Lessons in Statistical Thinking

Daniel Kaplan

12/27/22

Table of contents

Preface

> **Note to students in Math 300**
>
> Up to now, Math 300 has been following the *OpenIntro* textbook. For the remainder of the semester, however, we will continue with the lessons in this little book: *Lessons in Statistical Thinking*.

Lessons in Statistical Thinking is an update and reconsideration of the concepts and methods needed to extract information from data. Such an update is needed because the canon of traditional introductory statistics texts has long been obsolescent and fails to address the needs of the contemporary data scientist and decision-maker. That canon stems from an influential 1925 book, Ronald Fisher's *Statistical Methods for Research Workers*. Research workers of that era typically ran small benchtop or field experiments with a dozen or fewer observations on each of two treatments. A first task with such small data is to rule out the possibility that calculated differences might reflect only the accidental arrangement of numbers into groups.

Perhaps emblematic of the current dissatisfaction with small-data methods is the controversy over "statistical significance." Although situated at the core of many statistics textbooks, significance testing has little to do with the meaning of "significant" as "important" or "relevant." This article in the prestigious science journal *Nature* details the controversy. Figure **??** reproduces a cartoon from that article that puts the shortcomings of "statistical significance" in a historical context.

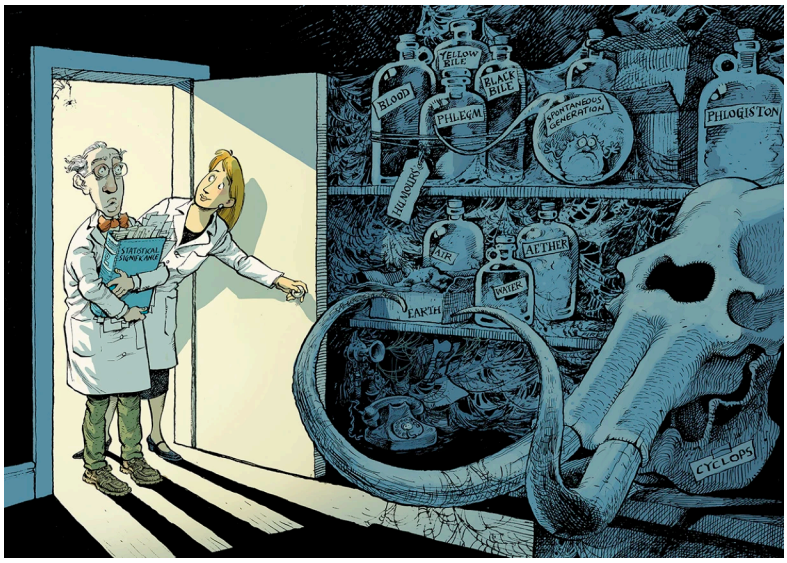


Figure 1: A cartoon published along with an article in *Nature*, "Retire statistical significance," showing this once-respected idea heading to the graveyard for outdated and misleading "scientific" concepts such as phlogiston and aether.

Statistical thinking

The work of today's data scientists is often to discover novel connections among multiple variables and to guide decision-making. It is common for data to be available in large masses from *observations* rather than *experiments*. One common purpose is "prediction," which might be as simple as the uses of medical screening tests or as breathtaking as machine-learning techniques of "artificial intelligence." Another pressing need from data analysis is to understand possible causal connections between variables.

The twenty lessons that follow describe a way of thinking that is historically novel, unfamiliar to most otherwise well-educated people, and incredibly useful for making sense of the world and what data can tell us about the world.

For reference: Important word pairs

Many of the vocabulary terms used in statistical thinking come in pairs. We list several such pairs below, in roughly the order they first appear in the Lessons. The pairs can be a reference

while reading, but it is also helpful to return to this list to sharpen your understanding of the distinctions.

Explanatory vs **response** variables. Models (in these Lessons) always involve a *single* response variable*. In contrast, models can have zero or more explanatory variables.

Variable vs **covariate**. "Covariate" is another word for an explanatory variable. The word "covariate" signals that the variable is not itself of direct interest to the modeler but puts another explanatory variable in a correct context.

Categorical vs **quantitative** variables. Always be aware of whether a model's response variable is categorical or quantitative. When categorical, expect to use `zero_one()` to convert it to quantitative before modeling. In contrast, explanatory variables can be either categorical or quantitative.

Regression model vs **classifier**. A regression model always has a *quantitative* response variable. A classifier has a *categorical* response variable. In these Lessons, as in much professional use of data, our categorical response variables will have *two levels* (e.g., healthy or sick, up or down, yes or no). In this situation, regression techniques suffice to build classifiers.

Model vs **model function**. By "model," we will almost always mean "regression model." A regression model, typically constructed by the `lm()` function, contains various information useful to summarize the model. The "model function" provides the mechanism for one important task, calculating from values from the explanatory variables the corresponding model output.

Model coefficient vs **effect size**. Model coefficients are numerical parameters. Training determines the appropriate values for the coefficients. In contrast, an effect size describes the relationship between the response variable and a selected explanatory variable.

Point estimate vs **interval estimate**. A point estimate is a single number. For instance, a model coefficient is a point estimate, as is the output from a model function. In contrast, interval estimates involve *two* numbers; one specifies the lower

end of the interval and the other number specifies the upper end.

Prediction interval vs **confidence interval**. A prediction interval describes the anticipated range of the actual result for which we have made a prediction, e.g., "tomorrow's wind will be between 5 and 10 mph." A **confidence interval** is often used to express the uncertainty in a coefficient or effect size.

Software guide

These Lessons use about a dozen new R functions. Some of these are used frequently in examples and exercises and are worth mastering. Others appear only in **demonstrations**.

> ⚠ Demonstrations
>
> These lessons contain *demonstrations* illustrating statistical concepts or data analysis strategies. We will place these in a distinctive box, of which this is an example. The demonstrations will often contain new computer commands that perform tasks used in teaching statistics. However, readers are **not** expected to be able to construct such commands on their own.

- Training models with data
  - `lm()` arguments: i. tilde expression, ii. `data=` data frame.
  - Occasionally, you will be directed to use `glm()` or `model_train()`, which work similarly to `lm()` but are specialized for models whose output is a *probability*.
  - `zero_one()` converts a two-level categorical variable to a 0/1 encoding.
- Summarizing models. These invariably take as input a model produced by `lm()` (or `glm()`) and generate a summary report about that model.

  - **`coef()`**: displays model coefficients. Each coefficient is a single number.
  - **`conf_interval()`**: displays model coefficients as an *interval* with a lower and upper value.
  - **`rsquared()`** calculates the R^2 of a model, and some related measures.
  - **`regression_summary()`**, like `conf_interval()`, but with more detail.

- Evaluating a model on inputs
  - **`model_eval()`** takes a trained model (as produced by `lm()`) and calculates the model output in both a point form and an interval form. `model_eval()` can also display the residuals from training or evaluation data.

- Graphics
  - `model_plot()` draws a graphic of a model's function optionally with prediction or confidence intervals.
  - `geom_violin()` is a modern alternative to `geom_boxplot()`.

- DAGs (directed, acyclic graphs)
  - **`sample()`** collects simulated data from a DAG
  - `dag_draw()` draws a picture of a DAG showing how the variables are connected.

- Used within the `summarize()` data wrangling function:
  - **`var()`** computes the variance of a single variable.

Demonstration

Here are some of the command structures that appear in demonstrations. These explanations give a general idea of the tasks they perform.

- `do(10) * {` *command* `}` causes the *command* to be executed repeatedly the indicated number of times. Such repetitions are useful when the *command* is a trial of a random process such as sampling, resampling, or shuffling.

- `function(`*arguments*`) { `*set of commands*` }` packages in a single unit a set of one or more commands. The packaging facilitates using them over and over again with specified arguments.
- `geom_errorbar()` works much like `geom_point()` but draws vertical bars instead of dots. Bar-shaped glyphs depict *intervals* such as confidence or prediction intervals.
- `geom_ribbon()` is like `geom_line()` but for *intervals*.
- `effect_size()` calculates the strength and direction of the input-output relationship between the response variable of a model and a selected *one* of the explanatory variables.

Part I

1-18 from ModernDive

1 Bogus

Foobar

2 Bogus

Foobar

3 Bogus

Foobar

4 Bogus

Foobar

5 Bogus

Foobar

6 Bogus

Foobar

7 Bogus

Foobar

8 Bogus

Foobar

9 Bogus

Foobar

10 Bogus

Foobar

11 Bogus

Foobar

12 Bogus

Foobar

13 Bogus

Foobar

14 Bogus

Foobar

15 Bogus

Foobar

16 Bogus

Foobar

17 Bogus

Foobar

18 Bogus

Foobar

Part II

Statistical Thinking

19 Preliminaries

19.1 Statistical thinking

These lessons are about "**statistical thinking**," a phrase which includes habits of mind, routine questions to ask, and understanding of which statistical measures are informative—and which not—in different contexts. The goal of statistical thinking is to understand "how and when we can draw valid inferences from data." [Source] The word "valid" means several things at once: faithful to the data, consistent with the process used to assemble the data, and informative for the uses to which the inferences are to be directed.

Every person has a natural ability to think. We train our thinking skills by observing and emulating the logic and language of people and sources deemed authoritative. We have resources spanning several millennia to hone our ability to think. However, statistical thinking is a comparatively recent arrival on the intellectual scene, germinating and developing over only the last 150 years. As a result, hardly anything that we hear or read exemplifies statistical thinking.

In general, effective thinking requires us to grasp various intellectual tools, for example, logic. Our mode of logical thinking was promulgated by Aristotle (384–322 BC) and, to quote the Stanford Encyclopedia of Philosophy, "has had an unparalleled influence on the history of Western thought." In the 2500 years since Aristotle's time, the use of Aristotelian logic has been so pervasive that we expect any well-educated person to be able to identify logical thinking. For example, the statement "John's car is red" has implications. Which of these two statements are among those implications? "That red car is necessarily John's," or "The blue car is not John's car." Not so hard!

The intellectual tools needed for statistical thinking are, by and large, unfamiliar and non-intuitive. These Lessons are intended to provide the tools you will need to engage in effective statistical thinking.

To get started, consider this headline from *The Economist*, a well-reputed international news magazine: "The pandemic's indirect effects on small children could last a lifetime." As support for this claim, the headlined article provides more detail. For instance:

> "Stress and distraction made some patients more distant. LENA, a charity in Colorado, has for years used wearable microphones to keep track of how much chatter babies and the care-givers exchange. During the pandemic the number of such "conversations" declined. ...."[g]etting lots of interaction in the early years of life is essential for healthy development, so these kinds of data "are a red flag"." The article goes on to talk of "*children starved of stimulation at home ....*"

This short excerpt might raise some questions. Think about it briefly and note what questions come to mind.

For those already along the road toward statistical thinking, the phrase, "the number of such conversations declined" might prompt this question: "By how much?" Similarly, reading the claim that "getting lots of interactions ... is essential for healthy development," your mind might insist on these questions: How much is "lots?" How does the decline in the number compare to "lots?"

Not finding the answer to these questions in the article's text, it would be sensible to look for the primary source of the information. In our Internet age, that's comparatively easy to do. The LENA website includes an article, "COVID-era infants vocalize less and experience fewer conversational turns, says LENA research team." The article contains two graphs. (**?@fig-lena-two-graphs**)

```
knitr::include_graphics("www/Lena-fig1.png")
knitr::include_graphics("www/Lena-fig2.png")
```

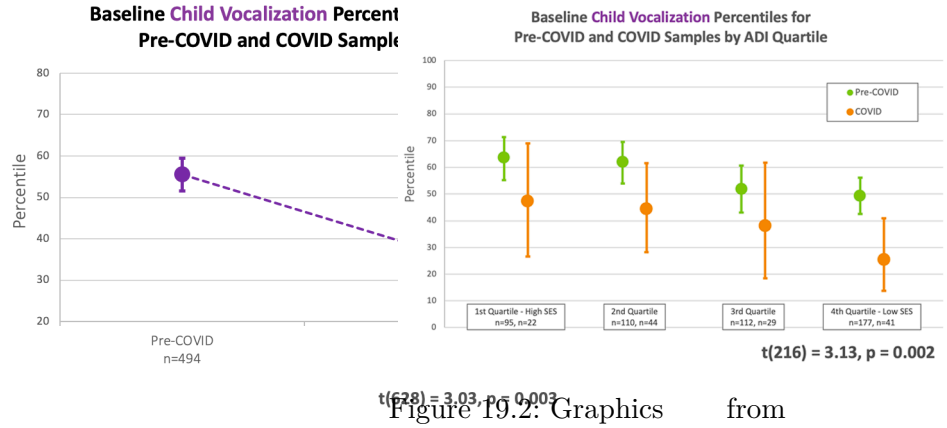


Figure 19.1: Graphics from the LENA website. The left is captioned, "Children from the COVID-era sample produced significantly fewer vocalizations than their pre-COVID peers." The right, "The differences in vocalizations and turns were greatest among children from families in the lowest SES [socio-economic status] quartile."

Figure 19.2: Graphics from the LENA website. The left is captioned, "Children from the COVID-era sample produced significantly fewer vocalizations than their pre-COVID peers." The right, "The differences in vocalizations and turns were greatest among children from families in the lowest SES [socio-economic status] quartile."

To make any proper sense of the graphs in **?@fig-lena-two-graphs**, you need some basic technical knowledge. For example, what do the vertical bars in the graph mean? And the subcaptions, "t(628) = 3.03, p = 0.003" and "t(216)= 2.13, p=0.002": What do they mean, if anything? Turning back to

the text of *The Economist*, do these graphs justify raising a "red flag?" More basically, are these graphs the "data," or is there more data behind the graphs? What would that data show?

The LENA article does not link to supporting data, that is, what lies behind the graphs in **?@fig-lena-two-graphs**. But the LENA article does point to other publications.

> "*These findings from LENA support a growing body of evidence that babies born during the COVID pandemic are, on average, experiencing developmental delays. For example, researchers from the COMBO (COVID-19 Mother Baby Outcomes) consortium at Columbia University published findings in the January 2022 issue of JAMA Pediatrics showing that children born during the pandemic achieved significantly lower gross motor, fine motor, and personal-social scores at six months of age.*"

To the statistical thinker, phrases like "red flag," "growing body of evidence," and "significantly lower" are **weasel words**, that is, terms "used in order to evade or retreat from a direct or forthright statement or position." [Source] In ordinary thinking, such evasiveness or lack of forthrightness would naturally prompt concern about the reliability of the claim. It makes sense to look deeper, for instance, by checking out the JAMA article. Many people would be hesitant to do this, anticipating that the article would be incomprehensible and filled with jargon. An important reason to study statistical thinking is to tear down barriers to substantiating or debunking claims. In fact, the JAMA article contains very little that requires knowledge of pediatrics or the meaning of "gross motor, fine motor, and personal-social scores," but a lot that depends on understanding statistical notation and convention and—more critical—the reasoning behind the conventions.

The tools of statistical thinking are the tools for making sense of data. Evaluating data is essential to determine whether to rely on claims supposedly based on those data. In the words of eminent engineer and statistician W. Edwards Demming: "In God we trust. All others must bring data." And former President

Ronald Reagan famously quoted a Russian proverb: "Trust, but verify." Unfortunately, until you have the statistical thinking tools needed to interpret data reliably, all you can do is trust, not verify.

19.2 Defining statistical thinking

Learning a new way of thinking is genuinely hard. As you learn statistical thinking, it may help to have a concise definition. The following definition captures much of the essence of statistical thinking:

> *Statistic thinking is the accounting for variation* in the context of *what remains unaccounted for.*

Implicit in this definition is a pathway for learning to think statistically:

1. Learn how to measure variation;
2. Learn how to account for variation;
3. Learn how to measure what remains unaccounted for.

The next three sections briefly touch on each of these three topics.

19.3 Variation

> *Variation itself is nature's only irreducible essence. Variation is the hard reality, not a set of imperfect measures for a central tendency. Means and medians are the abstractions.*
> —— Stephen Jay Gould (1941- 2002), paleontologist and historian of science.

To illustrate variation, let's consider a process fundamental to human life: gestation. We all know that human pregnancy "typically" lasts around nine-months, but that the duration isn't known in advance.

Figure **??** shows data from the `Gestation` data frame. In this data frame, each of the 1200 rows is one pregnancy and birth

about which several measurements were made. The `gestation` variable records the length of the pregnancy (in days).

```
Gestation <- Gestation %>%
  mutate(parity = ifelse(parity==0, "first-time", "previous-preg"))
Plot1 <- Gestation %>%
  ggplot(aes(x=parity, y=gestation)) +
  geom_jitter(alpha=0.2, width=0.2, height=0)
Plot1
```

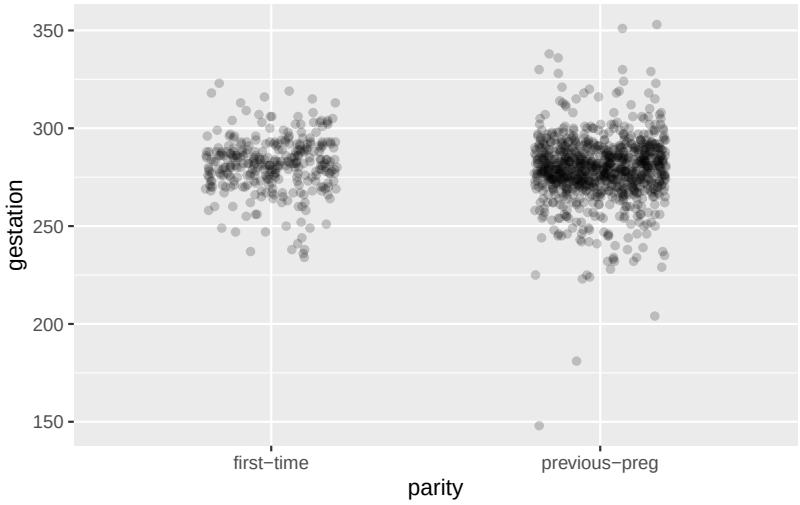


Figure 19.3: Gestational period for first-time mothers and mothers with a previous pregancy.

