was the first to propose it in the context of parametric models with nuisance parameters that are estimated at slower than $1/\sqrt{n}$ rates.

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

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

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

The statement

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

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

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

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

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

1) approximate sparsity for η_0 with respect to some dictionary calls for the use of Lasso, post-Lasso, or other sparsity-based techniques;
2) well-approximability of η_0 by trees calls for the use of regression trees and random forests;

3) well-approximability of η_0 by sparse deep neural nets calls for the use of ℓ_1-penalized deep neural networks;
4) well-approximability of η_0 by at least one model mentioned in 1)-3) above calls for the use of an ensemble/best choice method over the estimation methods mentioned in 1)-3).

There are performance guarantees for most of these ML methods that make it possible to satisfy the conditions stated above. Ensemble and best choice methods ensure that the performance guarantee is no worse than the performance of the best method.

The third key input is to use a form of sample splitting at the stage of producing the estimator of the main parameter θ_0, which allows us to avoid *biases* arising from overfitting.

Overfitting can easily occur when using highly complex fitting methods such as boosting, random forests, deep nets, ensembles, and other hybrid machine learning methods. We may heuristically think of overfitting as capturing noise that is particular to the observations used to fit a model in addition to signal. Using overfit estimates of nuisance parameters obtained using the same data as used to estimate the target parameter then heuristically leads to estimation error in these parameters being correlated to outcomes which introduces a type of bias. This bias can be very large, as illustrated in Figure 10.2. We specifically use cross-fitted forms, i.e. sample splitting, of the empirical moments, as detailed below, in estimation of θ_0 to avoid this problem.

Neyman Orthogonal Scores for Regression Problems

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

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

where $W = (Y, D, X)$ is a data vector, and η is the nuisance parameter $\eta = (\ell, m)$ with true value $\eta_0 = (\ell_0, m_0)$. Here, ℓ and

m are square-integrable functions mapping the support of X to $\mathbb{R}$ whose true values are given by

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

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

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

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

where

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

$W = (Y, D, X)$ is a data vector, and $\eta := (g, m)$ is the nuisance parameter with true value $\eta_0 = (g_0, m_0)$. Here, g is a square-integrable function mapping the support of (D, X) to $\mathbb{R}$, and m is a function mapping the support of X to $(\varepsilon, 1-\varepsilon)$ for some $\varepsilon \in (0, 1/2)$. The true values of g and m are given by

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

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

For estimation of GATEs we use the score

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

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

For estimation of the ATET parameter, we use the score

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

where $H(D, X)$ is given in (10.4.5), and $\eta = (g, m, p)$ is the nuisance parameter with the true value $\eta_0 = (g_0, m_0, p_0)$ for g_0 and m_0 defined in (10.4.6) and $p_0 = \mathrm{P}(D = 1)$. Note that this

score does not require estimating $g_0(1, X)$.

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

The DML Inference Method

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

Generic DML

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

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

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

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

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

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

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

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

where

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

and

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

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

$$[\ell'\hat{\theta} \pm c\sqrt{\ell'\hat{\mathrm{V}}\ell/n}],$$

where c is the $(1-\alpha/2)$ quantile of $N(0,1)$.

7. **Outputs**: Output the results of all steps.

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

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

In such cases the estimator takes the form

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

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

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

Remark 10.4.5 (Choosing the number of folds) The choice $K \geq 4-5$ works well based on a variety of empirical examples and in simulations for medium-sized data sets. The choice $K \geq 10$ works well for small data sets.

Properties of the general DML estimator

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

Definition 10.4.1 (Strong Identification) *We have that* $\mathrm{M}(\theta, \eta_0) =$

0 *if and and only if* $\theta = \theta_0$, *and that*

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

has singular values that is bounded away from zero.

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

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

Then, estimation of nuisance parameter does not affect the behavior of the estimator to the first order; namely,

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

where

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

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

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

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

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

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

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

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

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

excluding the k-th fold. We have that

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

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

- For each method j, compute the cross-fitted MSPEs
$$\mathbb{E}_n[\check{V}_{mj}^2].$$
- Select the best ML method for predicting V_m via
$$\hat{j}_m = \arg\min_j \mathbb{E}_n[\check{V}_{mj}^2].$$
- Use the method $\hat{j}_m$ as a learner of η_m in the Generic DML Algorithm.

