PREFACE

This book is for anyone who has felt the frustration of knowing that correlation isn't causation—and not knowing what to do about it. We wrote it for applied data analysts, scientists, and students across academia, health care, business, and policy—in short, anyone who works with data and wants to draw credible causal inferences. We avoid dense mathematical proofs and abstract theorems (those books already exist) and instead emphasize practical computational tools you can use right away.

Our approach is hands-on and example-driven. We lay out an explicit workflow that guides the reader from causal questions to credible answers. We use the `R` programming language throughout, with code snippets woven into the narrative and more detailed implementations available on the companion website. We've kept formal notation to a minimum and instead focus on building intuitions. Each chapter builds on the previous, highlighting core objectives and pointing you to further reading for going deeper.

Among other things, along the way you'll learn to:

- Use causal graphs to visualize causal assumptions and distinguish good control variables from bad ones.
- Think counterfactually and sharpen your target research question.
- Understand and estimate marginal and conditional effects, including with g-methods such as inverse probability weighting and g-computation.
- Tackle missing data as the central challenge it often is, using post-stratification and imputation to connect your sample to the target population.
- Work with multilevel data to account for clustering and some forms of unobserved confounding.
- Use simulation-based inference to plan, stress-test, and interpret analyses, including making counterfactual predictions.

In short, this is a compact, entry-level, and hands-on introduction to causal inference in practice. Our aim is to help you express reasonable

causal questions, design sensible analyses that match those questions, and communicate assumptions and results clearly and responsibly. We also aim for this book to set you up for pursuing more specialized texts and topics in causal inference.

Causal inference is difficult, and we won't pretend otherwise. But it's also learnable, and the payoff—being able to answer causal questions that actually matter—is worth the effort. We hope this book becomes a helpful guide on that journey.

CHAPTER 1. INTRODUCTION

In a world where data-driven decisions shape our daily lives, the pursuit of causation is more vital than ever. Whether you're a scientist probing the mysteries of the natural or social world, a business strategist seeking the secret to market success, or a health care professional striving to improve patient outcomes, the quest to unravel cause-and-effect relationships is a ubiquitous goal.

By now, it's a truism that correlation is not causation (Figure 1.1). But how, then, do we establish causation between things if not by their statistical association alone? How do we know that smoking causes cancer? How can we be sure that vitamin C protects against scurvy? Does advertising actually influence consumer behavior? Does education increase income?

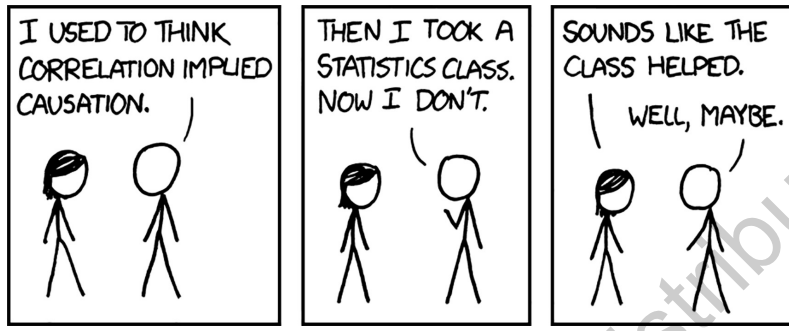
Welcome to the realm of applied causal inference. In the pages ahead, we will explore state-of-the-art concepts, techniques, and real-world applications of causal inference with both experimental and observational data. We will start with the basics (e.g., regression) and progressively build up to more advanced and general methods (e.g., g-methods, poststratification, imputation, and multilevel models). Along the way, we will show how to implement these techniques, as well as highlight their key strengths, limitations, and assumptions. We assume only a basic understanding of applied statistics, in particular linear and logistic regression—after all, this is the *data analyst's* guide to cause and effect, not the *theoretical statistician's*! Most importantly, we assume an interest in the extent to which we can ever know if one thing caused something else.

It's fair to ask if the world needs another book on causal inference. While there are many excellent introductions to causal inference out there (for some references, see in particular the end of the final chapter), many are quite technical. Here we take a different approach. We will use minimal formal notation and instead emphasize a practical, computational workflow using the `R` programming language.[1] Compact code snippets are interwoven into the main text, while the companion website goes into more detail on the programming side of things.[2] Note that when we show computer output, we sometimes crop the output to the most relevant bits. At the end of each chapter, we provide chapter highlights—the essential insights and

[1]You can download `R` via www.cran.r-project.org. You might also want to install RStudio via www.posit.co/download/rstudio-desktop, which many people find helpful as a graphical interface to `R`.

[2]The companion website is hosted at www.theissbendixen.com/dag-book.

Figure 1.1 Correlation is not causation.



Source: www.xkcd.com.

skills that later chapters will refer back to and build upon—as well as further readings for diving deeper into central concepts.

In short, it is our aim that this short book will equip both the experienced data analyst and the novice with practical and transferable skills to navigate the intricate science of cause and effect—as well as encourage the reader to pursue more specialized topics on their own.

Outlining the Data Analyst's Guide to Cause and Effect

With the present chapter, we begin the book by emphasizing the importance of establishing causation in nearly all aspects of everyday life. We introduce readers to the fundamental problem and promise of causal inference, especially the need to infer counterfactuals that are generally unobservable. We also lay out an acronym—"EEESI"—that will guide our workflow in subsequent chapters.

Chapter 2 introduces causal graphs, particularly DAGs, as essential tools for visualizing and understanding causal relationships. We discuss how DAGs help in identifying "good and bad controls"—that is, covariates to adjust for or not—and provide an overview of how DAGs can be constructed to represent causal hypotheses.

In Chapters 3 and 4, we emphasize the key distinction between conditional (i.e., "average people") and marginal (i.e., "people on average") effects. To this end, we introduce g-methods and demonstrate their basic implementation, key assumptions, and some more advanced extensions and applications.

Chapters 5 and 6 address common issues of missing data in causal analysis. We discuss the implications of missing data for both external and

internal validity, outline a variety of treatment effects relating to different target populations of interest, and introduce techniques like poststratification and imputation to handle common forms of missing data.

In Chapter 7, we shift focus to a few advanced estimation techniques, including multilevel modeling. We'll see how multilevel modeling, fixed effects, and the "Mundlak" model can be used to handle more complex data structures, manage unobserved confounding, and generally allow for more nuanced causal inferences.

Finally, Chapter 8 summarizes key insights from the preceding chapters, scans wider horizons, and points to routes the interested reader might want to take next.

The Fundamental Promise of Causal Inference

Consider the following questions: *Does this medical treatment cure that disease? Does this marketing strategy boost sales? Does this new policy improve social welfare?*

These are questions about cause and effect. Such questions are all around us. They are often what matter most in life. When we intervene in the world—with, say, a medical treatment, a marketing campaign, or social reform—we want to know what happened as a result. But—and here's the crucial insight—we also need to know what would've happened *in the absence of the intervention*, counter to fact and all else being equal. That is, according to contemporary views of causality, the causal effect of an intervention is the difference between what actually happened and the **counterfactual** (Morgan & Winship, 2015; Pearl, 2009; Rubin, 1974; Woodward, 2005).

Simple enough, right? Well, in principle. But here immediately we run up against the **Fundamental Problem of Causal Inference** (Holland, 1986): *We can never observe the counterfactual!* For instance, we can never give a medical treatment to a patient and also *not* give the treatment to said patient. The counterfactual outcome is missing. That's why causal inference can also fundamentally be viewed as a missing data problem, a perspective to which we'll continually circle back.

So, the best we can often do is to try and infer the counterfactual, for instance, through study design or a statistical model. We can then contrast our counterfactual with the observed factual state of things. More precisely, the best we can often hope to do is to aim for what is called an **average treatment effect** or variants thereof. We'll learn more about various flavors of average effects in later chapters—as well as the key assumptions that allow for their causal interpretation. For now, we might say that an average effect of, for instance, a medical treatment is the difference in

counterfactual outcomes under receiving the treatment compared to not receiving the treatment for individuals on average.[3] Further, these individuals must be similar enough in relevant characteristics such that whatever the difference in outcomes between the treated and nontreated is, in fact, explained by the treatment alone and not by some other events or characteristics.