Figure **??** divides the 1200 births in the `Gestation` data frame according to the variable `parity`, which describes whether or not the pregnancy is the mother's first.

The variation in `gestation` is evident directly from the dots in the graph. One strategy for describing variation is to specify an **interval**: the span between a lower and an upper value. For instance,

- The large majority of pregancies last between 250 and 300 days. Or,
- The majority of pregnancies are between 275 and 290 days.

A more subtle description avoids setting hard bounds in favor of saying which durations are common and which not. This

common-or-not description is called a "**distribution**." The "**histogram**" is a famous style of presentation of a distribution. Even elementary-school students are introduced to histograms; they are easy to draw. But we have more important concerns; we want to be able to show relationships between variables and we want, whenever possible, to put the graphical summaries of data as a layer on top of the data themselves. And we have the computer as a tool for making graphics. Consequently, our preferred format for displaying distributions is a smooth shape, oriented along the vertical axis. The width of the shape expresses how common is the corresponding region of the vertical axis. Figure **??** shows the density display layered on top of the pregnancy data. For reasons that may be evident, this sort of display is called a "**violin plot**."

```
Plot1 +
  geom_violin(aes(group=parity), fill="blue", alpha=0.65, color=NA)
```

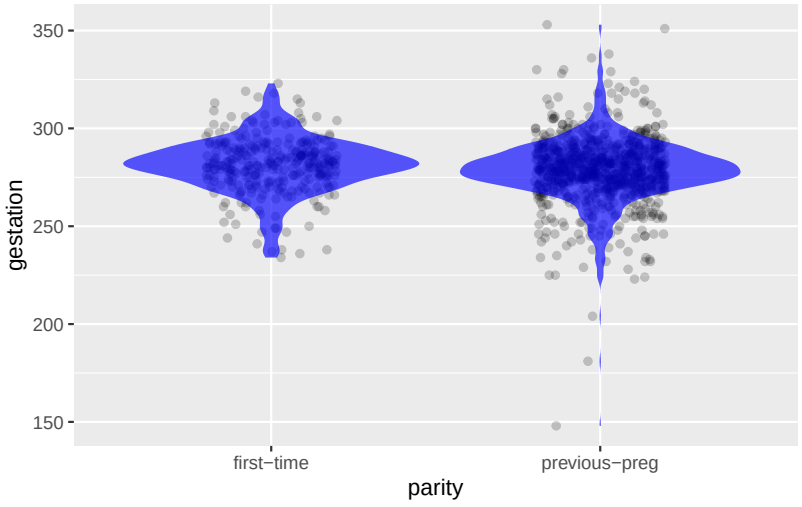


Figure 19.4: A violin plot. The long axis of the violin-like shape is oriented along the response-variable axis (that is, the vertical axis in our standard format). The width of the violin for each possible value of the response variable is proportional to the density of data near that value.

The shapes of the two violins in Figure **??** are similar, suggesting that the variation in the duration of pregnancy is about the same for first-time mothers as for mothers in a second or later pregnancy.

There is a strong link between the *interval* descriptions of variation and the density display. Suppose you specify the fraction of cases that you want to include in an interval description, say

50% or 80%. In terms of the violin, that fraction is a proportion of the overall **area** of the violin. For instance, the 50% interval would include the central 50% of the area of the violin, leaving 25% out at the bottom and another 25% out at the top. The 80% interval would leave out only 10% of the area at the top and bottom of the violin. This suggests that the interval style of describing variation really involves *three* numbers; the top and bottom of the interval *as well as* the selected percentage (say, 50% or 80%) used to find the location of the top and bottom.

Yet another style for describing variation—one that will take primary place in these Lessons—uses only a **single-number**. Perhaps the simplest way to imagine how a *single* number can capture variation is to think about the *spread* or *distance* between the top and bottom of an interval description. In taking such a distance as the measure of variation, we are throwing out some information. Taken together, the top and bottom of the interval describe two things: the *location* of the values and the *spread* among the values. These are both important, but it is the *spread* that gives a pure description of variation.

Early pioneers of statistics took some time to agree on a standard way of measuring the spread. For instance, should it be the spread between the top and bottom of a 50% interval or an 80% interval, or something else. In the end, the selected standard focussed on something more basic: the differences between pairs of individual values.

It works like this. For a data frame with $n = 2$ rows, the spread in a variable can be measured simply as the *difference* between the two values. For instance, suppose the `gestation` variable had only two entries, say, 267 and 293 days. The spread or distance between these is $293 - 267 = 26$ days. Of course, we don't intend to measure spread with a negative number. One solution is to use the absolute value of the difference. However, for subtle mathematical reasons relating to—of all things!—the Pythagorean theorem, we avoid the possibility of a negative spread by using the *square of the difference*, that is, $(293 - 267)^2 = 676$ days-squared.

To extend this very simple measure of variation to data with $n > 2$ is simple: look at the square difference between every pos-

sible pair of values, then average. For instance, for $n = 3$ with values 267, 293, 284, look at the differences $(267-293)^2$, $(267-284)^2$ and $(293-284)^2$ and average them! This simple way of measuring variation is called the "modulus" and dates from 1885. Since then, statisticians have standardized on a closely related measure, the "**variance**," which is the modulus divided by $\sqrt{2}$. Either one would work, but there are advantages to standardizing on one: the variance.

Calculating the variance is straightforward, Here's the variance of `gestation`:

```
Gestation %>%
  summarize(variance = var(gestation))
```

variance
256.887

A consequence of the use of squaring in defining the variance is the units of the result. `gestation` is measured in days, so `var(gestation)` is measured in days^2. The advantage to this will only become clear later in these Lessons. For now, you might prefer to think about the square-root of the variance, which has been given the name "**standard deviation**."

```
Gestation %>%
  summarize(standard_deviation = sd(gestation))
```

standard_deviation
16.02769

19.4 Accounting for variation

The word "account" has several related meanings.[1]

[1]These are drawn from the Oxford Languages dictionaries.

- To "account for something" means "to be the explanation or cause of something." [Oxford Languages]
- An "account of something" is a story, a description, or an explanation, as in the Biblical account of the creation of the world.
- To "take account of something" means "to consider particular facts, circumstances, etc. when making a decision about something."

Synonyms for "account" include "description,"report," "version," "story," "statement," "explanation," "interpretation," "sketch," and "portrayal." "Accountants" and their "account books" keep track of where money comes from and goes to.

These various nuances of meaning, from a simple arithmetical tallying up to an interpretation or version serve the purposes of statistical thinking well. When we "account for variation," we are telling a story that tries to explain where the variation might have come from. An accounting of variation is not necessarily definitive, true, or helpful. Just as witnesses of an event can have different accounts, so there can be many accounts of the variation even of the same variable in the same data frame.

There are many formats for stories, many ways of organizing facts and data, and many ways of accounting for variance. In these Lessons, we will use **regression modeling** almost exclusively as our method of accounting. Here, for example, are two different accounts of `gestation`:

```
lm(gestation ~ 1, data=Gestation) %>% coef()
```

```
(Intercept)
   279.3385
```

```
lm(gestation ~ parity, data = Gestation) %>% coef()
```

```
      (Intercept) parityprevious-preg
       281.261981           -2.585058
```

In the R language, expressions like `gestation ~ 1` and `gestation ~ parity` are called "tilde expressions." They are the means by which the modeler **specifies** the structure of the model that is to be built. Training (or "fitting") translates the **model specification** into an arithmetic formula that involves the explanatory variables and numerical coefficients.

The coefficients from a regression model are part of an accounting for variation. Learning how to read them is an important skill in statistical thinking. For instance, the coefficient from a model in the form `y ~ 1` is always the average value of variable `y`. In contrast, in a model like `y ~ x`, the "intercept" is a baseline value and the `x`-coefficient describes what part of the variation in `y` can be credited to `x`.

> **i** The RESPEX graphics format
>
> Figure **??** is an example of what we call the **RESPEX** graphics style. Each RESPEX graphic is made to coordinate with aregression model of the data. Every regression model has a response variable. Likewise, every RESPEX graphic shows the response variable on the vertical axis. Similarly, RESPEX graphics place an explanatory variable on the horizontal axis. If there is more than one explanatory variable, they are encoded graphically using color then faceting.
> RESPEX stands for "RESPonse versus EXplanatory," but you might like to think of it as data graphics drawn with "respect" to a model.

Regression models always have a quantitative response variable, although explanatory variables can be either quantitative or categorical. But, often, the modeling situation calls for a response variable that is *categorical*. Expert modelers can use specialized modeling methods to handle such situations. However, some of the power of these specialized methods is available to the beginning modeler by a little trick. When categorical response variables have just two levels, e.g., Alive/Dead, Promoted/Not, or Win/Loss, they can be transformed to a numerical representation using 0 for one level and 1 for the other.

We will identify the such variables as being of type "**yes/no**" or, equivalently, "**zero-one**" variables. With the zero-one encoding

This numerical "**0/1 encoding**" is directly suited for regression modeling and enables us to extend the scope of regression models. The *output* of the regression model is always numerical. Nothing in the regression technique restricts those outputs to exactly zero or one, even when the response variable is of the yes/no type. Usually, the modeler interprets such numerical output as probabilities or, more generally, as measures to be converted to probabilities.

R technique: `zero_one()`.

The `zero_one()` function converts a yes/no variable to the numerical zero-one format. `zero_one()` allows you to specify which of the two levels is represented by 1.
To illustrate, consider the `mosaicData::Whickham` data frame, which records a 1972-1974 survey, part of a study of the relationship between smoking and mortality. Twenty years after the initial survey, a follow-up established whether or not each person was still alive. Here are a few rows from the data frame:

outcome	smoker	age
Alive	Yes	23
Alive	Yes	18
Dead	Yes	71
Alive	No	67
Alive	No	64
Alive	Yes	38

The `outcome` variable in `Whickham` records the result of the follow-up survey. It is a categorical variable with levels "Alive" and "Dead." To examine what the data have to say about the relationship between smoking and mortality, we construct a model with `outcome` as the response variable and `smoking` as an explanatory variable. Before doing so, we translate `outcome` into a zero-one format. Like this:

```
Whickham %>%
  mutate(alive = zero_one(outcome, one="Alive"))
```

outcome	smoker	age	alive
Alive	Yes	23	1
Alive	Yes	18	1
Dead	Yes	71	0
Alive	No	67	1
Alive	No	64	1
Alive	Yes	38	1

Note the correspondence between the `outcome` and the newly created `alive` variable.

19.5 Variation unaccounted for

A model typically accounts for only some of the variation in a response variable. The remaining variation is called "**residual variation**."

Consider the model `gestation ~ parity`. In the next lines of code we build this model, training it with the `Gestation` data. Then we **evaluate** the model on the trained data. This amounts to using the model coefficients to generate a model output for each row in the training data, and can be accomplished with the `model_eval()` R function.

```
Model <- lm(gestation ~ parity, data = Gestation)
Evaluated <- model_eval(Model)
```

```
Using training data as input to model_eval().
```

```
Using training data as input to model_eval().
```

	.response	parity	.output	.resid	.lwr	.upr
1218	270	previous-preg	278.6769	-8.6769231	247.2800	310.0738
1219	275	first-time	281.2620	-6.2619808	249.8322	312.6917
1220	265	previous-preg	278.6769	-13.6769231	247.2800	310.0738
1221	291	previous-preg	278.6769	12.3230769	247.2800	310.0738
1222	281	first-time	281.2620	-0.2619808	249.8322	312.6917
1223	297	previous-preg	278.6769	18.3230769	247.2800	310.0738

i The `.response` variable

The output from `model_eval()` repeats some columns from the data used for evaluation. For example, the explanatory variables are listed by name. (Here, the only explanatory variable is `parity`.) The response variable is also included, but given a generic name, `.response` to make it easy to distinguish it from the explanatory variables.

To see where the `.output` comes from, let's look again at the model coefficients:

```
Model %>% coef()
```

```
       (Intercept) parityprevious-preg
         281.261981           -2.585058
```

The baseline value is 281.3 days. This applies to first-time mothers. For the other mothers, those with a previous pregnancy, the coefficient indicates that the model value is 2.6 days *less* than the baseline, or 279.7 days.

The output from `model_eval()` includes other columns of importance. For us, here, those are. the response variable itself (`gestation`, which has been given a generic name, `.response`) and the residuals from the model (`.resid`). There is a simple relationship between `.response`, `.output` and `.resid`:

$$\texttt{.response} = \texttt{.output} + \texttt{.resid}$$

Demonstration: Why the variance?

The subtle mathematical reasoning behind the choice of *variance* to measure variation is illuminated when we compute the variances of the three quantities in the previous equation.

```
Evaluated %>%
  summarize(var_response = var(.response),
            var_output = var(.output),
            var_resid  = var(.resid))
```

var_response	var_output	var_resid
256.887	1.273587	255.6134

The variances of the output and residuals add up to equal, exactly, the variance of the response variable! This isn't true for the standard deviations:

```
Evaluated %>%
  summarize(sd_response = sd(.response),
            sd_output = sd(.output),
            sd_resid  = sd(.resid))
```

sd_response	sd_output	sd_resid
16.02769	1.128533	15.98791

20 Simulation and sampling variation

Carl Wieman is a Nobel-prize-winning physicist and professor of education at Stanford University. Weiman writes, "For many years, I had two parallel research programs: one blasting atoms with lasers to see how they'd behave, and another studying how people learn." Some of Wieman's work on learning deals with the nature of "**expertise**." He points out that experts have ways to monitor their own thinking and learning; they have a body of knowledge relevant to checking their own understanding.

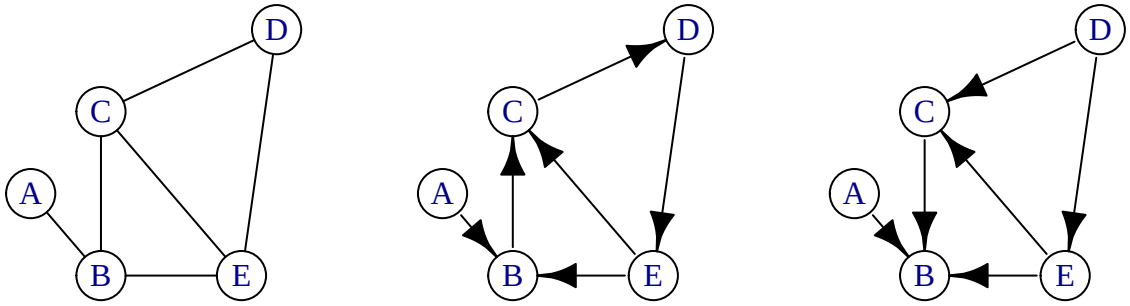
This lesson presents you with two tools: simulation and repetition. Simulation enables you to generate variation with known properties, where you know the sources of variation and how variables are connected. Then you can experiment to find out how to use statistical models to reveal the underlying properties.

The second tool, repetition, helps you deal with randomness and quantify precision. By repeating the same simulation many times while introducing basic random fluctuations, you can figure out the extent to which randomness can be tamed.

20.1 Directed Acyclic Graphs

A core tool in thinking about causal connections is a mathematical structure called a "directed acyclic graph" (DAG, for short). DAGs are one of the most popular ways for statistical thinkers to express their ideas about what might be happening in the real world. Despite the long name, DAGs are very accessible to a broad audience.

DAGs, despite the G for "graph," are not about data graphics. The "graph" in DAG is a mathematical term of art; a suitable synonym is "network." Mathematical graphs consist of a set of "nodes" and a set of "edges" connecting the nodes. For instance, Figure **??** shows three different graphs, each with five nodes labeled A, B, C, D, and E.



(a) undirected graph (b) directed but cyclic (c) directed acyclic graph

Figure 20.1: Graphs of various types

The nodes are the same in all three graphs of Figure **??**, but each graph is different from the others. It is not just the nodes that define a graph; the edges (drawn as lines) are part of the definition as well.

The left-most graph in Figure **??** is an "**undirected**" graph; there is no suggestion that the edges run one way or another. In contrast, the middle graph has the same nodes and edges, but the edges are **directed**. An excellent way to think about a directed graph is that each node is a pool of water; each directed edge shows how the water flows between pools. This analogy is also helpful in thinking about causality: the causal influences flow like water.

Look more carefully at the middle graph. There is a couple of loops; the graph is **cyclic**. In one loop, water flows from E to C to D and back again to E. The other loop runs B, C, D, E, and back to B. Such a flow pattern cannot exist without pumps pushing the water back uphill.

The rightmost graph reverses the direction of some of the edges. This graph has no cycles; it is **acyclic**. Using the flowing and pumped water analogy, an acyclic graph needs no pumps; the

pools can be arranged at different heights to create a flow exclusively powered by gravity. The node-D pool will be the highest, E lower. C has to be lower than E for gravity to pull water along the edge from E to C. The node-B pool is the lowest, so water can flow in from E, C, and A.

Directed acyclic graphs represent causal influences; think of "A causes B," meaning that causal "water" flows naturally from A to B. In a DAG, a node can have multiple outputs, like D and E, and it might have multiple inputs, like B and C. In terms of causality, a node—like B—having multiple inputs means that more than one factor is responsible for the value of that node. A real-world example: the rising sun causes a rooster to crow, but so can another intruder to the coop.

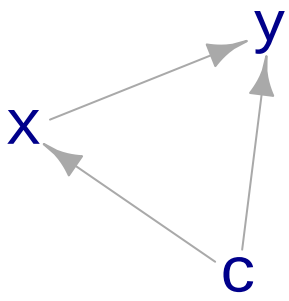
Often, nodes do not have any inputs. These are called "**exogenous factors**"at least by economists. The "genous" means "originates from." "Exo" means "outside." The value of an exogenous node is determined by something, just not something that we are interested in (or perhaps capable of) modeling. No edges are directed into an exogenous node since none of the other nodes influence its value.