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

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

Notebooks

- R Notebook on DML for Impact of Gun Ownership on Homicide Rates and Python Notebook on DML for Impact of Gun Ownership on Homicide Rates provide anapplication of DML inference to learn predictive/causal effects of gun ownership on homicide rates across U.S. counties.
- R Notebook on Dagitty-Based Identification in 401(K) Example and Python Notebook on Pgmpy-Based Identification in 401(K) Example analyze graph structures that enable identification of the causal effect of 401(K) eligibility on net financial wealth.
- R Notebook on DML for Impact of 401(K) Eligibility on Financial Wealth and Python Notebook on DML for Impact of 401(K) Eligibility on Financial Wealth provide application of DML inference to learn predictive/causal effects of 401(K) eligibility on net financial wealth. (Note:

The results produced in this notebook and provided in the text are slightly different than those in the original paper [2]. The replication files for [2] are given at the following Github repository. The difference is due to our use of a single split of the sample in producing the results for this text while the results in [2] are based on a method that aggregates results across multiple data splits.)

- R Notebook on DML for Growth Regression Analysis and Python Notebook on DML for Growth Regression Analysis provide an application of DML inference based on ML on predictive/causal effects of countries' initial wealth on the rate of economic growth.

Notes

For a detailed literature review and technical regularity conditions needed for each of theorems, see [2], which also gives an overview of various analytical methods for generating Neyman-orthogonal scores in a wide variety of problems.

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

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

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

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

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

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

debiasing correction proposed by Neyman; see, e.g. [14] for a review.

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

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

Targeted maximum likelihood estimation (TMLE) is another general approach for building orthogonal estimators [15]. This approach relies on doing maximum likelihood estimation for a target parameter, using a least favorable parametric submodel for the parameter of interest as the likelihood function. As with DML, TMLE needs to be combined with cross-fitting in order to deal with general ML estimators to avoid overfitting. The DML and cross-fitted TMLE should generally produce first order equivalent answers under correct specification. However, using TMLE can refine the finite-sample properties.

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

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

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

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

Study Problems

1. Experiment with one of the notebooks for the partially linear models (Guns example, Guns with DNNs, or Growth example). For example,

(a) Apply the methods to a different empirical example (e.g., Penn reemployment experiment from CI-1),
(b) or, using the same empirical example, try to use the H20 Auto ML framework as the machine learning tool to estimate m and ℓ functions. (See Chapter 9 H20 Auto ML to get started).

Explain what you are doing to a fellow student.

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

3. Study the 401(K) empirical analysis notebook (the part that does not deal with instrumental variables and LATE). Extend it to another empirical example of your choice (Penn reemployment experiment from Chapter 1, for example) or estimate ATE for 401(K) eligibility for a subset of low income (or high-income) workers (Group ATEs).

4. (Theoretical). Explain to a friend the concept of Neyman orthogonality, illustrating it with one of the examples in Appendix B. Extend the calculations in Appendix B to verify Neyman orthogonality for the ATET score specified in (10.4.8).

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

10.A Bias Bounds with Proxy Treatments

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

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

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

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

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

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

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

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

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

If we somehow knew that

$$R^2_{\tilde{D}\sim\tilde{G}} := \mathrm{E}[\epsilon_G^2]/(\mathrm{E}[\epsilon_G^2] + \mathrm{E}[\epsilon_D^2]) \geq 2/3$$

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

$$\beta \leq \alpha \leq \beta/R^2_{D\sim G} = (3/2)\beta.$$

10.B Illustrative Neyman Orthogonality Calcuations

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

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

by definition of β_0 of η_0; recall the 0 indices denote true values. Let $U = (Y - \ell_0(X)) - (D - m_0(X))\beta_0)$. Then, for any $\eta = (m, \ell)$ that are square integrable, the Gateaux derivative in the direction

$$\Delta = \eta - \eta_0 = (m - m_0, \ell - \ell_0)$$

is given by

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

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

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

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

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

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

is given by

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

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

Bibliography

[1] Jerzy Neyman. '$C(\alpha)$ tests and their use'. In: *Sankhyā: The Indian Journal of Statistics, Series A* (1979), pp. 1–21 (cited on page 247).

[2] Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. 'Double/debiased machine learning for treatment and structural parameters'. In: *Econometrics Journal* 21.1 (2018), pp. C1–C68 (cited on pages 251, 264, 277, 279, 280).

[3] Philip J. Cook and Jens Ludwig. 'The social costs of gun ownership'. In: *Journal of Public Economics* 90 (2006), pp. 379–391 (cited on page 257).

[4] Jeffrey M. Wooldridge. 'Two-Way Fixed Effects, the Two-Way Mundlak Regression, and Difference-in-Differences Estimators'. In: *Available at SSRN: https://ssrn.com/abstract=3906345 or http://dx.doi.org/10.2139/ssrn.3906345* (2021) (cited on page 257).

[5] Joshua D. Angrist and Alan B. Krueger. 'Empirical Strategies in Labor Economics'. In: *Handbook of Labor Economics. Volume 3*. Ed. by O. Ashenfelter and D. Card. Elsevier: North-Holland, 1999 (cited on page 267).