Let's make this concrete. Say we want to measure the effect of some intervention. For instance, this could be an intensive educational program for disadvantaged children. Imagine, too, that we measure the enrolled children's academic performance before and after the program and that we estimate an on-average increase. Is this estimate a valid average treatment effect of the program?

Unfortunately, things are rarely this straightforward. Many other things could account for the change. For one thing, the children grew older since the beginning of the program; maybe the better performance is simply explained by age. Or consider that, even in the absence of the program, some other educational initiative might have targeted these children, such that we cannot naively extrapolate the children's performance before the program into the future. Maybe you can think of other problems with simply taking a before-and-after measurement as a causal effect. The point is that we need a way to get at the unobserved counterfactual—that is, some way to compare *what actually happened* under intervention to *what would've happened* under no intervention, for instance, with a control group of children who were *not* enrolled in the program but are otherwise similar to the enrolled.

This, in a nutshell, is the Fundamental Problem of Causal Inference: The counterfactual is both essential and unobservable, so we have to infer it. This means that we have assumptions to make. But in return, we get something quite extraordinary: A rigorous, tried-and-tested workflow for thinking through and tackling cause-and-effect relationships.

The Fundamental Problem of Causal Inference, then, is also a promise. A promise that, if we are careful, honest, and transparent, we might actually get causal answers to the questions that matter most in life. Or, alternatively, we will be told that there exists no valid answer to the question we posed.

Either is a very valuable yield.

---

[3]We can compare the average treatment effect to the **INDIVIDUAL TREATMENT EFFECT**, that is, the effect of an intervention for a specific individual. This kind of effect is generally out of reach exactly because we cannot observe the same individual under both a treatment and a control scenario. But that has not stopped researchers from making some progress; see, e.g., Mueller and Pearl (2023).

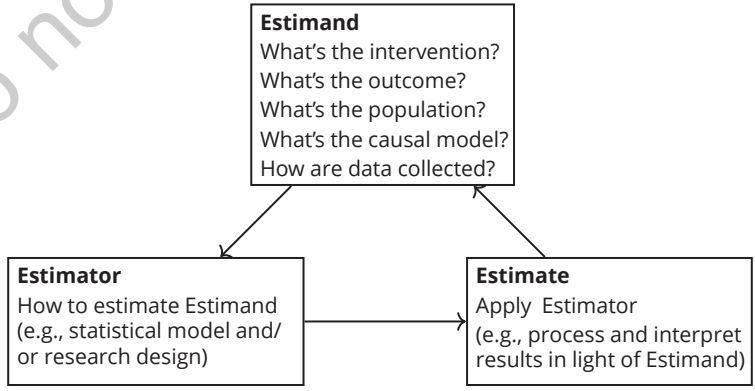
Causal Inference Is "EEESI"

That "tried-and-tested workflow" we mentioned above? Along the way, we'll familiarize ourselves with this workflow under the acronym "EEESI." EEESI stands for Estimand, Estimator, Estimate, and Simulation-based Inference (Figure 1.2). Let's break it down:

The first step in the EEESI workflow is to define the **ESTIMAND**. An estimand specifies the causal effect we aim to measure and quantify. This, in turn, requires a clear definition of the intervention and its possible effect, as well as the population of interest. In our example, this would be the effect of our intensive educational program on the academic performance of disadvantaged children at certain ages in certain geographies. Alongside this, we outline the causal assumptions necessary for our analysis. This includes designing data collection, identifying the appropriate set of variables for measurement and statistical modeling, and planning to handle potential sources of bias such as missing data. Starting with the next chapter, we'll learn to think through and evaluate all such choices through causal reasoning and simple data simulation. In particular, we'll familiarize ourselves with a tool—**DIRECTED ACYCLIC GRAPHS** or DAGs—for sketching out and formalizing causal assumptions. As we'll see, a DAG is a good starting point for defining an estimand.

The **ESTIMATOR** stage involves constructing a statistical model or method that can estimate the estimand. There exist many, many estimators, and new ones are constantly being developed. They all rely on central assumptions that are more or less plausible depending on the data context at hand.

Figure 1.2 Causal inference is "EEESI." Every step is planned and evaluated through simulation.



So the key here is to select an estimator that is well aligned with the defined estimand and is capable of accurately capturing the causal effect of interest under realistic assumptions. Again, simple simulation goes a long way. However, in this book, we won't get too fancy with the estimators. Instead, we'll give a flavor of a variety of very common estimators, focus on a few general ones, and point to further readings on more specialized topics.

Finally, the **ESTIMATE** step is where we apply our estimator to data to obtain an actual estimate of the estimand. This involves processing the results from the estimator and interpreting them in the context of our analysis. As with previous steps, simulation plays a crucial role in evaluating these estimates.

Okay, so those were the three Es of the EEESI acronym. Starting with the next chapter, we'll get practical and illustrate some implementations. But what about **SIMULATION-BASED INFERENCE**? We've already hinted at one important role that simulation can play, namely, in planning and evaluating our estimand, estimator, and estimate. The key idea here is to simulate data that resemble our real-world dataset in important ways and then test our methods in this simplified, simulated world. If our planned analysis works here, we're more confident that it could work in the real world, too. Of course, there are no guarantees. If our simulation misses central features of the real data, we'll be led astray.

But simulation has other roles to play as well. For instance, in Chapter 3 and beyond, we'll work with estimators that require us to not only fit a statistical model but also simulate predictions from that model. In essence, we ask the model what would happen if we set certain variables to particular values of interest—for instance, what happens under exposure to an intervention versus no exposure. That is, simulation helps us get at counterfactual scenarios. Needless to say, those simulated counterfactuals are only as good as the model that we build and the data on which the model is trained, so sound scepticism is warranted and encouraged.

A little later, in Chapter 5, we'll also use simulation to think carefully about how our analysis connects to the population that we're interested in making inferences about. This is the ultimate missing data problem, because very often, we're left with only an imperfect sample of the target population. Finally, in a more general sense, some of the computational methods we'll work with throughout the book require computational simulation and mathematical approximation.

If all of this sounds intriguing yet overwhelming, rest assured! We'll slowly build it all up as we go with practical and reproducible examples. For now, all of this is just to say that the EEESI workflow offers us a general road map for estimating causal effects, ensuring that each step of the way is carefully defined and implemented. Apart from the silly acronym,

this general approach is not original to us (in particular, see McElreath, 2020). In fact, there's a burgeoning movement in many fields, including in the health (e.g., Kahan, Hindley, Edwards, Cro, & Morris, 2024) and social sciences (e.g., Lundberg, Johnson, & Stewart, 2021), toward greater emphasis on explicit estimands and how they relate to estimation and interpretation.

In a sense, however, what we're showing here is still only a glimpse of the full picture. Causal inference is not easy (Chapter 8), and there's of course immensely more to statistical modeling of causal effects than we can cover in this short volume. For that reason, we keep most statistical models fairly simple and instead emphasize the key aspects of a general causal data analysis workflow. Only in later chapters (in particular, Chapter 7) do we discuss in some detail a few advanced statistical modeling tricks. For the remaining pieces of the picture—fitting, interpreting, evaluating, and diagnosing statistical models of various sorts and complexities—some useful starting points include Alexander (2023); Gelman and Hill (2006); Gelman et al. (2013, 2020); Gelman, Hill, and Vehtari (2021); Johnson, Ott, and Dogucu (2022); Kruschke (2014); Lambert (2018); McElreath (2020); Roback and Legler (2021).

The R Programming Language

To take full advantage of the EEESI workflow, we'll need some statistical software. We've chosen `R` (R Core Team, 2024) because it's open source, easy to install, widely used in both industry and academia, and has a very active and responsive community of users and developers. For each novel statistical technique and concept we introduce, we'll show the most pedagogically effective implementation that we know of in `R`, while the companion website will show some more advanced options. The companion website also shows code to reproduce all plots. While some familiarity with `R` is therefore useful, throughout we explain the code bit by bit, particularly in earlier chapters. Since we mostly make use of base `R` functions rather than contributed software (i.e., libraries or packages) throughout the book, the key procedures are also relatively straightforward to translate to another programming language of choice. Here, we just briefly introduce some very basic `R` programming that foreshadows later use cases. For a comprehensive introduction to `R`, plenty of good books are available (e.g., Wickham, Çetinkaya-Rundel, & Grolemund, 2023).