For simulating data, we go beyond drawing a graph of causal connections to outfit DAGs with specific formulas representing the mechanism imbued in each node. DAGs equipped with formulas can be used to generate simulated data.[1] Training a model on those data leads to a model function that we can compare to the DAG's formulas. Then check whether the formulas and the model function match. This practice helps us learn what can go right or wrong in building a model, just as practice in an aircraft simulator trains pilots to handle real-world situations in real aircraft.

We start with a simple example, `dag08`. The `dag_draw()` command draws a picture of the graph. Printing the dag displays the formulas that set the values of the nodes.

```
dag_draw(dag08)
```

[1]The value of exogenous nodes is usually set randomly, without input from the other nodes in the DAG.



The graph shows that both `c` and `x` contribute to `y`.

```
print(dag08)
```

```
c ~ exo()
x ~ c + exo()
y ~ x + c + 3 + exo()
```

The formulas show that `x` and `c` contribute equally to `y`, with coefficients of 1. To what extent can regression modeling recover this relationship from data?

To find out, we can generate simulated data using the `sample()` function. For instance,

```
sample(dag08, size=5)
```

c	x	y
-0.3260365	0.8479298	4.048341
0.5524619	1.1712517	3.928869
-0.6749438	-0.7876782	2.965133
0.2143595	1.1313877	2.878928
0.3107692	0.0875099	3.161596

Each row in the sample is one trial; in each trial, the node's formula sets the value for that node. For example, the formula might use the values of other nodes as input. Alternatively, the formula might specify that the node is exogenous, without input from any other nodes.

Models can be trained on the simulated data using the same techniques as for any other data. To illustrate, here we generate a sample of size $n = 50$, then fit the model specification `c ~ a + b` and summarize by taking the coefficients.

```
sample(dag08, size=50) %>%
  lm(y ~ c + x, data = .) %>%
  coef()
```

```
(Intercept)           c           x
  2.9451445   1.2606473   0.8235923
```

The coefficients, including the intercept, are close, but not exactly right.

In Lessons -Chapter **??** and -Chapter **??** we will figure out how close we can expect the coefficients to be to the precise values implemented in the simulation.

20.2 Samples, summaries of samples, and samples of summaries (of samples)

Beginners sometimes think that each row in a data frame is a sample. Better to say that each row is a "**specimen**." A "**sample**" is a collection of specimens, the set of rows in a data frame.

The "**sample size**" is the number of rows. "**Sampling**" is the process of collecting the specimens to be put into the data frame.

The following command illustrates computing a summary of a sample from `dag08`.

```
sample(dag08, size=10000) %>%
  lm(y ~ c + x, data = .) %>%
  coef()
```

```
(Intercept)           c           x
  3.0070253   1.0100177   0.9934592
```

An essential question in statistics is how the summary depends on the incidental specifics of a particular sample. DAGs provide a convenient way to address this question since we can generate multiple samples from the same DAG, summarize each, and compare those summaries.

To generate a sample of summaries, re-run many trials of the summary. The `do()` function automates this process, accumulating the results from the trials in a single data frame: a "**sample of summaries**." We will use `do()` mostly in demonstrations.

⚠ Demonstration: Conducting many trials with `do()`

In this demonstration, we will revisit a model used earlier in this Lesson to see how much the coefficients vary from one sample to another. Each trial consists of drawing a sample from `dag08`, training a model, and summarizing with the model coefficients. Curly braces (`{` and `}`) surround the commands needed for an individual trial. Preceding the curly braces, we have placed `do(5) *`. This instruction causes the trial to be repeated five times.

```
do(5) * {
  sample(dag08, size=50) %>%
    lm(y ~ c + x, data = .) %>%
    coef()
}
```

Intercept	c	x
3.019112	0.6794641	1.3353393
3.006728	0.9042066	0.8406397
2.966061	1.1619847	0.9307029
2.866499	1.0881640	1.0769612
3.080889	1.1088753	1.0009938

The five trials are collected together by `do()` into the five

rows of a single data frame. Such a data frame can be considered a "**sample of summaries**."

One of the things we will do with a "sample of summaries" is to … wait for it … summarize it. For instance, in the following code chunk, a sample of 40 summaries is stored under the name `Trials`. Then we will summarize `Trials`, in this case, to see how much the values of the `a` and `b` coefficients vary from trial to trial.

```
Trials <- do(40) * {
  sample(dag08, size=50) %>%
    glm(y ~ c + x, data = .) %>%
    coef()
}
Trials %>%
  summarize(mean_c_coef = mean(c), spread_a = sd(c),
            mean_x_coef = mean(x), spread_b = sd(c))
```

mean_c_coef	spread_a	mean_x_coef	spread_b
0.9858736	0.2215985	1.022228	0.2215985

The result of summarizing the trials is a "summary of a sample of summaries." This phrase is admittedly awkward, but we will use this technique often: summarizing trials, where each trial is a "summary of a sample" Often, the clue will be the use of `do()`, which repeats trials as many times as you ask.

20.3 Causal inference

Often, but not always, our interest in studying data is to reveal or exploit the causal connections between variables. Understanding causality is essential, for instance, if we are planning to intervene in the world and want to anticipate the consequences. Interventions are things like "increase the dose of medicine," "stop smoking!", "lower the budget," "add more cargo to a plane (which will increase fuel consumption and reduce the range)."

Historically, mainstream statisticians were hostile to using data to explore causal relationships. (The one exception was **experiment**, which gathers data from an actual intervention in the world. See Lesson **??**.) Statistics teachers encouraged students to use phrases like "associated with" or "correlated with" and reminded them that "correlation is not causation."

Regrettably, this attitude made statistics irrelevant to the many situations where intervention is the core concern and experiment was not feasible. A tragic episode of this sort likely caused millions of unnecessary deaths. Starting in the 1940s, doctors and epidemiologists saw evidence that smoking causes lung cancer. In stepped the most famous statistician of the age, Ronald Fisher, to insist that the statement should be, "smoking is associated with lung cancer." He speculated that smoking and lung cancer might have a common cause, perhaps genetic. Fisher argued that establishing causation requires running an experiment where people are randomly assigned to smoke or not smoke and then observed for decades to see if they developed lung cancer. Such an experiment is unfeasible and unethical, to say nothing of the need to wait decades to get a result.

Fortunately, around 1960, a researcher at the US National Institutes of Health, Jerome Cornfield, was able to show mathematically that the strength of the association between smoking and cancer ruled out any genetic mechanism. Cornfield's work was an important step in the development of a new area in statistics: "**causal inference**."

Causal inference is not about proving that one thing causes another but about formal ways to say something about how the world works that can be used, along with data, to make responsible conclusions about causal relationships.

As you will see in Lesson -Chapter **??**, DAGs are a major tools in causal inference, allowing you not only to represent a hypothesis about causal relationships, but to deduce what sorts of models will be able to reveal causal mechanisms.

The point of a DAG is to make a clear statement of a hypothesis about causation. Drawing a DAG does not mean that the hypothesis is correct, just that we believe the hypothesis is, in some sense, a possibility. Different people might have different

beliefs about what causes what in real-world systems. Comparing their different DAGs can help, sometimes, to discuss and resolve the disagreement.

We are going to use DAGs for two distinct purposes. One purpose is to inform responsible conclusions from data about what causes what. The data on its own is insufficient to demonstrate the causal connections. However, data *combined with* a DAG can tell us something. Sometimes a DAG includes a causal connection that should create an association between variables. The DAG is incomplete if the association does not appear in the data.

DAGs are also valuable aids for building models. For example, analysis of the paths in a DAG, as in Lesson **??**, can tell us which explanatory variables to include and which to exclude from a model if our modeling goal is to represent the hypothetical causal connections.

In these Lessons, we have a second, entirely different, use for DAGs: learning modeling technique. Our approach will be to ::: {.callout-warning} ## Reality check: DAGs and data

DAGs represent hypotheses about the connections between variables in the real world. They are a kind of scratchpad for constructing alternative scenarios and, as seen in Lesson **??**, thinking about how models might go wrong in the face of a plausible alternative causal mechanism.

In this book, we extend the use of DAGs beyond their scope in professional statistics; we use them as simulations from which we can generate data. Such simulations provide one way to learn about statistical methodology.

DAGs are aides to reasoning, scratchpads that help us play out the consequences of our hypotheses about possible real-world mechanisms. However, take caution to distinguish data from DAG simulations from data from reality.

Finding out about the real world requires collecting data from the real world. The proper role of DAGs in real work is to guide model building **from real data**.

In this course, we sample from DAGs to learn statistical techniques. But never to make claims about real-world phenomena.
:::

21 Signal and noise

Imagine being transported back to June 1940. The family is in the living room, sitting around the radio console, waiting for it to warm up. The news today is from Europe, the surrender of the French in the face of the German invasion. Press the play button and listen to recording #103.

You may have to scroll down to see the play button and the recordings.

The spoken words from the recording are discernible despite the hiss and clicks of the background noise. The situation is similar to a conversation in a sports stadium. The crowd is loud, so the speaker has to shout. The listener ignores the noise (unless it is too loud) and recovers the shouted words.

Engineers and others make a distinction between **signal** and noise. The engineer aims to separate the signal from the noise. That aim applies to statistics as well.

There are many sources of noise in data; every variable has its own story, part of which is noise from measurement errors and recording blunders. For instance, economists use national statistics, like GDP, even though the definition is arbitrary (a Hurricane can raise GDP!), and early reports are invariably corrected a few months later. Historians go back to original documents, but inevitably many of the documents have been lost or destroyed: a source of noise. Even in elections where, in principle, counting is straightforward, the voters' intentions are measured imperfectly due to "hanging chads," "butterfly ballots," broken voting machines, spoiled ballots, and so on.

The statistical thinker is well advised to know about the sources of noise in the system she is studying. Analysis of data will be better the more the modeler knows about how measurements are made and data collected.

Noise in hiring

The author has, on several occasions, testified in legal hearings as a statistical expert. In one case, the US Department of Labor audited the records of a contractor with several hundred employees and high employee turnover. The records led the Department to bring suit against the contractor for discriminating against Hispanics. The hiring records showed that many Hispanics applied for jobs; the company hired none. An open-and-shut case.

The lawyers for the defense asked me, the statistical expert, to review the findings from the Department of Labor. The lawyers thought they were asking me to check the arithmetic in the hiring spreadsheets. As a statistical thinker, I know that arithmetic is only part of the story; the origin of the data is critically important. So I asked for the complete files on all applicants and hires the previous year.

The spreadsheet files and the paper job applications were in accord; there were many Hispanic applicants. But the data on the paper job application form was not always consistent with the data on hiring spreadsheets. It turned out that whenever an applicant was hired, the contractor (per regulation) got a report on that person from the state police. The report returned by the state police had only two available race/ethnicities: white and Black. The contractor's personnel office filled in the hired-worker spreadsheet based on the state police report. So all the Hispanic applicants who were hired had been transformed into white or Black by the state police. Noise.

21.1 Signal and noise

To illustrate the statistical problem of signal and noise, let us turn to a DAG simulation: `dag01`. Here's a sample from `dag01`:

```
Tiny <- sample(dag01, size=2)
```

x	y
-0.3260365	2.836001
0.5524619	5.043052

The DAG simulation implements a relationship between `x` and `y`. In statistics, this *relationship* is the signal.

> Look at the 2-row sample (**?@tbl-tiny-dag01**) from the DAG and guess what the relationship might be.

Any of an infinite number of possible relationships could account for the `x` and `y` data. The noise reduction problem of statistics is to make a guess that is as good as possible. Unfortunately, for a sample with $n = 2$, as "good as possible" is not very good!

More data—a bigger sample—gives us a better shot at revealing the relationship hidden by the noise. **?@tbl-small-dag01** shows a sample of size $n = 10$:

```
Small <- sample(dag01, size=10)
```

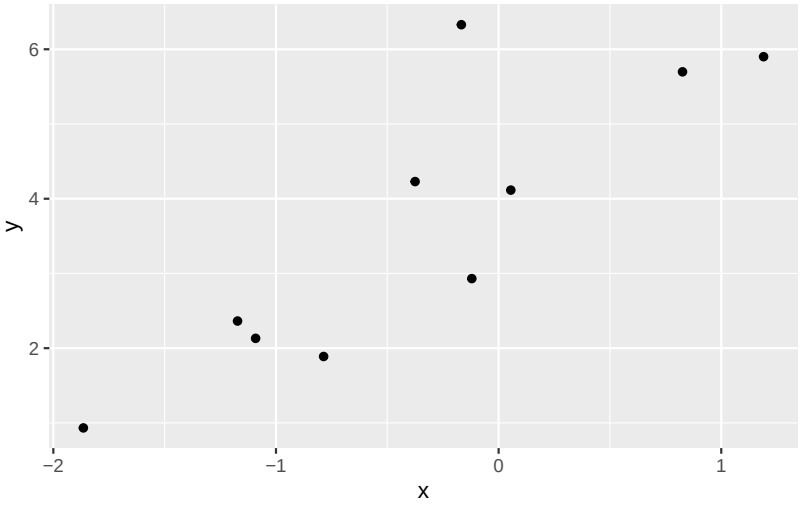
x	y
-0.7859732	1.8888204
0.0547389	4.1153256
-1.1725603	2.3632792
-0.1673128	6.3287614
-1.8650316	0.9329524
-0.1204402	2.9310384
0.8259787	5.6981878
1.1901595	5.9006170
-1.0914519	2.1314570
-0.3751124	4.2296648

A careful perusal of the `Small` sample suggests some patterns. `x` is never larger than about 2 in magnitude and can be positive or

negative. `y` is always positive. Furthermore, when `x` is negative, the corresponding `y` value is relatively small compared to the `y` values for positive `x`.

A sample of size $n = 10$ provides more information than a sample of $n = 2$, so we can make a more informed guess about the relationship between variables `x` and `y`.

Human cognition is not well suited to looking at long columns of numbers. Often, we can make better use of our natural human talents by translating the sample into a graphic:



Collecting more data can make the relationship clearer. Figure **??** displays an $n = 10,000$ sample.

```
Large <- sample(dag01, size=10000)
```

There are many possible ways to describe the x-y relationship in Figure **??**. For instance, we can see that when `x` is positive, `y` is almost always greater than 4, but for negative `x`, the value of `y` tends to be less than 4. Such a description might be apt for some purposes, but in these Lessons, we describe relationships by fitting models to data.

The following command uses the small sample (n=10) as training data for a model `y ~ x` that accounts for `y` on the basis of `x`:

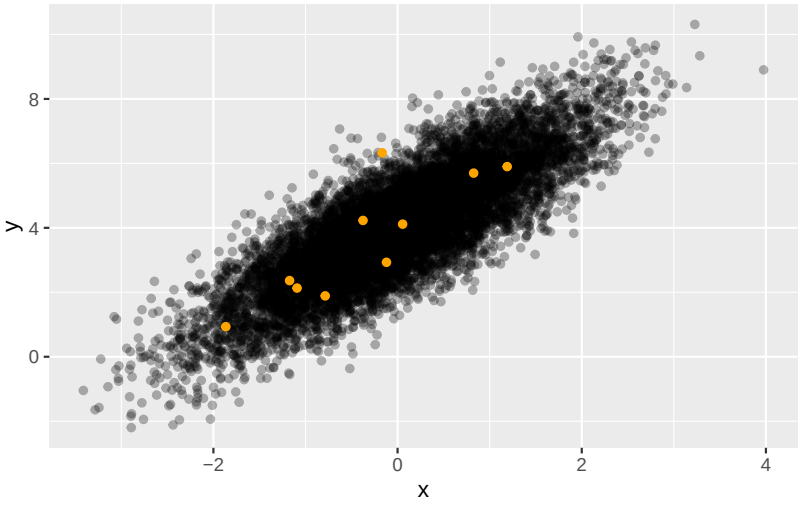


Figure 21.1: With $n = 10,000$ rows, the relationship between `x` and `y` is evident graphically. (The original `Small` sample is shown in orange.)

```
lm(y ~ x, data = Small) %>% coef()  # n = 10 sample
```

```
(Intercept)           x
   4.262846    1.741758
```

The coefficients provide the information needed to construct the *model function*:

$$y = 4.26 + 1.74x \ .$$

This mathematical formula is a guess of the *signal*—the relationship between the two variables in `dag01`. Unfortunately, the formula tells us nothing about the noise obscuring the signal nor how good the guess is.

The model coefficients produced by training the model on a much larger sample will presumably be a better guess:

```
lm(y ~ x, data = Large) %>% coef() #  n = 10,000 sample
```

```
(Intercept)           x
   4.008928    1.495904
```

Unfortunately, we cannot tell from the coefficients how good the guess is.

Luckily for us, since the data are a simulation from a DAG, we can see what the coefficients *should be* as well as the origin of the noise mixing in with the signal.

```
print(dag01)
```

```
x ~ exo()
y ~ 1.5 * x + 4 + exo()
```

The `Large` sample produced coefficients much closer than the `Small` sample to the mechanism in the DAG. The idea that larger samples lead to better accuracy has been appreciated since the 16th century and now has the prestige of being a "Law": the Law of Large Numbers.

However, "better accuracy" does not tell us whether the accuracy suffices for any given purpose. The model filters out some of the noise. However, the model coefficients still display a noisy legacy.

The challenge of real-world data is that we cannot open the black box that generated the data; all we have is the data! So how can we tell whether the data at hand are sufficient for giving a usefully accurate description of the actual relationships?

The key to the puzzle is the variation *within* the sample.

21.2 Measuring variation

Lesson **??** introduced the standard way to measure variation in a single variable: the **variance** or its square root, the **standard deviation**. For instance, we can measure the variation in the variables from the `Large` sample using `sd()` and `var()`:

```
Large %>%
  summarize(sx = sd(x), sy = sd(y), vx = var(x), vy = var(y))
```

sx	sy	vx	vy
0.9830639	1.779003	0.9664146	3.164851

According to the standard deviation, the size of the `x` variation is about 1. The size of the `y` variation is about 1.7.