[6] James M. Poterba, Steven F. Venti, and David A. Wise. '401(k) Plans and Tax-Deferred savings'. In: *Studies in the Economics of Aging*. Ed. by D. A. Wise. Chicago, IL: University of Chicago Press, 1994, pp. 105–142 (cited on pages 267, 269, 270).

[7] James M. Poterba, Steven F. Venti, and David A. Wise. 'Do 401(k) Contributions Crowd Out Other Personal Saving?' In: *Journal of Public Economics* 58.1 (1995), pp. 1–32 (cited on pages 267, 269, 270).

[8] Jerzy Neyman. 'Optimal asymptotic tests of composite hypotheses'. In: *Probability and Statsitics* (1959), pp. 213–234 (cited on pages 272, 279).

[9] Lester Mackey, Vasilis Syrgkanis, and Ilias Zadik. 'Orthogonal machine learning: Power and limitations'. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 3375–3383 (cited on page 279).

[10] Peter J. Bickel, Chris A.J. Klaassen, Ya'acov Ritov, and Jon A. Wellner. *Efficient and adaptive estimation for semiparametric models*. Johns Hopkins University Press Baltimore, 1993 (cited on page 279).

[11] Whitney K. Newey. 'The asymptotic variance of semiparametric estimators'. In: *Econometrica* 62.6 (1994), pp. 1349–1382 (cited on pages 279, 280).

[12] Peter M. Robinson. 'Root-N-consistent semiparametric regression'. In: *Econometrica* 56.4 (1988), pp. 931–954. DOI: 10.2307/1912705 (cited on pages 279, 280).

[13] James M. Robins and Andrea Rotnitzky. 'Semiparametric efficiency in multivariate regression models with missing data'. In: *J. Amer. Statist. Assoc.* 90.429 (1995), pp. 122–129 (cited on pages 279, 280).

[14] Victor Chernozhukov, Christian Hansen, and Martin Spindler. 'Valid post-selection and post-regularization inference: An elementary, general approach'. In: *Annual Review of Economics* 7.1 (2015), pp. 649–688 (cited on page 280).

[15] Mark J. van der Laan and Sherri Rose. *Targeted learning: causal inference for observational and experimental data*. Springer Science & Business Media, 2011 (cited on page 280).

[16] Daniel O. Scharfstein, Andrea Rotnitzky, and James M. Robins. 'Adjusting for Nonignorable Drop-Out Using Semiparametric Nonresponse Models'. In: *Journal of the American Statistical Association* 94.448 (1999), pp. 1096–1120. (Visited on 01/25/2023) (cited on page 280).

[17] Philipp Bach, Oliver Schacht, Victor Chernozhukov, Sven Klaassen, and Martin Spindler. *Hyperparameter Tuning for Causal Inference with Double Machine Learning: A Simulation Study*. 2024 (cited on page 280).

Feature Engineering for Causal and Predictive Inference

11

"It's all about paying attention. [...] Attention is vitality. It connects you with others."

– Susan Sontag [1].

Here we discuss feature engineering as an approach to transform complex objects such as text and images into a collection of relatively low-dimensional numerical features (embeddings) that can be used for standard predictive or causal applications, for example as regressors in a prediction problem. We consider principal components, variational autoencoders and neural networks as general approaches to generate embeddings. We then consider text embeddings in detail, introducing two popular neural network-based Natural Language Processing (NLP) algorithms: ELMo and BERT. We finally consider image embeddings, applying a hedonic price model to apparel data using a neural network algorithm (ResNet50) to generate embeddings.

11.1 Introduction

Thus far, we have imposed a significant restriction on the kinds of data on which we can perform inference. While empiricists often consider simple datasets that include variables that have a numeric representation (binary, factor and continuous variables), researchers are increasingly confronted with complex forms of data, such as images and text, that encode a vast amount of information. In this section, we generalize our approach to allow using these types of data.

As a motivating example, we consider the problem of predicting prices of products using the types of characteristics that one might find on a webpage, namely the text in the product description and the product's image. The resulting predicted prices are called hedonic prices, and predictive modeling of this form is motivated by the hedonic price models of economics.

In order to predict prices, we have to convert text and images into relatively low-dimensional numerical features, called "embeddings." The minimal requirement on embeddings is that similar products should have similar embeddings. This requirement guarantees that price predictions for similar products are also similar. The maximal requirement on embeddings is that they should parsimoniously approximate maximal information from text and images that is relevant for price predictions.

The main methods for generating successful embeddings include the following, in order of increasing generality:

- classical principal component analysis,
- auto-encoders, and
- neural networks solving auxiliary prediction tasks.

The auxiliary tasks in the final method may include solving image processing problems, such as object classification and image compression, or natural language processing problems, such as summarization and machine translation.