To start, we'll often need to define an object for later use. The following code snippet shows how to do this. Here, we define `A` as `1` using the operator `<-`.

```
A <- 1
```

If you call `A` now, it returns `1`. You can even add 1 to `A`, and it will yield 2. Try typing and running:

```
A + 1
```

An object also does not necessarily hold only one value. We can define a *vector* of values to a single object. For instance, from the next chapter and on, we'll start working with distributions of values. We'll simulate values from a distribution (usually the normal distribution for simplicity) of a given length, *n*. Let's see what this could look like.

```
n <- 1e4
Z <- rnorm(n = n, mean = 0, sd = 1)
```

We first defined `n` as 10,000, using the scientific notation short-hand `1e4` (a 1 with four zeroes). This is the number of values that we want to simulate. The next line defines `Z` as a random, normally distributed variable—`rnorm()` is short-hand for "random normal"—with a length of `n`, a mean of 0, and a standard deviation of 1 (we're explicit about the arguments `n`, `mean`, and `sd` here, but they can be left out, and we'll do that from here on out). Now, `Z` holds not a single value but a vector of values that is (approximately) normally distributed. You could try and call `hist(Z)` to get a simple histogram of the simulated variable. You'll see that most values are close to zero and that the vast bulk of the vector is within 2 and −2. This is because ±2 standard deviations of a random normal variable contain roughly 95% of the distribution.

The cool thing is that we can work with these vectors just like we could work with single numbers. For instance, vectors can be added, multiplied, or subtracted. This will come in very handy for us, when we're going to simulate a variable as having a causal effect on some other variable. The following code snippet shows an example.

```
n <- 1e4
Z <- rnorm(n, 0, 1)
bZ <- 0.5
X <- Z * bZ + rnorm(n, 0, 1)
```

The first two lines should look familiar. Just like before, we define `Z` as a normally distributed variable with length `n`, mean 0, and standard

deviation 1. The following two lines define `X` as a function of `Z`. Specifically, $X = Z \times \beta z$, where βz is defined as 0.5. That is, `X` is half that of `Z`. We also add some normally distributed random noise to `X` to get more realistic-looking values. To make results numerically reproducible, you'll notice that we'll often use the function `set.seed()` at the very beginning of a simulation. This allows the reader to follow along and obtain identical results to ours.

This section served as a very brief introduction to simulating and working with variables in `R`—for a more elaborate primer on data simulation, see, for example, Fox, Nianogo, Rudolph, and Howe (2022). It's a cornerstone in the EEESI workflow to be able to simulate a set of variables with a specific causal structure. That way, we always know what the right answer is—after all, we simulated it ourselves! Furthermore, it facilitates testing our estimators against this ground truth. We'll build on these simulation techniques throughout the volume.

Formal Notation

In addition to software examples, we'll use some simple formal notation and terminology. This allows us to make the discussion more succinct as we go. It's useful to learn basic notation for another reason other than succinctness. Perhaps you'll want to navigate the technical causal inference literature on your own when we're done here. In that case, knowing the basics will help you follow along and progress. Confusingly, notation differs across sources. For the sake of clarity and exposition, our notation is at times a bit more verbose than what is commonly found in the literature.

Throughout, we will refer to our outcome variable as Y, which is the variable we want to understand and measure. We will examine how Y might be affected by our focal predictor variable X. This variable X can take on different values that represent different conditions or groups. For example, we'll use the notation $X = 1$ to mean that an individual receives a particular treatment and $X = 0$ to mean that the individual does not receive the treatment. More generally, we can say that $X = x$ means that X takes the value of x.

To express what are called **potential outcomes** for each value of X, we introduce the notation $Y^{X=x}$, which represents the outcome Y that would occur when X is set to a specific value x. So, when $X = 1$, the outcome is denoted $Y^{X=1}$, which is the outcome we would observe if the individual received the treatment. Similarly, when $X = 0$, the outcome is $Y^{X=0}$, representing the outcome if the individual did not receive the treatment. In the technical

literature, these outcomes may sometimes be referred to simply as Y^1 for the treated group and Y^0 for the nontreated group.

Next, we introduce the concept of the expected, or average, outcome under each condition. Using the expectation notation E[.], we define $E[Y^{X=1}]$ as the average outcome we would expect for individuals if they receive the treatment and $E[Y^{X=0}]$ as the average outcome for individuals if they do not receive the treatment.

Under this notation, we can now formally express the average causal effect, also called the average treatment effect (ATE), as the difference between the expected outcomes under treatment and no treatment:

$$E[Y^{X=1}] - E[Y^{X=0}].$$

This difference represents the average effect of the treatment on the outcome, providing a measure of the causal effect of receiving ($X = 1$) versus not receiving ($X = 0$) the treatment on Y.

We'll use words like "treatment" and "exposure" interchangeably, although we attempt to reserve the former to experimental contexts and the latter to observational settings.

Next, let's clarify some notation about expected values and probabilities. For continuous outcome variables, we use the notation E[.] to represent the expected or average value of an outcome. However, when we're dealing with binary outcomes, which can only take values of 0 or 1, we'll use Pr[.]. This is because when our outcomes are binary, we will model the probability of getting a 1. Later, we'll also encounter expressions like $E[Y \mid X,Z]$, which just means "the expected value of Y given specific values of X and Z." Here, the vertical bar | signifies "given" or "conditional on." Such conditional expectations allow us to analyze how the distribution of Y changes when X and Z are held at specific values. Conditional expectations are central to understanding relationships between multiple variables, as it allows us to analyze the effect of one variable while accounting for the influence of others.

Now, we'll practice this notation as we proceed. In all but the very simplest cases, we'll tackle a novel concept with narrative explanation, followed by or interwoven with formal notation, compact working code, and simple visuals in order to reinforce key ideas. Without further ado, let's go!

Chapter Highlights

- The important role of causal inference in data analysis
- The counterfactual view of causal inference
- The EEESI workflow

- Simulating and working with variables in R
- Formal notation

Further Reading

Pearl, Glymour, & Jewell, 2016, chap. 1; Hernán & Robins, 2020, chap. 1; McElreath, 2020, chap. 5; Rohrer, 2018

CHAPTER 2. CAUSAL GRAPHS

It's the year 1747, and the world's great colonial powers clash over land and riches. Amid this tumultuous quest for dominion, a silent, relentless killer claimed more sailors than rough seas and smoke-saturated naval battles. With symptoms marked by bleeding gums, wounds that wouldn't heal, and aching limbs, doctors today recognize this illness as scurvy.

The root cause of scurvy was a mystery. Theories abounded, from imbalanced "humors" and laziness to a lack of fresh air and exercise on long voyages, but none pinpointed the true culprit. That is, until James Lind, a Scottish physician, boarded the *HMS Salisbury*. In a pioneering experiment, Lind divided scurvy-stricken sailors into groups that were given different dietary supplements but were otherwise fed and housed similarly. Only one group, the sailors who received a daily ration of citrus fruits, recovered noticeably.

Lind's experiment is widely recognized as one of the first **randomized controlled trials** (RCTs) in history, and it unveiled the vital role of vitamin C in preventing scurvy. However, it took decades before another physician, Gilbert Blane, replicated and popularized Lind's findings. Blane's advocacy led to the Royal Navy's implementation of lemon juice rations, a move that drastically curtailed the scourge of scurvy among British sailors (Ernst & Singh, 2008, chap. 1). You could say that the British carved their victories in this defining period not purely by brute force or stratagems but with evidence, logic, and lemons.

Confounding and DAGs

This historical backdrop sets us up to learn more about DAGs. DAGs, or **directed acyclic graphs**, are useful tools for thinking through causal modeling problems (Pearl et al., 2016). Using nodes and arrows, a DAG plots the assumed causal relationships between variables. **Nodes** are variables and **arrows** indicate the direction of a causal relationship. A DAG is helpful in at least two ways. First, it sketches out and makes transparent our causal assumptions. As such, it forces us to think harder about the system we're interested in understanding. Second, DAGs guide subsequent analytic strategies for recovering the causal effects of interest. Given its particular structure, a DAG can even reveal that no such effects can be recovered under any circumstances.

That a DAG is *directed* means that the relationships, denoted by arrows, are causal. *Acyclic* means that if you travel along the direction of the arrows of a DAG, you'll never end up where you started. That is, variables can't cause themselves, so there are no feedback loops—although in Chapter 4,

we will see how to incorporate time-varying relationships in DAGs. And *graph* means . . . well, it's a graph.