Look again at the formulas that compose `dag01`:

```
print(dag01)
```

```
x ~ exo()
y ~ 1.5 * x + 4 + exo()
```

The formula for `x` shows that `x` is endogenous, its values coming from a random number generator, `exo()`, which, unless otherwise specified, generates noise of size 1.

As for `y`, the formula includes two sources of variation:

1. The part of `y` determined by `x`, that is $y = \mathbf{1.5x} + 4 + \text{exo}()$
2. The noise added directly into `y`, that is $y = 1.5\mathbf{x} + 4 + \mathbf{exo}()$

The 4 in the formula does not add any *variation* to `y`; it is just a number.

We already know that `exo()` generates random noise of size 1. So the amount of variation contributed by the `+ exo()` term in the DAG formula is 1. The remaining variation is contributed by `1.5 * x`. The variation in `x` is 1 (coming from the `exo()` in the formula for `x`). A reasonable guess is that `1.5 * x` will have 1.5 times the variation in `x`. So, the variation contributed by the `1.5 * x` component is 1.5. The overall variation in `y` is the sum of the variations contributed by the individual components. This suggests that the variation in `y` should be

$$\underbrace{1}_{\text{from exo()}} + \underbrace{1.5}_{\text{from 1.5 x}} = \underbrace{2.5}_{\text{overall variation in y}}.$$

Simple addition! Unfortunately, the result is wrong. In the previous summary of the `Large`, we measured the overall variation in `y` as about 1.72.

The *variance* will give a better accounting than the standard deviation. Recall that `exo()` generates variation whose standard deviation is 1, so the variance from `exo()` is $1^2 = 1$. Since `x` comes entirely from `exo()`, the variance of `x` is 1. So is the variance of the `exo()` component of `y`.

Turn to the `1.5 * x` component of `y`. Since variances involve squares, the variance of `1.5 * x` works out to be $1.5^2 \text{ var}(x) = 2.25$. Adding up the variances from the two components of `y` gives

$$\text{var}(y) = \underbrace{2.25}_{\text{from 1.5 exo()}} + \underbrace{1}_{\text{from exo()}} = 3.25$$

This result that the variance of `y` is 3.25 closely matches what we found in summarizing the `y` data generated by the DAG.

The lesson here: When adding two sources of variation, the variances of the individual sources add to form the overall variance of the sum. Just like $A^2 + B^2 = C^2$ in the Pythagorean Theorem.

21.3 DAGs from data

In modeling data from `dag01` we could recover a good approximation to the formula for `y`.

```
Large %>%
  lm(y ~ x, data = .) %>%
  coef()
```

```
(Intercept)           x
   4.008928    1.495904
```

A DAG describes the causal links between variables. Data modeling reveals the formula implementing the causal link in `dag01`. Nevertheless, it is wrong to think we can determine the DAG that generated the data from the data alone. Only if we already know the structure of the data-generation DAG can we recover the mechanism inside that DAG. For instance, another statistical thinker might believe that the causal mechanism behind the data is `y` causing `x`. Based on this assumption, she also can find the mechanism inside her hypothesized DAG:

```
sample(dag01, size=10000) %>%
  lm(x ~ y, data = .) %>%
  coef()
```

```
(Intercept)           y
 -1.8261448   0.4559782
```

A DAG is a **hypothesis**, a statement that might or might not be true. DAGs are part of the statistical apparatus for thinking responsibly about **causality**. Use a DAG—or, potentially, multiple DAGs—when the issue of what causes what is relevant to the purpose behind the work.

When there are only two variables involved in the system under consideration—we will call them X and Y for simplicity—there are only two possible DAGs:

$$X \to Y \quad \text{and} \quad X \leftarrow Y$$

Our understanding of the world sometimes allows us to focus on one of these and not the other. Example: Does the rooster crowing cause the sun to rise, or does the rising sun cause the rooster to crow?

Beyond the two DAGs $X \to Y$ and $X \leftarrow Y$, additional DAG possibilities can account for the relationship between X and Y. For instance, if we introduce another variable, C, located between X and Y, four other DAGs need to be considered:

$$X \to C \to Y \quad \text{and} \quad X \leftarrow C \leftarrow Y \quad \text{and} \quad X \leftarrow C \to Y \quad \text{and} \quad X \to C \leftarrow Y$$

There are many other DAG configurations involving three variables. To keep things simple, we will restrict things to DAGs where X might or might not cause Y, but Y never causes X.[1] Figure **??** shows the ten configurations of 3-variable DAGs where Y does not cause X.

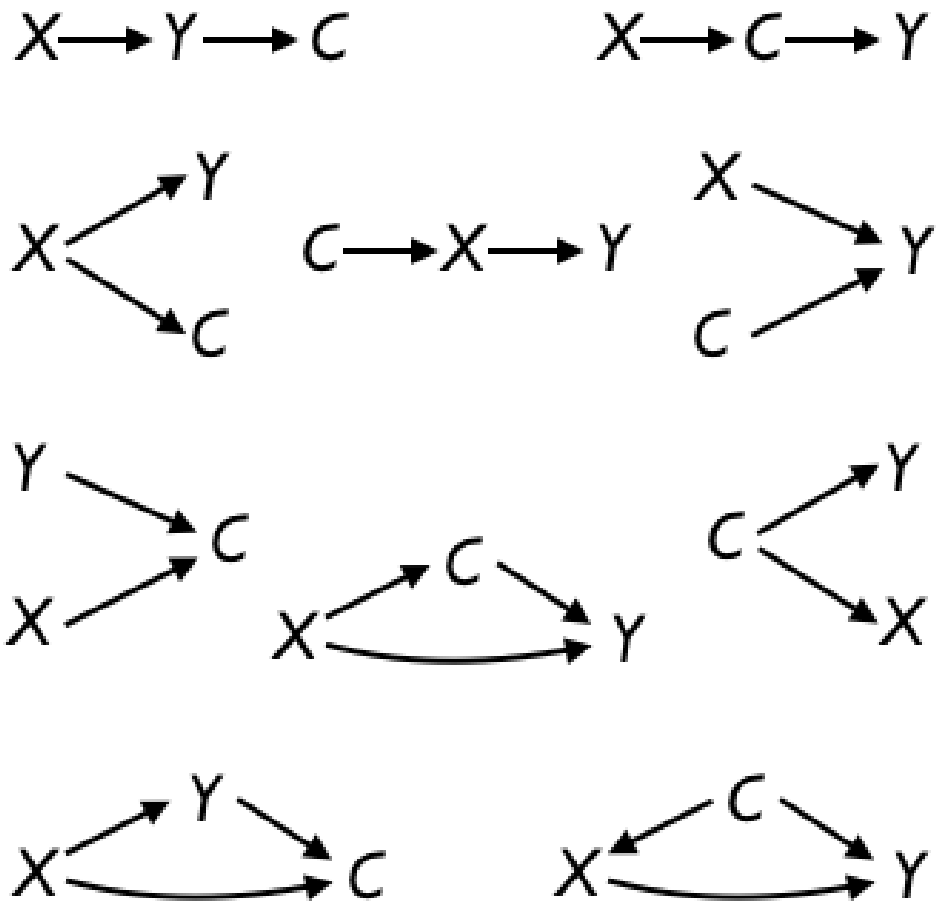


Figure 21.2: Ten DAG configurations involving three variables X, Y, and C.

With the conceptual tool of DAGs, the statistical thinker can consider multiple possibilities for what might cause what. Sometimes she can discard some of the possibilities based on common sense. (Think: roosters and the sun.) However, in other settings, there may be possibilities that she does not favor but might be plausible to other people. In Lesson **??**, we

[1] We do not lose generality by this restriction. The modeler gets to choose which real-world variable corresponds to X and which one to Y.

will explore how each configuration of DAG has implications for which model specifications can or cannot reveal the hypothesized causal mechanism.

22 Sampling and sampling variation

Sampling variation is a subtle concept. Part of the difficulty in understanding sampling variation is the meaning of the word "sample," which differs in use between everyday speech and statistical language. In everyday speech, a sample is:

> "*A small part or quantity intended to show what the whole is like.*" — Oxford Languages

A food market will give you a sample of an item on sale: a tiny cup of a drink or a taste of a piece of fruit or other food item. Laundry-detergent companies sometimes send out a sample of their product in the form of a small foil packet suitable for only a single wash cycle. Paint stores keep small samples on hand to help customers choose from among the possibilities. A fabric sample is a little swatch of cloth cut from a a bigger bolt that a customer is considering buying.

In contrast, a **sample** in statistics is always a *collection* of multiple items. Usually, a sample is presented to us in the form of a data frame which records the measured attributes of each of the items in the sample. While it is possible for a data frame to have just a single row, it is perverse to use the word "sample" to describe a single item. Instead, we use other words to point to a single item: for instance "a case," "a row," "an individual," "a datum," or "a specimen." Samples, like the word "data," are always plural. Think of "sample" as akin to words like "herd," "flock", "pack", or "school": a collective. A single fish is not a school and a single wolf is not a pack. Similarly, a single row is not a sample but an item.

The dictionary definition of "sample" uses the word "whole" to describe where the sample comes from. Similarly, a statistical sample is a collection of items selected from a larger "whole."

Traditionally, statisticians have used the word "**population**" as the name for the "whole." This is a nice metaphor; it's easy to imagine the population of a state being the source of a sample in which each individual is a specific person. But the "whole" from which a sample is collected does not need to be a finite, definite set of individuals like the citizens of a state. For example, you have already seen how to collect a sample of any size you want from a DAG.

Our *modus operandi* in these Lessons takes a sample in the form of a data frame and summarizes it in the form of one or more numbers. (Typically, the numbers are the coefficients of a regression model, but it might be something else such as the mean or variance of a variable.) Each such number is called a "**sample statistic**," but we think "**sample summary**" is a less confusing term and what we will use for these Lessons.

Practical statistical work almost always involves working with a single sample of size n. As a thought experiment, however, we can imagine having multiple samples, each collected independently and at random from the same source. Now picture a process for computing a sample summary, say, a regression coefficient for a particular model specification. If we apply that same process to each of our imagined samples, we will likely get equivalent sample summaries that differ one from another. Such sample-to-sample differences are called "**sampling variation**."

In this Lesson, we will simulate such a process of computing equivalent sample summaries from a set of samples. That way, we can see sampling variation directly.

In actual work with data, as opposed to simulations designed to illustrate statistical concepts, there is only one sample. We cannot see sampling variation directly in a single sample. But, that does not mean we can ignore the theoretical possibility.

Usually, we study a sample in order to inform our understanding of the broader process that generated the sample. Or, in the words of the dictionary definition at the start of this Lesson, we use a sample "*to show what the whole is like.*" Because of sampling variation, it would not be correct to say the "whole" is exactly like our sample. By quantifying sampling variation,

we give a more complete description of the relationship of our particular sample to the "whole."

> **i** Sampling distribution and sampling variance
>
> We have already gotten into the habit of illustrating the row-to-row variation within a variable with a violin plot. The shape is a picture of the **distribution** of that variable.
> In this Lesson, we will use simulation to generate many independent samples and the sample summary that goes along with each of those samples. The resulting varying set of numbers has, like any other variable, a distribution. Since the variation stems from sample-to-sample differences, we call it the "**sampling distribution**." But this sampling distribution is a theoretical thing: what we *would have gotten* if we had collected many samples. Still, it will be a useful theoretical thing.
> The obvious way to quantify the spread in the sampling distribution is—as usual—the variance. We will call this the "**sampling variance**."
> It's important to note the "ing" ending in "sampling variance" and "sampling distribution." Whereas the "sample variance" is the row-by-row variance calculated on a variable from a single sample, the "sampl**ing** variance" stems from the theoretical sample-by-sample variation.

22.1 Why sample?

Sometimes a data frame is not a sample. This happens when the data frame contains a row for every member of an actual, finite "population." Such a complete enumeration—the inventory records of a merchant, the records kept of student grades by the school registrar—has a technical name: a "**census**." Famously, many countries conduct a census of the population in which they try to record every resident of the country. For example, the US, UK, and China carry out a census every ten years.

In a typical setting, it is unfeasible to record every possible unit of observation.[1] Such incomplete records constitute a "**sample**." One of the great successes of statistics is the means to draw useful information from a sample, at least when the sample is collected correctly.

Sampling is called for when we want to find out about a large group but lack time, energy, money, or the other resources needed to contact every group member. For instance, France collects samples at short intervals to collect up-to-date data while staying within a budget. The name used for the process—*recensement en continu* or "rolling census"—signals the intent. Over several years, the French rolling census contacts about 70% of the population.

Sometimes, as in quality control in manufacturing, the measurement process is destructive: the measurement process consumes the item. In a destructive measurement situation, it would be pointless to measure every single item. Instead, a sample will have to do.

22.2 Sampling bias

Collecting a reliable sample is usually considerable work. An ideal is the "simple random sample" (SRS), where all of the items are available, but only some are selected—completely at random—for recording as data. Undertaking an SRS requires assembling a "sampling frame," essentially a census. Then, with the sampling frame in hand, a computer or throws of the dice can accomplish the random selection for the sample.

Understandably, if a census is unfeasible, constructing a perfect sampling frame is hardly less so. In practice, the sample is assembled by randomly dialing phone numbers or taking every 10th visitor to a clinic or similar means. Unlike genuinely random samples, the samples created by these practical methods are not necessarily representative of the larger group. For instance, many people will not answer a phone call from a

[1] Even a population "census" inevitably leaves out some individuals.

stranger; such people are underrepresented in the sample. Similarly, the people who can get to the clinic may be healthier than those who cannot. Such unrepresentativeness is called "**sampling bias**."

Professional work, such as collecting unemployment data, often requires government-level resources. Assembling representative samples uses specialized statistical techniques such as stratification and weighting of the results. We will not cover the specialized techniques in this introductory course, even though they are essential in creating representative samples. The table of contents of a classic text, William Cochran's *Sampling techniques* shows what is involved.

All statistical thinkers, whether expert in sampling techniques or not, should be aware of factors that can bias a sample away from being representative. In political polls, many (most?) people will not respond to the questions. If this non-response stems from, for example, an expectation that the response will be unpopular, then the poll sample will not adequately reflect unpopular opinions. Such **non-response bias** can be significant, even overwhelming, in surveys.

Survival bias plays a role in many settings. The `mosaicData::TenMileRace` data frame provides an example, recording the running times of 8636 participants in a 10-mile road race and including information about each runner's age. Can such data carry information about changes in running performance as people age? The data frame includes runners aged 10 to 87. Nevertheless, a model of running time as a function of age from this data frame is seriously biased. The reason? As people age, casual runners tend to drop out of such races. So the older runners are skewed toward higher performance. (We can see this by taking a different approach to the sample: collecting data over multiple years and tracking individual runners as they age.

> **i** Examples: Returned to base
>
> An inspiring story about dealing with survival bias comes from a World War II study of the damage sustained by bombers due to enemy guns. The sample, by necessity, in-

cluded only those bombers that survived the mission and returned to base. The holes in those surviving bombers tell a story of survival bias. Shell holes on the surviving planes were clustered in certain areas, as depicted in Figure **??**. The clustering stems from survivor bias. The unfortunate planes hit in the middle of the wings, cockpit, engines, and the back of the fuselage did not return to base. Shell hits in those areas never made it into the record.

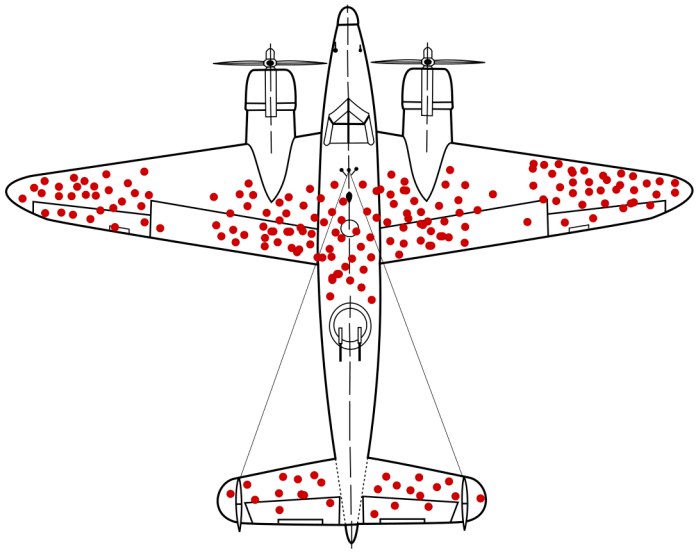


Figure 22.1: An illustration of sh
hole locations in planes that return
to base. Source: Wikipedia

22.3 Measuring sampling variation

Sampling variation is a form of noise. Unlike some other forms of noise, modeling cannot filter out sampling variation or reduce its magnitude. Sampling variation is easiest to see by collecting multiple samples from the same source and summarizing each one. The summaries likely will vary from sample to sample: sampling variation.

Typically, the data frame at hand is our only sample. With no other samples to compare it to, it may seem impossible to measure sampling variation. In this Lesson, we will use simulations

from DAGs to study sampling variation. DAG simulations are suited to this because we can effortlessly collect as many samples as we wish from a DAG. In Lesson **??**, we will use the knowledge gained from the simulations to see how to measure sampling variation even when there is only one sample.

In the spirit of starting simply, we return to `dag01`. This DAG is `x` ⟶ `y`. The causal formula setting the value of `y` is `y ~ 4 + 1.5 * x + exo()`.

It is crucial to remember that sampling variation is not about the row-to-row variation in a single sample. Rather, it is about the variation in the summary from one sample to another. So our initial process for exploring sampling variation will be to carry out many trials, each trial being a summary of a sample.