These auxiliary tasks are not the same as the "main" task. In our price prediction example, the main task is predicting product prices. Before turning to the primary price prediction task, we consider ResNet-50, which is a Residual Network of depth 50, which is designed to perform well on various tasks of object type classification. Consequently, application of ResNet-50 produces embeddings that are useful inputs for solving this auxiliary object classification task. However, because product type is an important determinant of price, the embeddings produced by

ResNet-50 that help classify products can also serve as useful inputs to the main task – price prediction.

Analogously, a neural network such as BERT is trained on auxiliary tasks aimed to make it learn word similarity and contextual meaning of words. Consequently, BERT can produce embeddings that provide a useful numerical summary of a product's text description. Because the product description is an important determinant of the price, these embeddings can also serve as useful inputs to the price prediction task.

Embeddings are useful in a variety of predictive and causal inference problems. For example, we can imagine using

- embeddings of product images and descriptions for modeling variety and demand for products;
- embeddings of text resumes for studying the wage offer structure;
- embeddings of countries' characteristics for studying the effect of institutions;
- and please list many of your own here (homework).

There is an emerging literature on the use of embeddings for causal inference; see this repository of papers about using text data in causal inference. See also [2] for a recent review article on the importance and subtleties of using text as data in the social sciences.

11.2 From Principal Components to Autoencoders

Principal components are probably the earliest classical example of embeddings. One way to frame principal components is that principal components find unit length orthogonal linear combinations, directions, of a collection of variables that are "best" at reproducing the underlying data. The idea is then that a small number of principal components should capture most of the variability in the original variables and thus may provide a useful low-dimensional summary of the original data.

Specifically, let $(W_1, ..., W_n)$ be a sample of n observations of a high-dimensional centered random vector W_i in $\mathbb{R}^d$,[1] and let $\Sigma_n = \mathbb{E}_n[WW'] \in \mathbb{R}^{d \times d}$ denote the empirical covariance matrix. In order to reduce the dimension of W_i, suppose we wish to find $K \ll d$ mutually orthogonal rotations

$$X_{ki} := c_k' W_i, \quad k = 1, ..., K,$$

1: Thus, $\mathbb{E}_n[W_j] = 0$ for $j = 1, ..., d$.

of the original W_i's where

$$c_\ell' c_k = 0 \text{ for } \ell \neq k \text{ and } c_k' c_k = 1 \text{ for each } k$$

such that linear combinations of these variables approximate the original data. These rotations are called principal components of W_i. In applications, W_i represent high-dimensional raw features (images, for example), and the principal components

$$X_i^K = (X_{i1}, ...X_{iK})'$$

represent a lower-dimensional encoding or embedding of W_i.

More formally, we wish to solve

$$\min_{\{a_j\}_{j=1}^d, \{c_k\}_{k=1}^K} \sum_{j=1}^{d} \sum_{i=1}^{n} (W_{ji} - \hat{W}_{ji})^2$$

subject to

$$\begin{aligned} \hat{W}_{ji} := a_j' X_i^K \text{ for } X_i^K = (X_{1i}, ...X_{Ki})' \; j = 1, ..., d \text{ and } i = 1, ..., n, \\ X_{ki} = c_k' W_i \text{ for } i = 1, ..., n \text{ and } k = 1, ..., K, \\ c_k' c_k = 1 \text{ for } \quad k = 1, \ldots, K \\ c_k' c_\ell = 0 \text{ for } \ell \neq k \leq K. \end{aligned}$$

The constructed variables resulting from solving this problem,

$$X_i^K = (X_{1i}, ...X_{Ki})'$$

are the first K principal components.

Remark 11.2.1 The analytical solution to the principal components problem is as follows: The optimal $C_K = [c_1, ..., c_K]$ are the eigenvectors of $\Sigma_n = \mathbb{E}_n[WW']$ corresponding to the K largest eigenvalues $\lambda_1, ..., \lambda_K$ of Σ_n. That is, $\Sigma_n c_k = \lambda_k c_k$ for each k. Furthermore, the optimal a_j is the j-th column of C_K'.

Another interesting feature is of principal components is that they satisfy

$$\mathbb{E}_n[X_k^2] = \lambda_k$$

for $k = 1, ..., K$ and

$$\mathbb{E}_n[X_k X_\ell] = 0$$

for $\ell \neq k$. These properties result from the fact that the c_k are eigenvectors of Σ_n.

Finding principal components offers one way to produce encod-

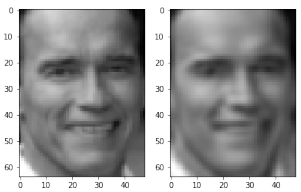


Figure 11.1: Featurizing a talented man: The original 3072-dimensional image W and image $\hat{W}$ produced from a 256-dimensional principal component embedding. As a by-product, we've just made an important causal discovery that, surprisingly, doing embedding causes one to be younger ;).