Figure 2.1 is one example of a DAG. It shows a simple but very common scenario in which both our cause X (e.g., diet) and outcome Y (e.g., scurvy) are influenced by a third variable Z (e.g., lifestyle factors). In other words, lifestyle is a **CONFOUNDER** in that it influences both our suspected cause and our outcome. We will discuss confounding further in this chapter. For now, suffice it to say that our estimate of the effect of X on Y will be biased if we do not take Z into account.

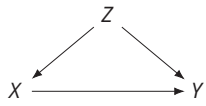
Citrus Fruits and Back-Door Paths

How do we know that Z confounds the relationship between X and Y and thereby induces bias? Simple, we can read it off the DAG. To see this, note that Z has a causal effect on both X and Y. This is technically called a **BACK-DOOR PATH**. A back-door path between X and Y is any path connecting X to Y that begins with an arrow pointing into X.

For the sake of the example, consider that if we find a statistical relationship between diet and scurvy, it could be due to the fact that sailors who prefer certain foods X (e.g., citrus fruits) also have other lifestyles Z (e.g., higher or lower naval rank) compared to sailors who eat differently and therefore have lower or higher risks of scurvy Y. In that case, we could not conclude that citrus fruits protect against scurvy; citrus fruits could simply be a proxy for other lifestyle factors that independently protect against scurvy symptoms. So, to make sure we are estimating an unbiased effect of diet on scurvy, we need a way to "block" the back-door path through lifestyle and remove this confounding effect.

In a basic sense, causal inference is about identifying such back-door paths and then blocking or circumventing them using study design—like the controlled trials of Lind and Blane—or statistical adjustment, which we'll explore in a little while. This simple insight makes analyzing DAGs a sort of party puzzle. It can actually be *fun* (for those so inclined) to draw out more or less complicated DAGs and then identify how—or even whether—a causal effect can be recovered under a particular graph.

Figure 2.1 Simple directed acyclic graph (DAG) of confounding.



As we'll see, there is much more to causal graph theory than looking for and blocking back-door paths, but this is a good place to start. So how do we block back-door paths? We just said that we can use particular study designs. Randomization is the best-known example of such a design.

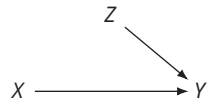
Randomizing a DAG

What **RANDOMIZATION** does, from the perspective of DAGs, is that it deletes any arrows that go into the randomized node, *X* in our case. This means that there can no longer be any back-door paths, by definition. Figure 2.2 illustrates this. It is similar to Figure 2.1, except that there is no arrow from *Z* to *X*. On this graph, an estimate of the effect of *X* on *Y* is no longer biased, and we do not need to measure *Z*.[1]

Randomization allows us to delete arrows going into *X* from all other variables because randomization is, well, random. And so—at least in expectation—this ensures that any differences in the outcome can be attributed solely to *X*, rather than to any preexisting differences between groups. In our scurvy example, consider that if we could randomize it, diet could not be impacted by other lifestyle factors. After all, we just randomly assigned different diets! So, assuming perfect adherence, whether an individual eats citrus fruits or not will be independent of anything else than the randomization mechanism that we used. And, in turn, and as recognized by Lind and Blane, any relationship we might find between vitamin C–rich foods and scurvy over time can only be due to the assigned diet. In the next section, we'll show in code and simple simulated data how this works.

In sum, randomization is an efficient strategy for causal estimation, because it makes *X* independent of all other variables in the graph. This is the reason why randomized controlled studies are often considered the "gold standard" for causal inference. But, unfortunately, our work here is

Figure 2.2 DAG of randomized *X*.



[1]However, we might want to measure and account for *Z* for another reason, which is to increase the *precision* of the treatment effect estimate, something we'll discuss again in Chapter 8.

not done. Randomization is often imperfect. For instance, participants might not perfectly adhere to the randomized treatment, or they might drop out of the study, potentially resulting in noticeable differences between groups despite randomized treatment assignment. Such differences can even arise by chance alone, particularly in small samples. Randomization is also often not feasible for practical or ethical reasons. For instance, how could we in real life—ethically or practically—assign people to strictly eat or live certain ways in a truly randomized fashion? A different but related problem is that the individuals we study in a randomized trial are often only partially—or not at all—overlapping with the population that we ultimately seek to make inferences about.

This does not mean that all hope is lost, and in later chapters, we'll tackle these challenges. Causal inference in observational or pseudo-randomized contexts is possible, but it does force us to make some assumptions and add some steps to our workflow. For instance, through statistical techniques like regression, we can block back-door paths that could not realistically be deleted by randomization. The next section introduces the promises and pitfalls of this approach.

Our objectives in the following, then, are twofold. First, we need to be able to recognize different causal relationships in DAGs in order to assess whether and how a causal effect can be estimated. Second, we want to illustrate and practice the EEESI workflow that we only briefly introduced in Chapter 1.

Elementary Ingredients of DAGs

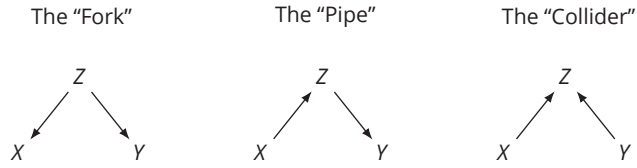
So, how do we identify and block back-door paths? It is actually pretty simple, at least in principle. Figure 2.3 shows three distinct patterns that we need to be able to recognize in a DAG. We could think of these patterns as the elementary *ingredients* of DAGs, since all DAGs more complicated than the simplest $X \rightarrow Y$ are constructed by one or more of these.

The "Fork"

Consider first the **FORK** on the left. You'll see that it's similar to Figure 2.1, except that we've left out the arrow $X \rightarrow Y$. A fork implies that X and Y are only associated because of a common causal influence of Z. In other words, the relationship between X and Y is *confounded*.

Given this DAG, then, we should not expect a direct causal influence of X on Y; they are only associated through the confounding back-door path $X \leftarrow Z \rightarrow Y$. By statistically adjusting for Z, we'd block the back-door path, making X and Y independent of each other. One way to think

Figure 2.3 Elementary ingredients of DAGs.



about adjustment is that we "hold constant"—or "control for"—the confounding variable by only comparing individuals with the same Z values.

For instance, in our scurvy example, we could look at the effect of lemon juice on scurvy only between sailors with similar lifestyles, such as diet and exercise habits. Say we indeed find a protective effect of lemon juice on scurvy. We know that lifestyle cannot account for this result, since we found the effect among individuals who lead very similar lifestyles. That is, the effect of lifestyle is effectively "held constant." In sum, adjusting for lifestyle Z achieves the same thing as randomizing X, in that it allows us to delete the arrow $Z \rightarrow X$.

Let's verify this in the simplest situation we can think of by practicing the EEESI workflow. To take full advantage of the EEESI workflow, there are techniques and concepts that we haven't introduced as of yet, but for now, what we can say is that our estimand is $X \rightarrow Y$, and our estimator should adjust for Z.

We want to check this with simulation, so we first simulate from the DAG. Based on the "fork" DAG in Figure 2.3, we don't expect an effect of X on Y (i.e., the effect $X \rightarrow Y$ is 0). Therefore, `X` is not a part of the function that defines `Y` in the simulation code. Ideally, our estimator recovers this noneffect, meaning that the effect of X on Y should be very close to 0. However, Z *does* feature in both the definition of X and Y, inducing a spurious relationship between the two. Except for the final line where we collect all variables in a dataset `d` using the `data.frame()` function, the code follows the same recipe as given in the short section in Chapter 1 on simulating variables in `R`—feel free to revisit that section if something looks unfamiliar. Throughout, we'll just stick with an arbitrary relationship between Z and the other variables. In this case, the relationship—as indicated by the regression term `Z * bZ`—will be 1. But remember, we're naive to all this in real-world data, and so we proceed as if we hypothesize a causal relationship between X and Y, our estimand.

```
set.seed(1747)

n <- 1e4
bZ <- 1
Z <- rnorm(n, 0, 1)
X <- Z * bZ + rnorm(n, 0, 1)
Y <- Z * bZ + rnorm(n, 0, 1)

d <- data.frame(X = X, Y = Y, Z = Z)
```

First, we show what happens if we ignore Z and simply regress Y on X. That is, we're after the expected value of Y given X, or $\mathrm{E}[Y \mid X]$. This is written with the formula `Y ~ X`. We use the in-built `lm()` function (for "linear model") to conduct our regression analysis, but this basic syntax is similar across many `R` packages.

```
lm(Y ~ X, data = d)
```

```
(Intercept)            X
   0.007385     0.484392
```

We see in the output that the regression coefficient with these settings for X is around 0.5, which is noticeably more than the "true" effect of 0. As we *know* it is supposed to be 0, this is not good!