22.4 Demonstration: Sampling trials

A single sampling trial consists of taking a random sample and computing a sample statistic. To illustrate, here is one trial using a sample size $n = 25$ and a simple model modification, `y ~ 1`.

```
Sample <- sample(dag01, size=25)
Sample %>%
  lm(y ~ 1, data = .) %>%
  coef()
```

```
(Intercept)
   4.317374
```

We cannot see sampling variation directly in the above result because there is only one trial. The sampling variation becomes evident when we run *many* trials. In each trial, a new sample (of size $n = 25$ is taken and summarized.)

```
Trials <- do(500) * {
  Sample <- sample(dag01, size=25)
```

```
  Sample %>%
    lm(y ~ 1, data = .) %>%
    coef()
}
```

Graphics provide a nice way to visualize the sampling variation. Figure **??** shows the results from the set of trials.

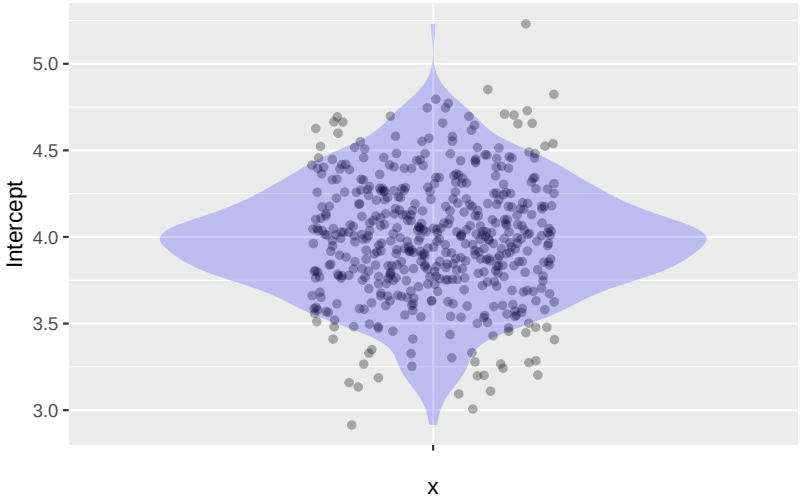


Figure 22.2: The sampling distribution as shown by 500 trials. Each dot is one trial where the model specification `y~1` is fitted to a sample from `dag01` of size $n = 25$.

The variance of the sampling distribution, that is, the sampling variance, is:

```
Trials %>%
  summarize(sampling_variance = var(Intercept))
```

sampling_variance
0.122632

Often, statisticians prefer to use the square root of the sampling variance, which has a technical name in statistics: the **standard error**. The standard error is an ordinary standard deviation in a particular context: the standard deviation of a sample of summaries. The words **standard error** should be followed by a description of the summary and the size of the individual samples involved. Here it would be, "The standard

error of the Intercept coefficient from a sample of size $n = 25$ is around 0.36."

It is easy to confuse "standard error" with "standard deviation." Adding to the potential confusion is another related term, the "margin of error." To avoid this confusion, we will tend to use an *interval* description of the sampling variation called the "**confidence interval**." However, for the present, we will continue with the standard error, sometimes written SE for short.

22.5 SE depends on the sample size

We found an SE of 0.36 on the Intercept in a sample of size $n = 25$. We can see how the SE depends on sample size by repeating the trials for several different sizes, say, $n = 25$, 100, 400, 1600, 6400, 25,000, and 100,000.

The following command estimates the SE a sample of size 400:

```
Trials <- do(1000) * {
  Sample <- sample(dag01, size=25)
  Sample %>%
    lm(y ~ 1, data = .) %>%
    coef()
}
Trials %>% summarize(svar400 = var(Intercept),
                     se400 = sd(Intercept))
```

svar400	se400
0.12766	0.3572954

We repeated this process for each of the other sample sizes. Table **??** reports the results.

There is a pattern in Table **??**. Every time we quadruple n, the sampling variance goes down by a factor of four. Consequently,

Table 22.1: Results of repeating the sampling variability trials for samples of varying sizes.

n	samping_variance	standard_error
25	0.1296000	0.3600
100	0.0361000	0.1900
400	0.0082810	0.0910
1600	0.0018490	0.0430
6400	0.0005290	0.0230
25000	0.0001210	0.0110
100000	0.0000314	0.0056

the standard error—which is just the square-root of the sampling variance—goes down by a factor of 2, that is, $\sqrt{4}$. (The pattern is not exact because there is also sampling variation in the trials, which are really just a sample of all possible trials.)

Conclusion: The larger the sample size, the smaller the sampling variance. For a sample of size n, the sampling variance will be proportional to $1/n$. Or, in terms of the standard error: For a sample size of n, the SE will be proportional to $1/\sqrt{n}$.

22.6 The confidence interval

The "confidence interval" is a more user-friendly format than SE for describing the amount of sampling variation. Being an interval, write it either as [lower, upper] or center±half-width. These styles are equivalent; both styles are correct. (The preferred style can depend on the field or the journal publishing the report.)

In practice, confidence intervals are calculated using special-purpose software such as the `conf_interval()` function, for instance:

```
Hill_racing %>%
  lm(time ~ distance + climb, data=.) %>%
  conf_interval()
```

term	.lwr	.upr
(Intercept)	-533.432471	-406.521402
distance	246.387096	261.229494
climb	2.493307	2.726209

Notice that there is a separate confidence interval for each model coefficient. The sampling variation is essentially the same, but that variation appears different when translated to the various coefficients' units.

⚠ Demonstration: How many digits?

The confidence intervals on the model `time ~ distance + climb`, report the results to many digits. Such a report is appropriate for further calculations that might need doing, but it is usually not appropriate for a human reader. To know how many digits are worth reporting to humans, look toward the standard error. The standard error is a part of a different kind of summary of a model: the "regression report." We will only need to look at regression reports in the last few Lessons of the course. Here we want to point out how many digits are worth reporting to humans. That requires looking at the standard error itself.

Previously, we looked at the confidence intervals on coefficients from the `Hill_racing` model. Now we look at the regression summary, which contains the information on sampling variation in a different format.

```
Hill_racing %>%
  lm(time ~ distance + climb, data=.) %>%
  regression_summary()
```

term	estimate	std.error	statistic	p.value
(Intercept)	-469.976937	32.3582241	-14.52419	0
distance	253.808295	3.7843322	67.06819	0
climb	2.609758	0.0593826	43.94821	0

Each coefficient's standard error appears in the

`std.error` column of the regression summary.
For the human reader, only the first two significant digits of the standard error are worth reporting. (This is true regardless of the data and model design.) Here, the SE is 32 for the Intercept, 3.8 for the distance coefficient, and 0.059 for the climb coefficient. The confidence interval will be the coefficient (column labeled `estimate`) plus or minus "twice" the `std.error`. It is appropriate to round the confidence interval (for a human reader) to the first two significant digits of the standard error.
For example, the confidence interval on the distance coefficient will be $253.808295 \pm 2 \times 3.78433220$. Keep only the digits before the first two significant digits of the SE, so the reported interval can be 253.8 ± 3.8.

23 Confidence intervals from a single sample

Lesson **??** introduced separating data into components: signal and noise. The *signal* is a summary of the data that tells us something we want to know. Often, the signal will be one or more coefficients from a regression report, but it might be something as simple as the mean, median, or standard deviation of a variable in a data frame.

The *noise* comes into the data from various sources: e.g., error in measurement or a data-entry blunder. Another source of noise is omnipresent (except in a perfect census): sampling variation as discussed in Lesson **??**. Sampling variation arises because the particular sample we happen to be working with reflects, to some extent, the play of luck. If we had happened to select another sample, the results would be different.

In general, whenever we measure something, say the altitude of a plane or the fuel economy of a car, it is helpful to know the "**precision**" of that estimate. It would be disastrous if the measured difference in altitude of two planes flying toward a common point were imprecise to the extent that the difference might actually be zero! Knowing the precision of the altitude measurement enables us to space the place in a safe way.

One way to think about precision is in terms of repeatability. If we make multiple measurements of the same object in the same manner, the precision is the degree to which those measurements vary from one another. With the instrumentation used for physical measurements, the precision of individual measurements is estimated by repeated measuring the same thing. Likewise, in statistical summaries, the precision is related to sampling variation.

In Lesson **??**, we repeated trials over and over again to gain some feeling for sampling variation. We quantified the repeatability in any of several closely related ways: the sampling variance or its square root (the "standard error") or a "margin of error" or a "confidence interval." Our experiments with simulations demonstrated an important property of sampling variation: the amount of sampling variation depends on the sample size n. In particular, the sampling variance gets smaller as n increases in proportion to $1/n$. (Consequently, the standard error gets smaller in proportion to $1/\sqrt{n}$.)

It is time to take off the DAG simulation training wheels and measure sampling variation from a *single* data frame. Our first approach will be to turn the single sample into several smaller samples: subsampling. Later, we will turn to another technique, resampling, which draws a sample of full size from the data frame. Sometimes, in particular with regression models, it is possible to calculate the sampling variation from a formula, allowing software to carry out and report the calculations automatically.

23.1 Subsampling

To "subsample" means to draw a smaller sample from a large one. "Small" and "large" are relative. For our example, we turn to the `TenMileRace` data frame containing the record of thousands of runners' times in a race, along with basic information about each runner. There are many ways we could summarize `TenMileRace`. Any summary would do for the example. We will summarize the relationship between the runners' ages and their start-to-finish times (variable `net`), that is, `net ~ age`. To avoid the complexity of a runner's improvement with age followed by a decline, we will limit the study to people over 40.

```
TenMileRace %>% filter(age > 40) %>%
  lm(net ~ age, data = .) %>% coef()
```

```
(Intercept)         age
```

```
(Intercept)         age
 4278.21279    28.13517
```

The units of `net` are seconds, and the units of `age` are years. The model coefficient on `age` tells us how the `net` time changes for each additional year of `age`: seconds per year. Using the entire data frame, we see that the time to run the race gets longer by about 28 seconds per year. So a 45-year-old runner who completed this year's 10-mile race in 3900 seconds (about 9.2 mph, a pretty good pace!) might expect that, in ten years, when she is 55 years old, her time will be longer by 280 seconds.

It would be asinine to report the ten-year change as 281.3517 seconds. The runner's time ten years from now will be influenced by the weather, crowding, the course conditions, whether she finds a good pace runner, the training regime, improvements in shoe technology, injuries, and illnesses, among other factors. There is little or nothing we can say from the `TenMileRace` data about such factors.

There's also sampling variation. There are 2898 people older than 40 in the `TenMileRace` data frame. The way the data was collected (radio-frequency interrogation of a dongle on the runner's shoe) suggests that the data is a census of finishers. However, it is also fair to treat it as a sample of the kind of people who run such races. People might have been interested in running but had a schedule conflict, lived too far away, or missed their train to the start line in the city.

We see sampling variation by comparing multiple samples. To create those multiple samples from `TenMileRace`, we will draw, at random, subsamples of, say, one-tenth the size of the whole, that is, $n = 290$

```
Over40 <- TenMileRace %>% filter(age > 40)
lm(time ~ age, data = Over40 %>% sample(size=290)) %>% coef()
```

```
(Intercept)         age
 4366.31506    31.19676
```

```
lm(time ~ age, data = Over40 %>% sample(size=290)) %>% coef()
```

```
(Intercept)         age
 4138.10256    35.16787
```

The age coefficients from these two subsampling trials differ one from the other by about 0.5 seconds. To get a more systematic view, run more trials:

```
# a sample of summaries
Trials <- do(1000) * {
  lm(time ~ age, data = sample(Over40, size=290)) %>% coef()
}
# a summary of the sample of summaries
Trials %>%
  dplyr::summarize(se = sd(age))
```

se
8.986386

We used the name `se` for the summary of samples of summaries because what we have calculated is the standard error of the age coefficient from samples of size $n = 290$.

In Lesson **??** we saw that the standard error is proportional to $1/\sqrt{n}$, where n is the sample size. From the subsamples, know that the SE for $n = 290$ is about 9.0 seconds. This tells us that the SE for the full $n = 2898$ samples would be about $9.0\frac{\sqrt{290}}{\sqrt{2898}} = 2.85$.

So the interval summary of the `age` coefficient—the *confidence interval*— is

$$\underbrace{28.1}_{\text{age coef.}} \pm 2\times \underbrace{2.85}_{\text{standard error}} \quad = \quad 28.1 \pm \underbrace{5.6}_{\text{margin of error}} \qquad \text{or, equivalently, 22.6 to 33.6}$$

23.2 Bootstrapping

There is a trick, called "**resampling**," to generate a random subsample of a data frame with the same n as the data frame: draw the new sample randomly from the original sample **with replacement**. An example will suffice to show what the "with replacement" does:

```
example <- c(1,2,3,4,5)
# without replacement
sample(example)
```

```
[1] 1 4 3 5 2
```

```
# now, with replacement
sample(example, replace=TRUE)
```

```
[1] 2 4 3 3 5
```

```
sample(example, replace=TRUE)
```

```
[1] 3 5 4 4 4
```

```
sample(example, replace=TRUE)
```

```
[1] 1 1 2 2 3
```

```
sample(example, replace=TRUE)
```

```
[1] 4 3 1 4 5
```

The "with replacement" leads to the possibility that some values will be repeated two or more times and other values will be left out entirely.

The calculation of the SE using resampling is called "**bootstrapping**."

⚠ Demonstration: Bootstrapping the standard error

We will apply bootstrapping to find the standard error of the `age` coefficient from the model `time ~ age` fit to the `Over40` data frame.
There are two steps:

1. Run many trials, each of which fits the model `time ~ age` using `lm()`. From trial to trial, the data used for fitting is a resampling of the `Over40` data frame. The result of each trial is the coefficients from the model.

2. Summarize the trials with the standard deviation of the `age` coefficients.

```
# run many trials
Trials <- do(1000) * {
  lm(time ~ age, data = sample(Over40, replace=TRUE)) %>%
       coef()
}
# summarize the trials to find the SE
Trials %>% summarize(se = sd(age))
```

se
2.859483

23.3 Confidence intervals from software

The same mathematical process that powers regression modeling software such as `lm()` can be used to compute standard

errors for model coefficients as part of the fitting process. So, for regression models, finding a confidence interval is just a matter of asking for it.

[Note: Experienced R users will know that the "standard" function for calculating confidence intervals is `confint()`, which is used in exactly the same manner as `conf_interval()`. Regrettably, `confint()` does not create a data frame. In keeping with these Lessons use of data wrangling, the `conf_interval()` from the `{math300}` package reformats the output of `confint()` into a data frame.]

There are several ways to do the asking. In R, the `conf_interval()` function makes it easy to extract the confidence intervals on each coefficient from a model. For example:

```
lm(net ~ age + sex, data = TenMileRace) |> conf_interval()
```

term	.lwr	.upr
(Intercept)	5270.45170	5407.85920
age	15.04242	18.74483
sexM	-765.85978	-687.37917

Each row of the result reports the confidence interval for one coefficient from the model.

24 Effect size

Regression modeling and confidence intervals provide a substantial toolbox to support statistical thinking. This Lesson starts to develop methods using modeling to inform decision-making. Decision-making takes many guises: whether to administer medicine, change a budget, raise or lower a price, respond to an evolving situation, and so on.

A useful simplification splits support for decision-making into two broad categories.

1. **Making a prediction** for an individual choice. The need for predictions arises in both mundane and critical settings. For instance, an airline needs to set prices. They want to maximize revenue. Higher prices will bring in more money per seat but may reduce the number of people flying. To make the pricing decision, the airline needs a prediction about what the demand will be for those seats, which may vary based on price, day of the week, time of day, time of year, origin and destination of the flight, and so on. Another example: Merchants and social media sites must choose what products or posts to display to a viewer. Merchants have many products, and social media has many news feeds, tweets, and competing blog entries. The people who manage these websites want to promote the products or postings most likely to cause a viewer to respond. To identify viable products or postings, the site managers construct predictive models based on earlier viewers' choices. We will study prediction models in Lessons 25 and 26,
2. **Intervening** in a system. Such interventions occur on both grand scales and small: changes in government policies such as funding for preschool education or subsidies for renewable energy, closing a road to redirect traffic or

opening a new highway or bus line, changing the minimum wage, etc. Before making such interventions, it is wise to know what the consequences are likely to be. Figuring this out is often a matter of understanding how the system works: what causes what. As interventions often affect multiple individuals, influencing the overall trend of the effect across individuals might be the goal instead of predicting how each individual will be affected.

This Lesson focuses on "**effect size**," a measure of how changing an explanatory variable will play out in the response variable. Built into the previous sentence is an assumption that the explanatory variable *causes* the response variable. In Lessons 28 through 31, we will look into ways to make responsible claims about whether a connection between variables is causal. Here, we will focus on the calculation and interpretation of effect size.

24.1 Effect size: Input to output

An intervention changes something in the world. Some examples are the budget for a program, the dose of a medicine, or the fuel flow into an engine. The thing being changed is the *input*. In response, something else in the world changes, for instance, the reading ability of students, the patient's serotonin levels (a neurotransmitter), or the power output from the engine. The thing that changes in response to the change in input is called the "output."

"**Effect size**" describes the change in the output with respect to the change in the input. The simplest case is when the output is a quantitative variable. In this case, the change in the output is a difference between two numbers. The form of the effect size depends on the input type. For example, for a quantitative input, the effect size will be a *ratio*, that is, a rate. (For calculus students: the effect size is a derivative of the output with respect to the input.)

To measure an effect size from data, construct a model with the output as the response variable and the input as an explanatory variable.

Example: Fuel economy

A person buying a car typically has multiple objectives in mind. Perhaps the buyer is deciding whether to order a more powerful engine. This decision has consequences, including a reduction in fuel economy. The decision variable—the engine size—is the input; the fuel economy is the output.

Since both input and output are quantitative, the effect size will be a rate: change in fuel economy per change in engine size. To inform a decision, use data such as the `math300::MPG` data frame, which compares various car models. `MPG` records the engine size in terms of `displacement`, in liters. Fuel economy is listed in miles per gallon, differently for city versus highway driving.