Next, we demonstrate our desired estimator, namely, a model that adjusts for Z, that is, $\mathrm{E}[Y \mid X, Z]$. This model correctly picks up the lack of an effect of X on Y, namely, a coefficient for X that is very close to 0. Success![2]

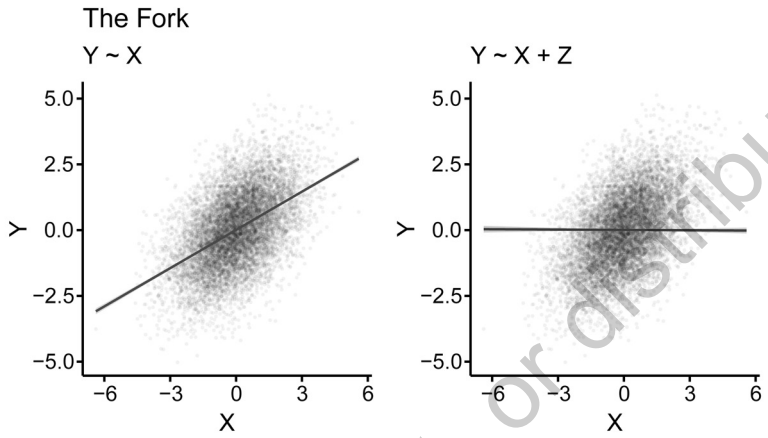
```
lm(Y ~ X + Z, data = d)
```

```
(Intercept)           X           Z
   0.007601   -0.004406    0.987202
```

Let's visualize these results. We can graphically depict how our two models estimate different regression lines for X depending on the adjustment of Z in Figure 2.4: The model that ignores Z (left panel) picks up a spurious noncausal association, whereas the model that adjusts for Z (right panel) correctly estimates no causal effect of X on Y.

[2]The coefficient is close to but not *exactly* 0 due to the finite sample size and sampling variation. This caveat applies to all our simulation examples.

Figure 2.4 "Breaking the fork." Regression lines for X on Y without (left) and with (right) adjustment for Z in the "fork" scenario.



The "Pipe"

Now, let's focus on the **pipe**. As with the fork, the pipe implies that X and Y are only associated through Z. This time around, though, Z is not a confounder, since it is causally influenced by X. Instead, Z is what is known as a **mediator** on the path between X and Y. This complicates our adjustment strategy.

Consider that vitamin C intake is a mediator Z on the causal path from diet X to scurvy Y. In other words, diet impacts scurvy risk but only through vitamin C; diet composition does not directly affect scurvy risk without considering its vitamin C content. If we adjust for vitamin C intake—meaning looking only at sailors with the same level of vitamin C consumption—we would find little to no effect of different diets on scurvy. In the pipe example, there's no arrow from X to Y, so it's not diet in and of itself that causes or prevents scurvy. What matters is whether the diet provides sufficient vitamin C. So, once we take vitamin C intake into account, other aspects of diet no longer help explain whether a sailor develops scurvy. We have, as it were, blocked the effect of diet by adjusting for a mediator.

More generally, if we're interested in estimating the **total effect** of X on Y, this includes all paths that leave X and enter Y. In this case, there is only one such path, $X \rightarrow Z \rightarrow Y$. If our estimand is the total effect, then we would not adjust for Z, since adjusting for Z would block the part of the

causal path from X to Y that runs through Z, as we just saw. However, say we hypothesized a **DIRECT EFFECT** $X \rightarrow Y$ in addition to the mediating path through Z. For instance, maybe different diets could have a direct effect on scurvy above and beyond vitamin C–rich foods. If we were interested in estimating only that direct effect of different diets X, we would have to adjust for vitamin C intake Z, since Z would otherwise bias the estimate.

As with the fork, we'll again take this opportunity to consider the EEESI workflow. Say that our estimand is the direct effect of X on Y, meaning we want an estimator that adjusts for Z. As above, we first simulate some simple data that are consistent with a pipe: Note that X causes Z with the `X * bX` term and that Z causes Y through the `Z * bZ` term, but that X does not feature in the definition of Y.

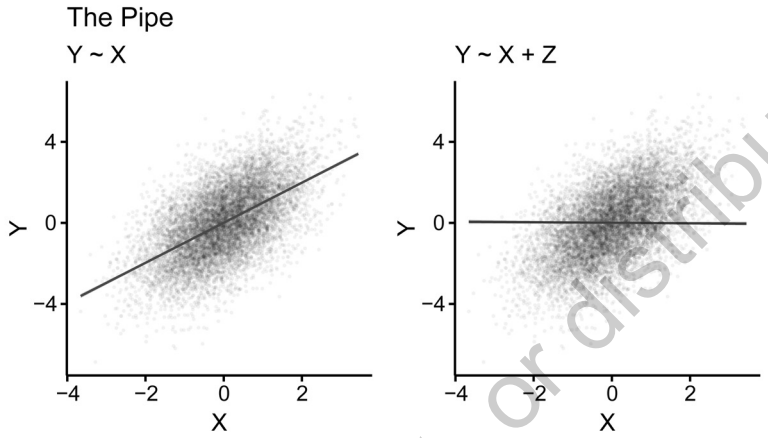
```
set.seed(1747)

n <- 1e4
bZ <- 1
bX <- 1
X <- rnorm(n, 0, 1)
Z <- X * bX + rnorm(n, 0, 1)
Y <- Z * bZ + rnorm(n, 0, 1)

d <- data.frame(X = X, Z = Z, Y = Y)
```

And then we'll fit two models, one that ignores Z and another that adjusts for Z. These would look identical to the two models fit under the fork, so no need to repeat them. The model that ignores Z will pick up an effect of X on Y, which is the effect $X \rightarrow Z \rightarrow Y$. Had our estimand been the total effect, this would be the correct model. Instead, the model that adjusts for Z rightly recovers the lack of a direct effect of X on Y. We can again plot the two models to see our inferences in action (Figure 2.5).

The "Collider"

With both the fork and the pipe, then, X and Y are associated, unless we adjust for Z, either because Z is a common cause or a mediator. With our final ingredient, the **COLLIDER**, it's different. In fact, it's the opposite. In a collider, X and Y are associated *only if* we adjust for Z. That is, rather than closing the back-door path, adjusting for Z opens one. This is sometimes referred to as **COLLIDER BIAS**. If you're like most people, this is not very intuitive on a first read, so let us spend a little bit more time on the collider concept, continuing with James Lind's scurvy studies.

Figure 2.5 "Blocking the pipe." Regression lines for X on Y without (left) and with (right) adjustment for Z in the "pipe" scenario.



Suppose Lind wanted to study the relationship between scurvy Y and sailors' laziness X, for instance, by measuring sailors' activities on deck. Since laziness was hypothesized as a potential cause of scurvy at the time, it's not an entirely unreasonable research question. The trouble is that observing the relationship between scurvy and laziness could be distorted by an important variable: whether a sailor appears on deck for duty Z. After all, we can only measure engagement with shipboard activities among sailors who are physically present on deck. In other words, appearing on deck is a collider on the path Laziness → Appears on Deck ← Scurvy—both laziness and scurvy independently make a sailor less likely to report for duty. So, if we condition on appearing on deck—for instance, by restricting our analysis to only include sailors observed working at a given time or by adjusting statistically so that we only compare sailors with similar deck attendance—we might wrongly conclude that scurvy and laziness are causally related, since both can keep sailors from showing up.