The buyer is debating between a 2-liter and a 3-liter engine. Most driving will be in the city. To calculate the effect size, first build a model with the output (`mpg_city`) as the response variable and the input (`displacement`) as an explanatory variable.

```
Mod <- lm(mpg_city ~ displacement, data=MPG)
```

Second, evaluate that model for the range of inputs under consideration.

```
model_eval(Mod, displacement=c(2, 3))
```

displacement	.output	.lwr	.upr
2	24.01437	15.91915	32.10959
3	20.86976	12.77698	28.96254

The change in the input from 3 liters displacement to 2 liters leads to a change in fuel economy of $24.0 - 20.9 = -3.1$ miles per gallon. The change in displacement is $3 - 2 = 1$ liters. The effect size is the ratio between the output change and the input change. Here, that is -3.1 miles per gallon per liter.

The decision-maker may be more concerned about the cost

of driving than with the miles per gallon. Then the appropriate response variable might be `EPA_fuel_cost`, denominated in dollars per year.

```
Mod2 <- lm(EPA_fuel_cost ~ displacement, data=MPG)
model_eval(Mod2, displacement=c(2, 3))
```

displacement	.output	.lwr	.upr
2	1585.887	1000.649	2171.125
3	1882.534	1297.473	2467.596

The change in output is about $300 per year. However, the change in input is still 1 liter. The effect size is, therefore, $300 per year per liter.

Some decision variables are categorical. For instance, the buyer might like the idea of an engine that automatically turns off when the car is stopped at a light or in traffic. The `start_stop` variable, which has categorical levels "Yes" and "No," records whether the car has this feature. Effect size estimation is slightly different when the input is categorical rather than quantitative. Still, build a model and compare the change in output to the change in input:

```
Mod3 <- lm(EPA_fuel_cost ~ start_stop, data=MPG)
model_eval(Mod3, start_stop=c("No", "Yes"))
```

start_stop	.output	.lwr	.upr
No	1872.193	916.0164	2828.369
Yes	1945.194	989.0637	2901.324

In this case, the change in output is $73 per year; the change in input is "Yes" - "No." But, of course, it is meaningless to subtract one categorical level from another. Consequently, the effect size of `start_stop` on fuel cost cannot be quantified as a ratio. So, instead, the effect size is simply the difference in the output: a $73 per year increase with the Start/Stop feature.

The statistical thinker knows to pay attention to whether a calculated result makes sense. It seems unlikely that the Start/Stop feature causes more fuel to be consumed. Was there an error? Perhaps we did the subtraction backward? Check the report from `model_eval()` to make sure.

Here, the problem is not arithmetic. However, there is another possibility. It might be that manufacturers include the Start/Stop feature with big cars but not little ones. Then, even if Start/Stop might save gas when everything else is held constant, because the big cars use more fuel than little cars, it only *appears* that Start/Stop hurts fuel economy. This theory is, at this point, speculation: a hypothesis. Such a mixture of effects—big versus small car mixed with availability of Start/Stop—is called "**confounding**." In Lessons 28 through 30, we discuss identifying and dealing with possible confounding.

> ⚠ Confounding?
>
> The surprising positive effect size of the Start/Stop feature caused a double take and led us to think of ways to make sense of the result. Right now, we simply have a hypothesis that Start/Stop is associated with bigger cars. (We will check that out in a little bit.)
> The effect size of annual fuel cost with respect to engine displacement, $300 per year per liter, did not surprise us. Perhaps it should have. After all, larger vehicles tend to have larger engines. This relationship might lead to confounding between vehicle size and engine displacement. We think we are looking at engine displacement, but instead, the effect might be due to vehicle size. Again, just a hypothesis at this point. The statistical thinker knows to consider possible confounding from the start.

24.2 Categorical outputs

Sometimes the relevant effect size involves a categorical output variable. A case in point is the possible confounding of the

Start/Stop feature with vehicle size. To investigate this, we should build a model with Start/Stop as the output and vehicle size as the input.

In this case, the issue of whether vehicle size causes Start/Stop is not essential. We are not concerned with the decisions made by automobile designers so much as with the possible confounding.

When the output variable is categorical, it is not reasonable to calculate the change in output as the difference in categories. As before, "Yes" - "No" is not a number. Still, there is a meaningful and helpful way to quantify a change in a categorical output.

The essential insight is quantifying the change in output in terms of probabilities. For instance, a small effect size would reflect a slight chance of the output changing from one level to another.

The appropriate model type for a categorical output is to transform the output to a zero-one variable, as introduced in Lesson **??**. We will present this in a demonstration here and return to the topic more fully in Lesson 34.

⚠ Demonstration: Start/Stop and vehicle size

As described earlier, we are interested in the possibility that Start/Stop is available mainly on large, higher-fuel-consumption cars. If so, that would explain why the effect size we calculated of fuel cost with respect to Start/Stop was positive.
The model we build will have a zero-one encoding of Start/Stop as the response and the vehicle's fuel cost as the explanatory variable.

```
MPG <- MPG %>%
    mutate(has_start_stop = zero_one(start_stop, one="Yes"))
Mod4 <- lm(has_start_stop ~ EPA_fuel_cost, data = MPG)
model_eval(Mod4, EPA_fuel_cost=c(1600, 2000))
```

EPA_fuel_cost	.output	.lwr	.upr
1600	0.4901341	-0.4891981	1.469466
2000	0.5207835	-0.4583924	1.499959

The `.output` here is interpreted as a *probability* of `start_stop` having the value "Yes." (That is because we set `one="Yes"` in the `zero_one()` conversion.) The `model_eval()` report indicates $400 per year increase in fuel cost is associated with a three percentage point increase in the probability of a vehicle having a Start/Stop feature. That is a small effect, so we see little support for our hypothesis that Start/Stop tends to be installed on larger, more fuel-efficient vehicles.

24.3 Multiple explanatory variables

When a model has more than one explanatory variable, each has a different effect size.

As an example, consider the price of books. We have some data that might be informative, `moderndive::amazon_books`. What is the effect size of page count on price. The appropriate model here is `list_price ~ num_pages`. The effect size is easy to compute:

```
Mod1 <- lm(list_price ~ num_pages, data = moderndive::amazon_books)
model_eval(Mod1, num_pages = c(200, 400))
```

num_pages	.output	.lwr	.upr
200	15.82014	-11.636987	43.27726
400	19.79643	-7.637503	47.23037

We elected to compare 200-page books with 400-page books, simply because those seem like reasonable book lengths. However, the longer book costs about 4 dollars more. So the effect

size, to judge from this model, is $4 divided by 200 more pages, which comes to 2 cents per page.

Another effect size is needed to address the question: Are hardcovers more expensive than paperbacks? The output is still price. But now, the input is categorical. In the `moderndive::amazon_books` data frame, the variable `hard_paper` has levels "P" and "H." A possible model: `list_price ~ hard_paper`.

```
Mod2 <- lm(list_price ~ hard_paper, data = amazon_books)
model_eval(Mod2, hard_paper = c("P", "H"))
```

hard_paper	.output	.lwr	.upr
P	17.13523	-10.62291	44.89338
H	22.39393	-5.46052	50.24839

A hardcover book costs about $5.25 more than a paperback book. Since the input is categorical, there is no change of input to divide by, so the effect size is $5.25 when going from a paperback to a hardcover.

We can look at the effects of page length and cover-type separately. Instead, we can include both as explanatory variables.

```
Mod3 <- lm(list_price ~ hard_paper + num_pages, data = amazon_books)
model_eval(Mod3, hard_paper = c("P", "H"),  num_pages=c(200, 400))
```

hard_paper	num_pages	.output	.lwr	.upr
P	200	14.52494	-12.641928	41.69182
H	200	19.48253	-7.785720	46.75077
P	400	18.43605	-8.709404	45.58151
H	400	23.39363	-3.847698	50.63497

This output requires some interpretation. We have got short and long paperback books and short and long hardcover books. What should we compare to what?

The convention is to consider each of the two inputs separately and hold the other input constant when we compare.

Effect size of ***num_pages*** *on* ***list_price***. To hold `hard_paper` constant, we will compare the two rows of the `model_eval()` report that have a "P" value for `hard_paper`. The difference in output for these two rows is $3.90. The effect size divides by the change in input—200 pages—so the effect size is just under 2 cents per page. *Effect size of* ***hard_paper*** *on* ***list_price***. This time we will hold `num_pages` constant, say at 200 pages. Comparing the corresponding rows in the `model_eval()` output shows a change in list price of $4.96 when going from paper back to hard cover. There is no special reason we decided to hold `hard_paper` constant at "P" rather than "H" or hold `num_pages` constant at 200 rather than 400. In general, the effect size will depend on the value being held constant. Choose a value that's relevant to the purpose at hand.

In these Lessons we are building models with additive effects.That is what the `+` means in, say, `list_price ~ hard_paper + num_pages`. We do this to keep the effect-size story as simple as possible. (Occasionally, you will see examples with *multiplicative* effects, called "**interactions**." The tilde expressions for such models involve `*` rather than `+`, as in `list_price ~ hard_paper + num_pages.`

24.4 Interval estimates

Statistical thinkers know that any estimate they make, including estimates of effect sizes, involves sampling variation. Consequently, give an *interval* estimate. The interval communicates to the decision-maker the uncertainty in the estimated quantity. Sophisticated decision-makers keep this uncertainty in mind, considering the range of outcomes likely from whatever use they make of effect size. On the other hand, statistically naive decision makers—even highly educated decision-makers can be statistically naive—look at the interval and sometimes ask the modeler, "Just give me a number. I don't know what to do with two numbers." Such a request might elicit a frank response: "If you don't know what to do with two numbers, you

also won't know what to do with one number." Unfortunately, that kind of frankness is not often well received; a reasonable alternative is: "The interval indicates the amount of uncertainty in the result. We'll need to collect more data if you want to reduce the uncertainty." (Lesson **??** introduces a not-always-available alternative to collecting more data: building a better model!)

For the additive models that we are mainly using in these Lessons, the effect size is identical to a model coefficient. For these models, the confidence interval on the coefficient is the confidence interval on the effect size. For instance,

```
Mod3 %>% conf_interval()
```

term	.lwr	.upr
(Intercept)	7.0390179	14.1886534
hard_paperH	1.5580344	8.3571295
num_pages	0.0102362	0.0288749

25 Mechanics of prediction

An effect size describes the relationship between two variables in an input/output format. Lesson **??** introduced effect size in the context of causal connections as if turning a knob to change the input will produce a change in the output. Such mechanistic connections make for a nice mental image for those considering intervening in the world but can be misleading.

First, the mere calculation of an effect size does not establish a causal connection. The statistical thinker has more work to do to justify a causal claim, as we will see in Lesson **??**.

Second, owing to noise, the input/output relationship quantified by an effect size may not be evident in a single intervention, say, increasing a drug dose for any given individual patient. Instead, effect sizes are descriptions of *average* effects—trends—across a large group of individuals.

This Lesson is about *prediction*: what a model can properly say about the outcome of an individual case. Often, the setting is that we know values for some aspects of the individual but have yet to learn some other aspect of interest.

The word "prediction" suggests the future but also applies to saying what we can about an unknown current or past state. Synonyms for "prediction" include "classification" (Lessons 34 and 35), "conjecture," "guess," and "bet." The phrase "informed guess" is a good description of prediction: using available information to support decision-making about the unknown.

> **i** Example: Differential diagnosis
>
> A patient comes to an urgent-care clinic with symptoms. The healthcare professional tries to diagnose what disease or illness the patient has. A diagnosis is a prediction. The

inputs to the prediction are the symptoms—neck stiffness, a tremor, and so on—as well as facts about the person, such as age, sex, occupation, and family history. The prediction output is a set of probabilities, one for each medical condition that could cause the symptoms.
Doctors learn to perform a *differential diagnosis*, where the current set of probabilities informs the choices of additional tests and treatments. The probabilities are updated based on the information gained from the tests and treatments. This update may suggest new tests or treatments, the results of which may drive a new update. The television drama *House* provides an example of the evolving predictions of differential diagnosis in every episode.

Differential diagnosis is a cycle of prediction and action. This Lesson, however, is about the mechanics of prediction: taking what we know about an individual and producing an informed guess about what we do not yet know.

25.1 The prediction machine

A statistical prediction is the output of a kind of special-purpose machine. The inputs given to the machine are values for what we already know; the output is a value (or interval) for the as-yet-unknown aspects of the system.

There are always two phases involved in making a prediction. The first is building the prediction machine. The second phase is providing the machine with inputs for the individual case, turning the machine crank, and receiving the prediction as output.

These two phases require different sorts of data. Building the machine requires a "historical" data set that includes records from many instances where we already know two things: the values of the inputs and the observed output. The word "historical" emphasizes that the machine-building data must already have known values for each of the inputs and outputs of interest.

The evaluation phase—turning the crank of the machine—is simple. Take the input values for the individual to be predicted, put those inputs into the machine, and receive a predicted value as output. Those input values may come from pure speculation or the measured values from a specific case of interest.

25.2 Building and using the machine

To illustrate building a prediction machine, we turn to a problem first considered quantitatively in the 1880s: the relationship between parents' heights and their children's heights at adulthood. The `Galton` data frame records the heights of about 900 children, along with their parents' heights. Suppose we want to predict a child's adult height (variable name: `height`) from his or her parents' heights (`mother` and `father`). An appropriate model specification is `height ~ mother + father`. We use the model-training function`lm()` to transform the model specification and the data into a model.

```
Mod1 <- lm(height ~ mother + father, data = Galton)
```

As the output of an R function, `Mod1` is a computer object. It incorporates a variety of information organized in a somewhat complex way. There are several often-used ways to extract this information in ways that serve specific purposes.

One of the most common ways to see what is in a computer object like `Mod1` is by printing:

```
print(Mod1)
```

```
Call:
lm(formula = height ~ mother + father, data = Galton)

Coefficients:
(Intercept)       mother       father
    22.3097       0.2832       0.3799
```

Newcomers to technical computing tend to confuse the printed form of an object with the object itself. For example, the `Mod1` object contains many components, but the printed form displays only two: the model coefficients and the command used to construct the object.

We have already used some other functions to extract information from a model object. For instance,

```
Mod1 %>% coef()
```

```
(Intercept)      mother      father
 22.3097055   0.2832145   0.3798970
```

```
Mod1 %>% conf_interval()
```

term	.lwr	.upr
(Intercept)	13.8569119	30.7624990
mother	0.1867750	0.3796540
father	0.2898301	0.4699639

```
Mod1 %>% regression_summary()
```

term	estimate	std.error	statistic	p.value
(Intercept)	22.3097055	4.3068968	5.179995	3e-07
mother	0.2832145	0.0491382	5.763635	0e+00
father	0.3798970	0.0458912	8.278209	0e+00

Another extractor, `model_eval()`, is particularly convenient for prediction. Perhaps the most common use is to provide new input values to the model function, with `model_eval()` producing a data frame showing the output of the model function. To illustrate, here is how to calculate the predicted height of the child of a 63-inch-tall mother and a 68-inch father.

```
Mod1 %>% model_eval(mother = 63, father=68)
```

mother	father	.output	.lwr	.upr
63	68	65.98521	59.33448	72.63594

The data frame includes the input values along with a point value for the prediction (`.output`) and a prediction interval (`.lwr` to `.upr`).

Naturally, the predictions depend on the explanatory variables used in the model. For example, here is a model that uses only `sex` to predict the child's height:

```
Mod2 <- lm(height ~ sex, data = Galton)
Mod2 %>% model_eval(sex=c("F", "M"))
```

sex	.output	.lwr	.upr
F	64.11016	59.18024	69.04009
M	69.22882	64.29928	74.15835

This model includes three explanatory variables:

```
Mod3 <- lm(height ~ mother + father + sex, data = Galton)
Mod3 %>% model_eval(mother=63, father=68, sex=c("F", "M"))
```

mother	father	sex	.output	.lwr	.upr
63	68	F	63.20546	58.97128	67.43964
63	68	M	68.43141	64.19783	72.66499

In Lesson **??**, we will look at the components that make up the prediction interval and some ways to use it.

26 Constructing a prediction interval

Lesson **??** introduced predictions in two forms:

1. a **point quantity**, the direct output of the model function.
2. the **prediction interval**, which indicates a range of likely values for the quantity being predicted.

To clarify this distinction, consider this three-step procedure that trains a model, extracts the model function, and applies the model function to inputs to generate a prediction in point-estimate form.

```
Time_mod <- lm(time ~ distance + climb, data = Hill_racing)
Time_mod_fun <- makeFun(Time_mod)
Time_mod_fun(distance=10, climb=500)
```

In the first line, `lm()` is used to train a model.

The second line, `Time_mod_fun <- makeFun(Time_mod)`, creates and names a *function* that implements the input/output relationship defined by the model.

The third line uses the ordinary parentheses notation to apply the newly created `Time_mod_fun()` to specific values of the argument, generating the corresponding output value.

```
# applying the function to arguments
Time_mod_fun(distance=10, climb=500)
```

```
       1
3372.985
```

In these Lessons, whenever we refer to the "model function," we mean a model translated into the form of a function. The point is to emphasize the input-to-output relationship implied by a model.

In topics like calculus, functions are the primary objects of interest. Calculus operations such as differentiation, anti-differentiation, and zero-finding always act on functions. However, calculus software hardly ever lets one interrogate a function to find properties such as the range, domain, continuity, and asymptotes. Instead, students are expected to look at the formula of a function to deduce these properties.

In statistical modeling, model functions are *not* an object of primary interest. Why? Because there are several other properties of models are essential to interpreting the results of using a model. These properties include the residuals from model training and more abstract and advanced ones, such as the model's "degrees of freedom." People design software to construct model objects—for us, objects of class "lm"—from which these properties can be accessed and translated by software into valuable forms.