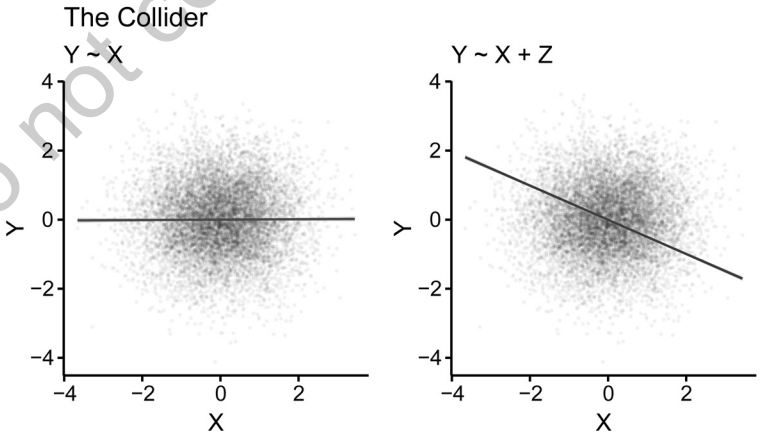
To build good habits, let's see how this looks in a simulation. The important difference from above is that Z is now influenced by X and Y via the terms `X * bX` and `Y * bY`. Once again, we'll stick with the arbitrary effects of 1 for our coefficients.

```
set.seed(1747)

n <- 1e4
bX <- 1
bY <- 1
X <- rnorm(n, 0, 1)
Y <- rnorm(n, 0, 1)
Z <- X * bX + Y * bY + rnorm(n, 0, 1)

d <- data.frame(X = X, Y = Y, Z = Z)
```

We can again consider two models, one that does not adjust for Z (`Y ~ X`) and another one that does (`Y ~ X + Z`). Again, these models would look identical to the two regressions fit with the fork, so we won't repeat them. Under these particular simulated values, the first model correctly recovers the lack of a relationship, a coefficient of 0, while the latter model picks up a noncausal association between X and Y of around −0.5. Figure 2.6 illustrates this.

Once you've built some intuition for collider bias, you'll start seeing it everywhere. In particular, our collider bias alarm should sound whenever there is filtering or selection on variables that might be related to our outcome of interest—just as in our scurvy and deck attendance example.

Figure 2.6 "Statistical collision." Regression lines for X on Y without (left) and with (right) adjustment for Z in the "collider" scenario.



Collider bias is also related to what we will later learn to recognize as **SELECTION BIAS**. Both concepts are critical to understanding how sampling strategies and missing data can bias inference, topics that we'll explore in more depth in Chapters 5 and 6.

What Makes Societies Tick?

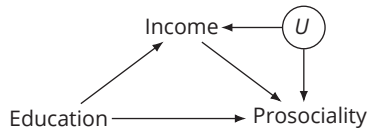
Let's bring in a real-world example to tie all of this together. Consider the following research question: What makes people cooperate? Cooperation is at the heart of human social life, and social scientists have long speculated about the roots of human sociality. Is it education, markets, religion, or shared history, resource scarcity, resource abundance, "cultural norms," all, or none of the above? As it turns out, researchers have not stopped short at speculation. An entire industry of studies has tried to tackle these questions empirically by probing people from a wide range of diverse societies with batteries of measurement tools (Pisor, Gervais, Purzycki, & Ross, 2020).

One common metric of cooperativeness—or "prosociality"—are behavioral economic games. In a typical behavioral economic game, participants are endowed with some resource such as money, and are then asked to make decisions about sharing, investing, or dividing this resource with others. When researchers play such games with people from around the globe, different cross-cultural patterns seem to emerge. For instance, some keep most or all of the money for themselves, while others donate most of their endowment. Others again split the money more or less evenly (Spadaro et al., 2022). The question is, what determines these differences in gameplay across cultures?

Imagine we hypothesized that education makes people more or less prosocial. Our estimand is thus Education → Prosociality. For instance, formal education might render people more "economically rational," and therefore, they might keep more coins for themselves. Alternatively, education might make people more sensitive to others' needs, and hence, educated individuals would donate a larger amount to the anonymous other. Whatever the hypothesized causal link, it's very likely that any effect of education on prosociality would also work through other mediating variables. One such mediating variable is income. We can, for instance, assume that education increases income on average, and income levels, in turn, plausibly influence how selflessly people would play the game.[3]

[3]Of course, many other variables could be at play here, but we keep it simple for the sake of illustration.

Figure 2.7 DAG with unobserved confounder *U*. Our estimand is the direct effect Education → Prosociality. But Income is both a mediator and a collider.



Consider now that some unobserved variable influences both income and prosociality, resulting in the causal structure depicted in Figure 2.7. We've encircled the variable *U* to emphasize that it's unobserved. Here, *U* could be many things, but a popular candidate is what's often called "market integration." Market integration is a technical term that economists use to describe how reliant people are on markets for food and other necessities, in contrast to being self-sufficient (Henrich et al., 2001). The key idea is that routinely dealing in markets with strangers requires norms of interpersonal trust and reciprocity. These norms might psychologically spill over into the experimental game setting, in turn making people more prosocial. *U*, then, could also more generally stand for certain aspects of "culture."

Here's the rub. To evaluate the direct effect of education on prosociality, we'd want to adjust for the mediating variable income. However, income is a collider in that it's influenced by both education and *U*. Adjusting for income therefore opens up a back-door path between education and prosociality through *U*. Under this scenario, we have no way of estimating the direct effect of education on prosociality in lieu of additional variables and assumptions. At best, in this simplified example, we could settle for the total effect by not adjusting for income.

This is a very unfortunate scenario. It's also quite common. For instance, various discrimination scenarios have been argued to take this general causal structure (Lundberg et al., 2021), and as we'll see in Chapter 4, it's also a very real threat to mediation analysis. We'll revisit the prosociality example with real-world data in Chapter 7 to see what we can do about *U*.

Good and Bad Controls

In the preceding section, we saw how DAGs are useful for sketching out causal problems and identifying a statistical modeling strategy. And, as with the direct effect of education on prosociality in Figure 2.7, DAGs also reveal if no solution is available for a particular estimand. Both are valuable

insights. Further, we saw how we can use multiple regression as an effective way to block back-door paths. All of this constitutes a wonderful promise, since it allows us to study causal relationships that would not be feasible or ethical to study in an experimental context. It also allows us to decompose effects of interest, which is particularly relevant in mediation scenarios, and for estimating an unbiased causal effect even in the presence of unmeasured confounding, using a formula known as the **FRONT-DOOR CRITERION**. We will discuss mediation and the front-door criterion in Chapter 4.

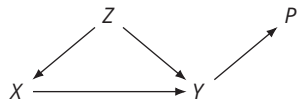
However, when using statistical adjustment, we must tread with care. There are both good and bad controls out there. Just as statistical adjustment can block back-door paths and thereby de-confound a true causal effect, statistical adjustment can induce bias if we adjust for the wrong variables (Achen, 2005; Cinelli, Forney, & Pearl, 2020). We saw this in the pipe and collider examples: Adjusting for a mediator *blocks* a true causal effect, while adjusting for a collider *opens up* a non-causal path.

Posttreatment Bias

A related scenario is when a "posttreatment variable" is adjusted for. **POSTTREATMENT BIAS** occurs when adjusting for variables that are downstream consequences of the treatment, and it can likewise distort a causal effect (Montgomery, Nyhan, & Torres, 2018). In the context of Lind and Blane's scurvy experiments, imagine if they had adjusted their analysis based on a variable that was influenced by a vitamin C–rich diet on scurvy, such as wound recovery or energy levels. This could have masked the observed effect of diet on scurvy. Let's see how.

Figure 2.8 shows an example of a posttreatment variable P that is causally influenced by the treatment X through the outcome Y. We can simulate from this DAG and see what happens. To be explicit about the workflow, our estimand is $X \rightarrow Y$, and our graph analysis tells us that we want an estimator that *ignores* the posttreatment variable P but *adjusts* for the confounder Z. This should recover the "true" effect of 0.5 (in the code, `bX <- 0.5`).