For this reason, there is no need to construct the model function explicitly. Consequently, one generally does not use the function application syntax directly as we did with `Time_mod_fun(distance=10, climb=500)`. Instead, one invokes the model function with other software that can use all the information in a model object. For us, that software will be the `model_eval()` extractor.

Use `model_eval()` as you do the other familiar extractors such as `coef()` or `conf_interval()`. To generate a prediction, give `model_eval()` arguments specifying the desired inputs to use with the model function.

```
Time_mod %>% model_eval(distance=10, climb=500)
```

distance	climb	.output	.lwr	.upr
10	500	3372.985	1664.41	5081.56

Notice that the result from the above command includes a column `.output` which will always be an exact match to the output the model function will have generated. However, there is more to the output of `model_eval()`. The interval form of the prediction is of particular importance, contained in the columns `.lwr` and `.upr`.

Many people prefer a point prediction, possibly because the single number suggests a single, correct answer, which is somehow emotionally comforting. *But the comfort is unjustified.*

The proper form for a prediction is a **prediction interval**: two numbers bounding the lower and upper limits for the likely outcome. For the hill-racing model, the point prediction is 3372.985 seconds, which is a running time of just under one hour. Nothing about this single number even tells us how many digits are appropriate. The prediction interval tells a different story. The interval, 1700 to 5100 seconds, conveys the appropriate uncertainty in the prediction.

26.1 Where does the prediction interval come from

The prediction interval has two distinct components:

1. The uncertainty in the model function and hence in the output of the model function.
2. The size of the residuals found when training the model.

Consider first the model function. For the running-time model, we can construct the model function from the coefficients of the linear model. These are:

```
Time_mod %>% coef()
```

```
(Intercept)    distance       climb
-469.976937  253.808295    2.609758
```

The algebraic expression for the model function is straightforward:

$$t(d, c) \equiv -470 + 254d + 2.61c \ .$$

The statistical thinker knows that such coefficients have uncertainty due to sampling variation. That uncertainty is, of course, quantified by the confidence interval.

```
Time_mod %>% conf_interval()
```

term	.lwr	.upr
(Intercept)	-533.432471	-406.521402
distance	246.387096	261.229494
climb	2.493307	2.726209

Since we cannot legitimately claim to know the values of the coefficients any better than indicated by these confidence intervals, we ought to temper our claims about the model function so that it reflects the uncertainty in the coefficients. For instance, we might provide an interval for the model output, using in an "upper" function the high ends of the confidence intervals on the coefficients and another "lower" function that uses the low ends of the confidence interval. Like this:

$$t_{upr}(d, c) \equiv -407 + 261d + 2.72c t_{lwr}(d, c) \equiv -533 + 246d + 2.49c$$

Evaluate both the lower and upper functions to get an *interval* on the model output. That would give us $t_{lwr}(10, 500) = 3172$ and $t_{upr}(10, 500) = 3569$.

This idea for generating the "lower" and "upper" functions has the right spirit but is not on target mathematically. The reason is that using the low end of the confidence interval for all coefficients is overly pessimistic; usually, the uncertainty in the different coefficients cancels out to some extent.

The mathematics for the correct "lower" and "upper" functions are well understood but too advanced for the general reader.

For our purposes, it suffices to know that `model_eval()` knows how to do the calculations correctly.

The prediction interval produced by `model_eval()` includes both components (1) and (2) listed above. Insofar as we are interested in component (1) in isolation, the correct sort of interval—a *confidence interval*—can be requested from `model_eval()`.

```
Time_mod %>%
    model_eval(distance=10, climb=500, interval="confidence")
```

distance	climb	.output	.lwr	.upr
10	500	3372.985	3335.264	3410.706

This report shows that the confidence interval on the model output—that is, just component (1) of the prediction interval—is pretty narrow: 3335 seconds to 3411 seconds, or, in plus-or-minus format, 3373 ± 38 seconds.

The prediction interval—that is, the sum of components (1) and (2)—is comparatively huge: 1700 to 5100 seconds or, in plus-or-minus format, 3400 ± 1700 seconds. That is almost 50 times wider than the confidence interval.

Why is the prediction interval so much more comprehensive than the confidence interval? The confidence interval reports on the sampling variation of a model constructed as an average over all the data, the $n = 2236$ participants recorded in the `Hill_racing` data frame. However, each runner in `Hill_racing` has their own individual time: not an average but just for the individual. The individual value might be larger or smaller than the average. How much larger or smaller? The residuals for the model provide this information. As always, we can measure the individual-to-individual variation with the standard deviation.

```
Time_mod %>% model_eval() %>% summarize(se_residuals = sd(.resid))
```

```
Using training data as input to model_eval().
```

se_residuals
870.6588

Keeping in mind that the overall spread of the residuals is plus-or-minus "twice" the standard deviation of the residuals, we can say that the residuals indicate an additional uncertainty in the prediction for a runner of about ±1700 seconds. This ±1700 second is our estimate of the **noise** in the measurements. In contrast, the **confidence interval** is about the sampling variation in the signal.

In this case, the **prediction interval** is wholly dominated by noise; the sampling variability contributes only a tiny amount of additional uncertainty.

i Example: Graphics for the prediction interval

We shift the running scene from Scotland to Washington, DC. The race now is a single 10-miler with almost 9000 registered participants. We wish to predict the running time of an individual based on his or her `age`.

```
Age_mod <- lm(net ~ age, data = TenMileRace)
```

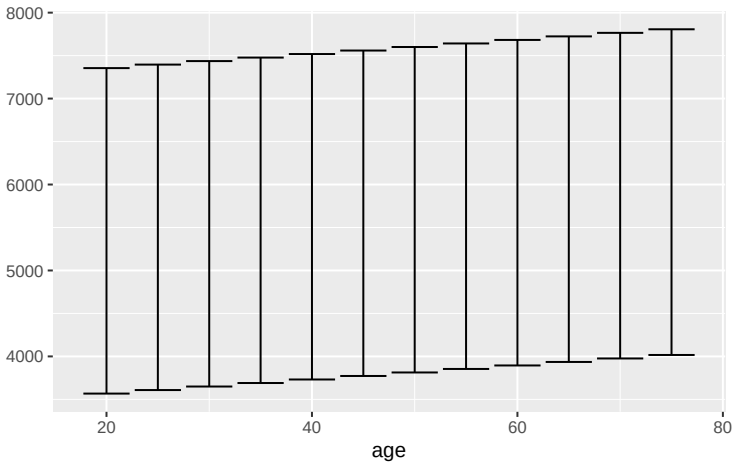
We can see the prediction interval for an individual runner using `mod_eval()`. For example, here it is for a 23-year-old.

```
Age_mod %>% model_eval(age=23)
```

age	.output	.lwr	.upr
23	5485.587	3592.054	7379.119

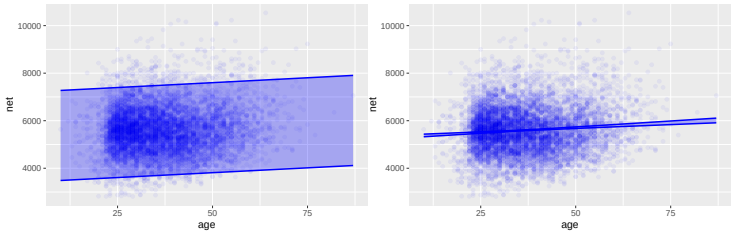
We can also calculate the prediction interval for several different ages and graph out the results with the "errorbar" glyph:

```
Age_mod %>%
  model_eval(age=c(20,25,30,35,40,45,50,55,60,65,70,75)) %>%
  ggplot(aes(x=age)) +
  geom_errorbar(aes(ymin=.lwr, ymax=.upr))
```



For convenience, the `model_plot()` function will do this work for us, plotting the prediction interval along with the training data. We can also direct `model_plot()` to show the confidence interval.

```
### #| column: page-right
model_plot(Age_mod, x=age, interval="prediction", data_alpha=0.05)
model_plot(Age_mod, x=age, interval="confidence", data_alpha=0.05)
```



(a) prediction interval (b) confidence interval

Figure 26.1: The prediction and model-function confidence intervals for the model `net ~ age`.

Since we are looking at the intervals as a function of an input variable, what we formerly showed using the "error-bar" glyph is now shown using a ribbon or **band**.

Notice that the prediction interval covers almost all of the data points. There are hundreds of data points outside the interval, but with almost 9000 rows in the `TenMileRace` data frame, an interval that covers 95% of the data will have about 450 rows *outside* the interval.

Such a prediction interval is of little use; it cannot give a precise prediction about the running time of an individual. The honest prediction of an individual's outcome needs to reflect the spread of all the individuals with a similar age.

In contrast, the *confidence* band on the model function is pleasingly narrow and precise. It covers only a tiny fraction of the raw data. For this very reason, the confidence interval is *inappropriate* for presenting a prediction. As always, confidence intervals only show general trends in the data, not the range of results for an individual prediction. For instance, Figure **??** shows a clear upward trend in running time with age. There is no flat or negatively sloping line compatible with the confidence interval.

To summarize:

1. When making a prediction, report a prediction interval.
2. The prediction interval is always larger than the confidence interval and is usually *much* larger.

The confidence interval is not for predictions. Use a confidence interval when looking at an effect size. Graphically, the confidence interval is to indicate whether there is an overall trend in the model.

27 Review of Lessons 19-26

> **Warning**
>
> I'll put learning challenges here. The class day will be given over to the QR.

28 Covariates

Dr. Mary Meyer is a statistics professor at Colorado State University. In 2006, she published an article recounting an episode from family life:

> *When my daughter was in fourth grade, I took her shopping for dress shoes. I was disappointed in the quality of girls' shoes at every store in the mall. The shoes for boys were sturdy and had plenty of room in the toes. On the other hand, shoes for girls were flimsy, narrow, and had pointed toes. In spite of the better construction for boys, the costs of the shoes were similar! For children the same age, boys had shoes they could run around in, while girls' shoes were clearly for style and not comfort.*
>
> *Upon complaining about this state of affairs, I was told by sales representatives in two stores that boys actually had wider feet than girls, so needed wider shoes. Being very skeptical, I thought I would test this claim.*

We will return to Dr. Meyer's project in a little bit. However, for now, imagine how this situation might be addressed by someone who still needs to develop good statistical thinking skills. We will call this imagined protagonist "Mr. Shoebuyer." Since the salesmen claimed that girls' feet are narrower than boys, Mr. Shoebuyer heads out to measure the widths of girls' and boys' shoes.

A shoe store provides a convenient place to measure the widths of many different shoe styles. Mr. Shoebuyer gets to the shoe store, heads to the children's section, and starts measuring. For each shoe on display, he records the shoe width and whether the shoe is for girls or boys. Here are his data:

sex	width
G	9.0
G	8.5
G	9.0
G	9.5
B	8.6
B	8.4
B	8.8
B	9.4

Once back home, Mr. Shoebuyer uses his calculator to find the mean width of the shoes in each group. His results surprise him:

sex	mean width
Girls	9.0 cm
Boys	8.8 cm

Mr. Shoebuyer happens to be your uncle. He knows you are taking a statistics course and asks you to check his arithmetic. Putting on a statistical thinking hat to the effect size of sex on shoe width, you note the absence of a confidence interval. This omission is easy to fix.

```
Shoebuyer_data %>% lm(width ~ sex, data=.) %>% conf_interval()
```

term	.lwr	.upr
(Intercept)	8.2857603	9.3142397
sexG	-0.5272448	0.9272448

Your uncle is at the table at Thanksgiving break. "Sorry, Uncle, but you don't have nearly enough data to conclude that girls' feet are wider than boys'." Translating the confidence interval into plus-or-minus format, you point out that the difference between the sexes is 0.2 ± 0.8 cm. "You'll need enough data to get that 0.8 margin of error down to something like 0.2." You also point out that there might be a better place to collect data than a shoe store. "It's the feet, not the shoes, that you want to look at."

Aware of these pitfalls, Dr. Meyer worked with the third- and fourth-grade teachers at her daughter's school to collect data. Being a statistical thinker, she thought about what data would illuminate the matter before carrying out the data collection. Her data, a sample of size $n = 39$, are recorded in the `KidsFeet` data frame.

```
lm(width ~ sex, data = KidsFeet) %>% conf_interval()
```

term	.lwr	.upr
(Intercept)	8.9758882	9.4041118
sexG	-0.7125476	-0.0990313

In plus-or-minus format, this confidence interval is -0.4 ± 0.3. Whatever the format, Dr. Meyer's data provides some evidence that girls' feet are narrower than boys'.

As a statistical thinker, Dr. Meyer knows that even though the foot width is the original quantity of interest, other factors might play a role in the system. For example, boys' feet might trend longer or shorter than girls' feet. This possibility should be taken into account by looking at the effect size of `sex` on width, holding length constant. After all, a shoe buyer first tells the salesperson their foot length (or "size"); the salesperson then brings shoes of that size to try on.

```
lm(width ~ sex + length, data=KidsFeet) %>% conf_interval()
```

term	.lwr	.upr
(Intercept)	1.1048182	6.1775184
sexG	-0.4947759	0.0297408
length	0.1202348	0.3218151

Although `sex` is the explanatory variable of primary interest to Dr. Meyer's question, she knows to include other explanatory variables that might play a role. Such explanatory variables, not of direct interest, are called "**covariates**." Dr. Meyer's statistical expertise led her to consider possible covariates *before*

collecting her data and took the trouble of measuring both foot length and width.

The confidence interval on the `sexG` coefficient includes zero when `length` is taken into account. Dr. Meyer's little study provides evidence that even if girls' shoes tend to be narrower than boys', the feet inside them have about the same shape for both sexes.

28.1 All other things being equal

The common phrase "all other things being equal" is a critical qualifier in describing relationships. To illustrate: A simple claim in economics is that a high price for a commodity reduces the demand. For example, increasing the price of heating fuel will reduce demand as people turn down thermostats to save money. Nevertheless, the claim can be considered obvious only with the qualifier *all other things being equal*. For instance, the fuel price might have increased because winter weather has increased the demand for heating compared to summer. Thus, higher prices may be associated with higher demand. Therefore, increased price may not be associated with lower demand unless holding other variables, such as weather conditions, constant.

In economics, the Latin equivalent of "all other things being equal" is sometimes used: "**ceteris paribus**". So, the economics claim would be, "higher prices are associated with lower demand, *ceteris paribus*."

Although the phrase "all other things being equal" has a logical simplicity, it is impractical to implement "all." So instead of the blanket "all other things," it is helpful to consider just "some other things" to be held constant, being explicit about what those things are. Other phrases along the same lines are "taking into account …" and "controlling for …." Those additional variables that are to be considered are called "**covariates**.

Example: Covariates and Death

This news report appeared in 2007:

> **Heart Surgery Drug Carries High Risk, Study Says.** A drug widely used to prevent excessive bleeding during heart surgery appears to raise the risk of dying in the five years afterward by nearly 50 percent, an international study found. The researchers said replacing the drug—aprotinin, sold by Bayer under the brand name Trasylol—with other, cheaper drugs for a year would prevent 10,000 deaths worldwide over the next five years.
> Bayer said in a statement that the findings are unreliable because Trasylol tends to be used in more complex operations, and the researchers' statistical analysis did not fully account for the complexity of the surgery cases. The study followed 3,876 patients who had heart bypass surgery at 62 medical centers in 16 nations. Researchers compared patients who received aprotinin to patients who got other drugs or no antibleeding drugs. Over five years, 20.8 percent of the aprotinin patients died, versus 12.7 percent of the patients who received no antibleeding drug. [This is a 64% increase in the death rate.] When researchers adjusted for other factors, they found that patients who got Trasylol ran a 48 percent higher risk of dying in the five years afterward. The other drugs, both cheaper generics, did not raise the risk of death significantly. The study was not a randomized trial, meaning that it did not randomly assign patients to get aprotinin or not. In their analysis, the researchers took into account how sick patients were before surgery, but they acknowledged that some factors they did not account for may have contributed to the extra deaths. - Carla K. Johnson, Associ-

ated Press, 7 Feb. 2007

The report involves several variables. Of primary interest is the relationship between (1) the risk of dying after surgery and (2) the drug used to prevent excessive bleeding during surgery. Also potentially important are (3) the complexity of the surgical operation and (4) how sick the patients were before surgery. Bayer disputes the published results of the relationship between (1) and (2) holding (4) constant, saying that it is also essential to hold variable (3) constant.

The total relationship involves a death rate of 20.8 percent of patients who got aprotinin versus 12.7 percent for the patients taking the generic drugs: an increase in the death rate by a factor of 1.64. However, when the researchers looked at a partial relationship (holding constant patient sickness before the operation), the effect size of aprotinin on mortality was less: a factor of 1.48. In other words, the model `death ~ aprotinin` shows a 64% increase in the death rate, but the model `death ~ aprotinin + sickness` shows a slightly smaller increase in death rate: 48%. The difference between the two estimates reflects doctors being more likely to give aprotinin to sicker patients.

The story's last paragraph states that the choice of patients receiving aprotinin versus the generic drugs was not made at random. Some readers may find this reassuring. Why in the world would anyone prescribe a drug at random? The point, however, is to select randomly who gets which drug *among the patients for whom the drugs would be appropriate*. The phrase "randomized trial" used in the paragraph means specifically an *experiment* in which one treatment or the other—aprotinin versus the generic drugs—is assigned at random. The virtues of experiment and the vital role of random assignment are detailed in Lesson **??**.

28.2 Letting things change as they will

Using covariates in models enables the relationship between a response and an explanatory variable to be described *ceteris paribus*, that is, "all other things being equal." Another phrase used in news stories is "after adjusting for ...." This is appropriate since the *all* in "all other things" is, in reality, refers only to those particular factors used as the covariates in the model. So, Dr. Meyer's foot width results might be stated in everyday language as, "After adjusting for foot width, she found no difference in the widths of girls' and boys' feet."