Figure 2.8 DAG with a posttreatment variable P.



```
set.seed(123)

n <- 1e4
bZ <- 1
bX <- 0.5
bY <- 1
Z <- rnorm(n, 0, 1)
X <- Z * bZ + rnorm(n, 0, 1)
Y <- Z * bZ + X * bX + rnorm(n, 0, 1)
P <- Y * bY + rnorm(n, 0, 1)

d <- data.frame(Y = Y, X = X, Z = Z, P = P)
```

The simulation code looks a little more involved, but there are really no new tricks: We simulate variables with relationships dictated by the DAG and note that *Y* enters in the definition of the posttreatment variable *P*. As we're getting used to, we then fit two models: one that adjusts for the posttreatment variable *P* and another that ignores *P*.

```
lm(Y ~ X + Z + P, data = d)
```

```
(Intercept)           X           Z           P
-0.0008578   0.2514835   0.5166032   0.5006431
```

Inspecting the results from the first model, we find that this model estimates a coefficient for *X* of approximately 0.25, that is, an underestimate from the simulated effect of 0.5. In contrast, the second model recovers 0.5.

```
lm(Y ~ X + Z, data = d)
```

```
(Intercept)           X           Z
 -0.007081    0.496815    1.023246
```

One way to think of posttreatment bias is that if we control for variables (e.g., energy levels) that are influenced by the treatment (e.g., vitamin C–rich diet), we change the estimand: We're no longer estimating the effect of diet on scurvy. We're estimating the effect of diet on scurvy *among sailors with similar posttreatment energy levels*. And among sailors with similar posttreatment energy levels, we're unlikely to see a clear effect of diet on scurvy, because energy level is at least partly a consequence of diet through its effect on scurvy—vitamin C reduces the risk of scurvy, which increases energy. So by adjusting for posttreatment energy levels, we're

accounting for much of the effect of diet on scurvy. We're in a sense "controlling away" the treatment effect.

Posttreatment bias thus helps illustrate a more general point: Beware of descendants! **DESCENDANTS** are nodes that are downstream of other nodes. The opposite of a descendant is an **ANCESTOR**, a node that is upstream from another node. In Figure 2.8, X and Y are descendants of Z, Y is also a descendant of X, and P is a descendant of Y. When we adjust for a descendant, say P, we remove variation in the ancestor node, Y. Depending on the estimand in question, this is often undesirable because it can distort the causal effect of interest.

The takeaway from all this? Look out for posttreatment variables and descendants!

The "Table 2 Fallacy"

All of this suggests that we cannot generally interpret regression coefficients as "causes" of the outcome. A statistical adjustment set (e.g., whether or not to include Z in a regression) is built for a particular purpose and a particular set of causal assumptions (e.g., estimating the effect of X on Y, adjusting for assumed confounder Z). In other words, a statistical model is bespoke to the estimand. This, in turn, means that we cannot expect that coefficients from the variables used for statistical adjustment also represent particular causal effects. For all we know, parts of the adjustment set might be a collider or a mediator on some other path in the causal graph, rendering a causal interpretation invalid.

This fundamental insight stands in contrast to common practice in many empirical sciences, where a table (usually Table 2) with all estimated regression coefficients is often presented and interpreted—implicitly or explicitly—as direct causal effects. This is a very unfortunate habit as it potentially misleads the reader to consider coefficients of the control variables causally. To raise awareness of this malpractice, Westreich and Greenland (2013) dubbed it the "Table 2 Fallacy."

Where Do DAGs Come From?

As we have seen, DAGs are useful tools as visual representations of causal relationships among variables. DAGs help us map out assumptions, understand causal structures, and guide the selection of appropriate methods for causal estimation. But where do DAGs come from? While there are no hard-and-fast rules, several routes are often taken to arrive at a sensible DAG or set of DAGs.

First, prior studies or analyses offer clues to potential causal links and hypotheses. By systematically reviewing literature, we can start sketching DAGs that reflect both established theories and results, ensuring that the graph is rooted in some kind of empirical reality.

Second, domain experts can provide insights into potential causal relationships that are not immediately apparent from existing literature or data patterns. They can help identify confounders, suggest plausible mechanisms, and clarify the direction of causal arrows. Engaging with experts can be especially useful in fields where empirical evidence is limited or where causal processes are complex.

Third, common sense should come into play. For instance, since only the past can cause the future and not the reverse, the temporal ordering of variables can dictate causal directions. Also, particularly in the health and social sciences, biological variables like age or sex are often considered relevant factors, but nothing causes them other than time or random assignment. So we can be quite certain that no causal influences flow into age or biological sex. That is, such variables can only ever be ancestors, not descendants.

Fourth, important metadata might be necessary to include in your DAG. That is, practical factors that went into the study are often worth considering. Such factors could include the order of tasks in a study design, how data were collected, and whether there are known sources of measurement error or missing data (Chapters 5 and 6).

Beyond these, qualitative approaches like interviews and focus groups can offer insights into causal hypotheses, and quantitative methods, like data-driven "causal discovery" algorithms, are able to rule out certain causal patterns via a series of analytic steps and assumptions (Glymour, Zhang, & Spirtes, 2019; Molak, 2023).

Constructing a DAG, then, has parallels to how science progresses more generally. Like science, DAG construction is a cumulative—if error-prone—process, whereby competing theories are continually proposed, empirically evaluated, and conceptually refined. In the same way, we can propose competing DAGs and then evaluate their implications against data and prune or extend them on that basis. As in science, this is an iterative, open-ended process. And also like science, the goal is not necessarily to arrive at one "true" causal graph. A causal graph, like a scientific theory, will almost always only be provisional. Rather, the best we can often hope for is to edge ever closer to a plausible and useful approximation of how the world—or at least a tiny sliver of it—really works.

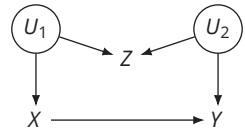
Assumptions Before Analysis

Our preceding discussion of DAGs and adjustment strategies implies a perhaps somewhat counterintuitive point that complicates DAG construction: *Causes are not in the data*. That is, we need causal assumptions before we can do causal data analysis. In general, we cannot go in the other direction, from data analysis to assumptions. Even causal discovery methods that promise to infer causal structures from data themselves require—often very strong—assumptions!

Why is it challenging to infer causal structures (e.g., DAGs) from data alone? To see why, let's look at a basic example. Consider, for instance, that it's not generally possible from data alone to distinguish between a pipe and a fork. This is because they have similar statistical implications. In both cases, X and Y are associated unless we adjust for Z. But, for any given hypothesis—think again of a medical treatment, policy program, or marketing campaign—whether Z is a mediator or a confounder is critical information from a causal perspective. In the case of the pipe, you can influence Y through Z by manipulating X. But in a fork scenario, manipulating X will not cause a change in Y. To proceed in such cases, then, we are forced to make assumptions about the role that Z plays in the causal structure based on external information.

Consider another scenario, depicted in Figure 2.9. Here, Z is associated with both X and Y through two unobserved variables U_1 and U_2, resulting in confounding between Z and X and Z and Y. From looking only at the data, then, Z masquerades as a confounder, and an analyst could be tempted to adjust for Z to estimate the effect of X on Y. But this would be a mistake. Strictly speaking, Z is not a confounder, since there's no open back-door path from Y to X through Z. Z is instead a collider, and the back-door path is only open if we adjust for Z. This scenario is sometimes referred to as **M-BIAS** due to the particular "M" shape of the DAG.

Figure 2.9 M-bias DAG. Z masquerades as a confounder: Z is associated with both X and Y through two unobserved variables U_1 and U_2 but is in fact a collider.



Such examples highlight that a DAG generally cannot be built on the basis of any number of statistical models alone. Regressions are just prediction engines, so their coefficients only have a valid causal interpretation to the extent that causal assumptions hold. Instead, regressions and other models can be used to *evaluate* the implications of a particular DAG or set of DAGs (for discussion of this point in the wild, see Bendixen, 2023; Purzycki, Bendixen, & Lightner, 2022). All this means that selecting which variables to include in our adjustment set should not be fully automated. Rather, variable selection strictly requires prior knowledge and explicit causal assumptions (Hernán & Robins, 2020, chap. 18; Westreich, 2019, chap. 3).

Add to this predicament that causal graphs such as DAGs leave very important information on the table, namely, the *functional relationships* between variables. What do we mean by functional relationships? So far, we've modeled variables as linear functions that are fit for analysis with linear regression. This allowed us to introduce basic concepts with simple simulation examples. But, alas, many if not most real-world contexts are often complex and nonlinear. And the point here is that we cannot read this information off of a DAG. A DAG only declares the *direction* of causes between variables but not the exact *statistical nature* of those relationships.[4] We'll have more to say on this point in later chapters. But for now, it serves as a reminder of why simulation is such an important element in our EEESI workflow: Nonlinear relationships are often unintuitive, and they can even sometimes change our estimand. So, instead of trying to work it all out mentally, we simulate our hypotheses and check that our estimand is indeed recoverable with the estimator we had in mind.

That is, to avoid the Table 2 Fallacy and similar pitfalls, our EEESI workflow, as we've practiced it so far, prescribes the following set of iterative steps:

1. Identify estimand: What is it that we want to estimate?
2. Draw out causal assumptions, for instance, using DAGs.

[4]When a DAG is combined with this kind of information—that is, when the statistical distributions of the variables in the DAG as well as their mathematical relationships are specified—we say we have a **STRUCTURAL CAUSAL MODEL**. So, for instance, our simulation exercises are in fact structural causal models, since they are both informed by their respective DAGs *and* we specify distributions and relationships of and among the variables.

3. Identify an adjustment set that is consistent with the estimand: We want to block back-door paths and be on the lookout for mediators, colliders, descendants, and posttreatment variables.
4. Build the estimator and simulate data to ensure that the estimator recovers the estimand.
5. Apply the estimator to real data.
6. Report and interpret the focal estimate(s) in light of the estimand.

We'll continue to practice and build on this workflow as we move through later chapters. But notice that we've already made quite a bit of progress!

Average People and People on Average

Before we close this chapter, we need to familiarize ourselves with one more important issue: the distinction between **CONDITIONAL** and **MARGINAL** effects. If that doesn't ring a bell, all is good. Statisticians can be lovely people, but they tend to be terrible at naming things. So don't worry, you'll get used to this technical jargon down the road. We're laying the foundations here for later chapters, so let's take it from the top. This distinction helps us build a bridge to subsequent chapters and connect them to the point above about how functional relationships can change estimands.

In all but the simple cases, regression analysis yields **CONDITIONAL EFFECT ESTIMATES**. All this means, for now, is that the regression coefficient of interest is often *not* the average effect for the whole sample but rather for a subset, conditional on the other variables in the statistical model (Hanmer & Kalkan, 2013). To see this, consider the following example.

Say we randomize some marketing campaign X and assess its effect on sales Y (a binary variable, buy = yes/no) but suspect that age A might be modifying the effect of this intervention. The DAG would then be similar to Figure 2.2, except we'll switch out Z for A, such that it looks like this: $X \rightarrow Y \leftarrow A$. Since we randomized X, A cannot be a confounder. But we're interested in modeling A anyway, because it could interact with the intervention in interesting ways (e.g., younger people might be more or less willing to be persuaded by the "once-in-a-lifetime offer!!!" that we're launching).

To ease interpretation, let's imagine that we've centered and standardized age. What this means is just that age A is scaled such that its mean is represented as 0 and its standard deviation is 1. We'll see in a little bit why this is useful. Since X and Y are binary, observations are drawn from a

binomial distribution (using `rbinom()`) defined by the number of observations, *n*, the number of trials (here, `size = 1`), and the probability (`prob`) of a "success." In the present example, each participant has a 50% chance of getting exposed to the marketing campaign ($X=1$), whereas the probability of a sale ($Y=1$) is a function of X and A. The logistic function—`plogis()`—allows us to simulate a binary outcome with coefficients on the same scale as our simulated values. This simulation sets us up to illustrate how conditional and marginal effects can differ.

```
set.seed(1)

n <- 1e4
bX <- 1
bA <- 1
bXA <- 1
A <- rnorm(n, 0, 1)
X <- rbinom(n, size = 1, prob = 0.5)
Y <- rbinom(n, size = 1,
prob = plogis(-0.5 + X * bX + A * bA +
              X * A * bXA))

d <- data.frame(A = A, X = X, Y = Y)
```

To model the binary outcome Y, we use a logistic regression, which we specify in the `glm()` function by setting `family = "binomial"`. We fit an interaction between the intervention X and age A using the `:` operator, allowing us to estimate a different effect of the treatment across different ages.

```
mod <- glm(Y ~ X + A + X : A,
           data = d,
           family = "binomial")
```

Now, call the `mod` object and check the regression coefficient for the treatment effect `X` in the output. It's around 1 and the intercept is around −0.5, just as we set in the simulation. We also recover the right coefficients for age `A` and the interaction term `X:A`, namely (roughly) 1, but ignore those for now.

```
(Intercept)           X           A         X:A
    -0.4911      0.9545      1.0163      0.9179
```

Since we've used a logistic regression, these coefficients are in log-odds. But log-odds are not very intuitive, so to transform this to the probability scale, we use the logistic function, $\frac{\exp(x)}{1+\exp(x)}$. Then, using rounded coefficients for simplicity, we calculate the probability of buying the product for the treatment group, which is the intercept plus the treatment coefficient wrapped in the logistic function:

$$\Pr(Y = 1 | X = 1, A = 0) = \text{logistic}(-0.5 + 1) = \frac{\exp(0.5)}{1 + \exp(0.5)} \approx 0.62$$

And, next, for the control group, which is just the intercept, also wrapped in the logistic function:

$$\Pr(Y = 1 | X = 0, A = 0) = \text{logistic}(-0.5) = \frac{\exp(-0.5)}{1 + \exp(-0.5)} \approx 0.38$$

Subtracting the results and converting to percentages yields a difference of roughly 24%. This is the predicted increase in the probability of buying our product under exposure to the marketing campaign compared to no exposure. In R, we could perform the full calculation like so:

```
(plogis(-0.5 + 1) - plogis(-0.5)) * 100
```

So far, so good. But consider now what all of this means. In multiple regression, a coefficient represents the association of the predictor with the outcome when all other covariates are held constant. In this case, with age A standardized to have a mean of 0 and interacted with the treatment, the regression coefficient for the treatment X therefore represents the effect of the marketing campaign on sales, *when age is held at 0*—that is, at the average age. This is a conditional effect estimate: the treatment effect for average-aged individuals.

However, often we're not interested in the quantity of the effect of an intervention only at some specific level of a covariate. If we have many covariates that are both categorical and continuous, there might not even exist a single "typical" or "average" person who corresponds to the reference values of all the covariates (e.g., a mean-aged male, with mean years of education, mean number of children, living in the reference region of residence). In particular, in complex models, this means that our treatment coefficient might not represent an estimated effect for any given observed individual.

Because they are only conditional effect estimates, only in special cases do raw regression coefficients correspond to causal estimates of interest,

such as the average treatment effect. So we often want something else than the raw regression coefficients. Back to our marketing campaign example, imagine we want to roll out the campaign at scale in the full population from which our sample is a sufficiently large, random sample. In that scenario, as is often the case in applied contexts, we're interested in the effect of the intervention in the sample as a whole, which requires us to take an average *over the distribution of age*. In `R`, this can be done with the following procedure.

```
EX1 <- predict(mod,
               newdata = transform(d, X = 1),
               type = "response")

EX0 <- predict(mod,
               newdata = transform(d, X = 0),
               type = "response")

mean(EX1) - mean(EX0)
```

That might look rather mysterious at this point, but in the next chapter, we'll carefully explain each step of this procedure and learn to refer to it as "g-computation." So for now, we simply display it for completeness and delay a deeper discussion until then. The key insight to take away here is that the final estimate, where we subtract the means of `EX1` and `EX0`, is a marginal effect estimate. That is, it is the effect of the marketing intervention in the sample *on average*. In our simple simulated example, the marginal effect happens to be around 18%, which is quite different from the conditional effect estimate calculated above of around 24%.

Put another way, there is an important distinction between making inferences for "average people" (i.e., a conditional effect) or "people on average" (i.e., a marginal effect). The conditional and marginal effect estimates differ, because they are expressions of different estimands.

So there we have it. Our estimand is often the marginal effect, but in moderately complex cases, where we're modeling variables with potentially nonlinear relationships, regression only yields conditional estimates. At the same time, adjustment for covariates is often critical for de-confounding, particularly in observational settings, so we cannot do without statistical adjustment. This seems paradoxical, so what gives? The next chapter introduces a family of methods that promises to solve this problem—and much, much more.

Chapter Highlights

- DAGs, their elementary ingredients, and how to derive them
- The role of randomization, statistical adjustment, and the connection between the two
- The Table 2 Fallacy
- The distinction between conditional and marginal effects (average people vs. people on average)

Further Reading

Cinelli et al., 2020; Pearl et al., 2016, chaps. 2–3; Hernán & Robins, 2020, chaps. 6–7; McElreath, 2020, chaps. 5–6; Rohrer, 2018; Wysocki, Lawson, & Rhemtulla, 2022