Not including covariates in a model amounts to "letting other things change as they will." In Latin, this is "*mutatis mutandis*." In the foot-width example, the model `width ~ sex` looks at the differences in foot width for the two sexes. However, sex is not the only thing associated with foot width. The model `width ~ sex` ignores all other factors than sex; it compares boys and girls *mutatis mutandis*, that is, letting other things change as they will. In this case, comparing boys and girls involves not just the possible differences in foot width but also the differences in other factors such as foot length and body weight.

> **Example: One change can bring another**
>
> I was once involved in a budget committee that recommended employee health benefits for the college where I worked. At the time, college employees who belonged to the college's insurance plan received a generous subsidy for their health insurance costs. Employees who did not belong to the plan received no subsidy but were given a modest monthly cash payment. After the stock market crashed in 2000, the college needed to cut budgets. One proposal called for eliminating the cash payment to employees who did not belong to the insurance plan. Proponents of the plan claimed that this would save money without reducing health benefits. I argued that this claim was an "all other things being equal" analysis: how expenditures would change assuming the number of people belonging to the insurance plan remained constant. In re-

ality, however, the policy change would play out *mutatis matandis*; the loss of the cash payment would cause some employees, who currently received health benefits through their spouse's health plan, to switch to the college's health plan. That is what happened, contributing to an overall increase in healthcare expenses.

i Example: Spending and student performance

To illustrate how covariates set context, consider an issue of interest to public policy-makers in many societies: How much money to spend on children's education? State lawmakers in the US are understandably concerned with the quality of public education provided. However, they also have other concerns and constraints and constituencies who give budget priority to other matters.

In evaluating their various trade-offs, lawmakers could benefit by knowing how increased educational spending will shape educational outcomes. What can available data tell us? Unfortunately, there are various political constraints that work against states adopting and publishing data on a standard, genuine measure of educational outcome. Instead, we have high-school graduation rates, student grades, and other non-standardized data. These data might have some meaning but can also reflect system gaming by administrators and teachers, for which there is little systematic data.

Although imperfect, college admissions tests such as the ACT and SAT provide consistent data between states. For example, Figure **??** shows the average SAT score in 2010 in each state versus expenditures per pupil in public elementary and secondary schools. Layered on top of the data is a flexible linear model (and its confidence band) of SAT score versus expenditure.

The overall impression given by the model is that the relationship is negative, with lower expenditures corresponding to higher SAT scores. However, the confidence band is broad; it is possible to find a smooth path with almost zero slope through the confidence band. Either way,

this graph does not support the conventional wisdom that higher spending produces better school outcomes.

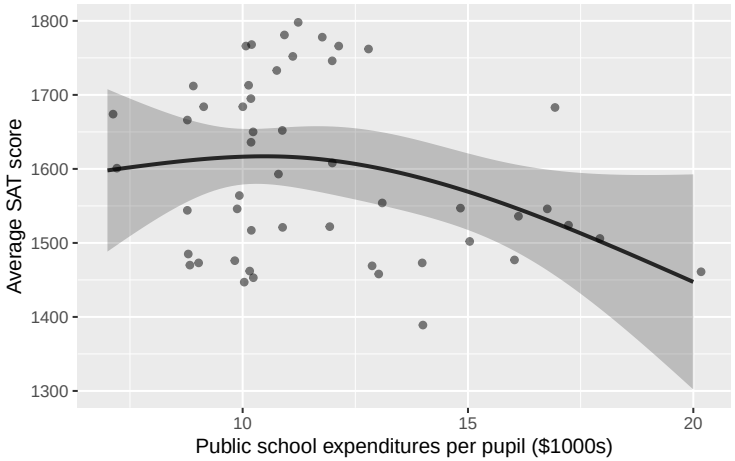


Figure 28.1: State by state data (from 2010) on average SAT college admissions test scores and expenditures for public education.

Of course, other factors play a role in shaping education outcomes: for instance, poverty levels, parental education, and how the educational money is spent (higher pay for teachers or smaller class sizes? administrative bloat?).
At first glance, it is tempting to ignore these additional factors. For instance, we may not have data on them. Moreover, as our interest is in understanding the relationship between expenditures and education outcomes, we are not directly concerned with the additional factors. However, the lack of direct concern does not imply that we should ignore the factors but that we should do what we can to "hold them constant".
To illustrate, consider the fraction of eligible students (those in their last year of high school) who take the college admission test. This fraction varies widely from state to state. In a poor state where few students go to college, the fraction can be tiny (Alabama 8%, Arkansas 5%, Mississippi 4%, Louisiana 8%). In some other states, the large majority of students take the SAT (Maine 93%, Mas-

sachusetts 89%, New York 89%). In states with low SAT participation rates, the students who take the test tend to be those applying to schools with competitive admissions. Such strong students will get high scores. In contrast, the scores in states with high participation rates reflect both strong and weak students. Consequently, the scores will be lower on average than in the low-participation states.
Putting the relationship between expenditure and SAT scores in the context of the fraction taking the SAT is accomplished with the model `SAT ~ expenditure + fraction` rather than just `SAT ~ expenditure`. Figure **??** shows a model with `fraction` as a covariate.

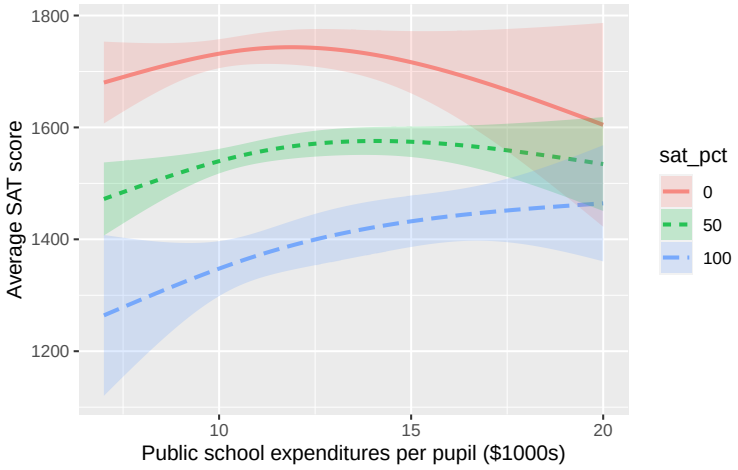


Figure 28.2: The model of SAT score versus expenditures, including as a covariate the fraction of eligible students in the state who take the SAT.

Note that the effect size of spending on SAT scores is positive when the expenditure level is less than $10,000 per pupil. Notice as well that when the fraction taking the SAT is tiny, the average scores do not depend on expenditure. This flat relationship suggests that, among elite students, state expenditure does not make a discernible difference. Perhaps the college-bound students in such states have other educational resources to draw on.
The relationship shown in Figure **??** is genuine. However,

so is the very different relationship seen in Figure **??**. How can the same data be consistent with two utterly different displays? The answer, perhaps unexpectedly, has to do with the connections among the explanatory variables. Whatever the relationship between an individual explanatory variable and the response variable, the *appearance* of that relationship will depend on which covariates the modeler chooses to include.

29 Covariates eat variance

In Lesson **??**, we introduced covariates to set the relationship between an explanatory variable and the response variable in the correct context. In Lesson 30, we will return to this context-setting role to show that the appropriate choice of covariates to include in a model depends on the modeler's opinion about the relevant structure of a DAG. Here, we will treat covariates as commodity items to show a surprising property of models. This property is a boon to the modeler, helping to enable sound decisions about whether to include any given covariate. However, it is also a pitfall lieing in wait for the wishful thinker.

29.1 How much variation is explained

We start by returning to the definition of statistical thinking introduced at the start of these Lessons:

> *Statistic thinking is the explanation or description of variation* in the context of *what remains unexplained or undescribed.*

In this Lesson, we will work with a straightforward measure of "what remains unexplained or undescribed." The fitted model values represent the explained part of the variation. The residuals are what is left over, the difference between the actual values of the response variable and the fitted model values.

As a reminder, we will construct a simple model of the list price of books as a function of the number of pages and whether the book is a paperback or hardcover.[1]

[1] If you seek to duplicate the results presented in this chapter, please note that we have deleted six rows from 'amazon_books because the rows are either duplicates or have one of the variables missing. The deleted rows are 62, 103, 205, 211, 242, and 303.

```
Price_model <- lm(list_price ~ num_pages + hard_paper,
                  data = amazon_books)
```

The `model_eval()` function can extract the fitted model values and the residuals from the model. We show just a few rows here, but we will use the entire report from `model_eval()`. Remember that when `model_eval()` is not given input values, it uses the model *training* data as input.

```
Results <- model_eval(Price_model)
```

```
Using training data as input to model_eval().
```

.response	num_pages	hard_paper	.output	.resid	.lwr	.upr
12.95	304	P	16.60	-3.65	-10.73	43.94
15.00	273	P	15.98	-0.98	-11.36	43.32
1.50	96	P	12.45	-10.95	-14.97	39.88
15.99	672	P	23.95	-7.96	-3.57	51.47
30.50	720	P	24.91	5.59	-2.67	52.49
28.95	460	H	24.57	4.38	-2.89	52.03

The first book in the training data is a 304-page paperback with a list price of $12.95. The fitted model value for that book is $16.60. (Ordinarily, we refer to the output of the model function simply as the "output" or the "model output." However, the output of the model function, when applied to rows from the *training* data also called the *fitted model value*.)

At $16.60, the fitted model value is $3.65 *higher* than the list price. This difference is the residual for that book, the sign reflecting the definition

$$\text{residual} \equiv \text{response value} - \text{fitted model value} \ .$$

When the residual is small in magnitude, the fitted model value is close to the response value. Conversely, a large residual means the model was way off target for that book.

The standard measure of the typical size of a residual is the standard deviation or, equivalently, the variance.

```
Results %>% summarize(se_resids = sd(.resid), v_resids=var(.resid))
```

se_resids	v_resids
13.81885	190.9606

As always, the standard deviation is easier to read because it has sensible units, in this case, dollars. On the other hand, the variance has strange units (square dollars) because it is the square of the standard deviation. We will use the variance for measuring the typical size of a residual for the reasons described in Lesson **??**; variances add nicely in a manner analogous to the Pythagorean Theorem.

A simple measure of how much of the variation in the response variable remains *unexplained* is the ratio of the variance of the residuals and the variance of the response variable.

```
Results %>% summarize(unexplained_fraction = var(.resid)/var(.response))
```

unexplained_fraction
0.9246907

More than 90% of the variation remains unexplained by the `Price_model`! This high fraction of unexplained variance suggests the model has little to tell us. In the spirit of putting a positive spin on things, statisticians typically work with the complement of the unexplained fraction. Since the unexplained fraction is 92.5%, the complement is 7.5%. This number is written R^2 and pronounced "R-squared." (It also has a formal name: the "coefficient of determination." In Lesson **??**, we will meet the inventor of the coefficient of determination, Sewall Wright, who is an early hero of causal reasoning.)

R^2 is such a widely used summary of how the explanatory variables account for the response variable that a software extractor calculates it and some related values.

```
Price_model %>% R2()
```

n	k	Rsquared	F	adjR2
317	2	0.0753093	12.78651	0.0694196

Many modelers act as if their goal is to build a model that makes R^2 as big as possible. Their thinking is that large R^2 means that the explanatory variables account for much of the response variable's variance. Unfortunately, it is a naive goal. Instead, always focus on the model's suitability for the purpose at hand. Often, shooting for a large R^2 imposes costs that can undermine the purpose for the model. Furthermore, even models with the largest possible R^2 sometimes have nothing to say about the response variable.

29.2 Getting to 1

R^2 can range from zero to one. Zero means that the model accounts for *none* of the variation in the response variable. We can construct such a model quickly enough: `list_price ~ 1` has no explanatory variables and, therefore, no ability to distinguish one book from another.

```
Null_model <- lm(list_price ~ 1, data = amazon_books)
Null_model %>% R2()
```

n	k	Rsquared	F	adjR2
319	0	0	NaN	0

We are using the word "null" to name this model. "Null" is part of the statistics tradition. The dictionary definition of "null" is "having or associated with the value zero" or "lacking distinctive qualities; having no positive substance or content."[2]

In the null model, the fitted model values are all the same; all the variation is in the residuals.

[2]Source: Oxford Languages

```
Null_model %>% model_eval()
```

```
Using training data as input to model_eval().
```

```
Using training data as input to model_eval().
```

.response	.output	.resid	.lwr	.upr
12.95	18.6	-5.65	-9.69	46.89
15.00	18.6	-3.60	-9.69	46.89
1.50	18.6	-17.10	-9.69	46.89
15.99	18.6	-2.61	-9.69	46.89
30.50	18.6	11.90	-9.69	46.89
28.95	18.6	10.35	-9.69	46.89

At the other extreme, where $R^2 = 1$, the explanatory variables account for every bit of variation in the response variable. We can try various combinations of explanatory variables to see if we can accomplish this. For example, `publisher` explains 67% of the variation in list price.

```
lm(list_price ~ publisher, data = amazon_books) %>% R2()
```

n	k	Rsquared	F	adjR2
319	158	0.6749786	2.103008	0.3540199

We can also check whether `author` has anything to say about the list price.

```
lm(list_price ~ author, data = amazon_books) %>% R2()
```

n	k	Rsquared	F	adjR2
319	250	0.9434046	4.534044	0.7353333

Incredible! How about if we use *both* `publisher` and `author` as explanatory variables? We get very close to $R^2 = 1$.

```
lm(list_price ~ publisher + author, data = amazon_books) %>%
  R2()
```

n	k	Rsquared	F	adjR2
319	281	0.9821609	7.249441	0.84668

The modeler discovering this tremendous explanatory power of `publisher` and `author` can be forgiven for thinking he or she has found a meaningful explanation. But, unfortunately, the high R^2 is an illusion in this case.

To see why, consider another possible explanatory variable, the International Standard Book Number (ISBN). The ISBN is a ten- or thirteen-digit number that marks each book with a unique number.

Figure 29.1: The ISBN number from one of the Project MOSAIC textbooks.

There is a system behind ISBNs, but despite the "N" standing for "number," an ISBN is a character string or word (written using only digits). Consequently, the `isbn_10` variable in `amazon_books`is categorical.

```
ISBN_model <- lm(list_price ~ isbn_10, data = amazon_books)
ISBN_model %>% R2()
```

n	k	Rsquared	F	adjR2
319	318	1	NaN	NaN

The `isbn_10` explains all variation in the list price!

Given that the ISBN is, as we have said, an arbitrary sequence of characters, why does it do such a good job of accounting for the list price? The answer lies not in the content of the ISBN but in another fact: each book has a unique ISBN. As well, each book has a single price. So the ISBN identifies the price of each book. Cleverness is not involved; the list price could be anything, and the ISBN would still identify it precisely. The model coefficients store the whole set of ISBNs and the corresponding set of list prices.

We can substantiate the claim just made—that the list price could be anything at all—by synthesizing a data frame with random list prices:

```
amazon_books %>%
  mutate(random_list_price = rnorm(nrow(.))) %>%
  lm(random_list_price ~ isbn_10, data = .) %>%
  R2()
```

n	k	Rsquared	F	adjR2
319	318	1	NaN	NaN

Similar randomization can be accomplished by *shuffling* the `isbn_10` column of the data frame so that each ISBN points to a random book. Of course, such shuffling destroys the link between the ISBN and the list price. Even so, the R^2 remains high.

```
lm(list_price ~ shuffle(isbn_10), data=amazon_books) %>% R2()
```

n	k	Rsquared	F	adjR2
319	318	1	NaN	NaN

```
lm(shuffle(list_price) ~ isbn_10, data=amazon_books) %>% R2()
```

n	k	Rsquared	F	adjR2
319	318	1	NaN	NaN

Statistical nomenclature is obscure here. So we will make up a name for such incidental alignment with no true explanatory power: the "**ISBN-effect**."

Statistical thinkers know to be aware of situations where categorical variables have many levels and check whether the ISBN effect is in play.

29.3 The ISBN effect as a benchmark

Shuffling an explanatory variable (while keeping the response variable in the original order) voids any possible explanatory connection between the two. An R^2=0, as we get from any model of the form `y ~ 1`, signals that the `1` cannot account for any variation. However, this does not mean shuffling will lead to $R^2 = 0$. Instead, there is a systematic relationship between the number of model coefficients associated with the shuffled variable, the sample size n, and R^2.

We can demonstrate this relationship by conducting many trials of modeling the `list_price` with a shuffled explanatory variable: either `publisher`, `author`, or `isbn_10`.

⚠ Demonstration: Counting coefficients

The `amazon_books` data frame has $n = 319$ rows.[a] In the next computing chunk, we fit the model `list_price ~ publisher` and collect the coefficients for counting:

```
Publisher_model <- lm(list_price ~ shuffle(publisher),
                      data=amazon_books)
Coefficients <- Publisher_model %>% coef() %>% data.frame()
nrow(Coefficients)
```

```
[1] 159
```

There are 161 coefficients in the model, the first one being the "Intercept." We will show only the first few.

```
Coefficients %>% head()
```

	value
(Intercept)	14.95
shuffle(publisher)Adams Media	0.05
shuffle(publisher)Akashic Books	13.00
shuffle(publisher)Aladdin	15.05
shuffle(publisher)Albert Whitman & Company	-0.95
shuffle(publisher)Alfred A. Knopf	0.05

Altogether, there are $k = 160$ coefficients relating to `shuffle(publisher)`.

[a]The data frame in the `moderndive` package has six additional rows, which we have deleted as duplicates or because of missing data.

The theory relating R^2 to the number of coefficients associated is straightforward for shuffled explanatory variables: R^2 will be random with mean value $\frac{k}{n-1}$.

⚠ Demonstration: The mean R^2 across many trials

For the `shuffle(publisher)` model, the theoretical mean across many trials will be $R^2 = 158/324 = 0.49$. The demonstration below confirms this using 100 trials:

```
Pub_trials <- do(100) * {
  lm(list_price ~ shuffle(publisher), data=amazon_books) %>%
    R2()
}
Pub_trials %>% summarize(meanR2 = mean(Rsquared))
```

meanR2
0.4955462

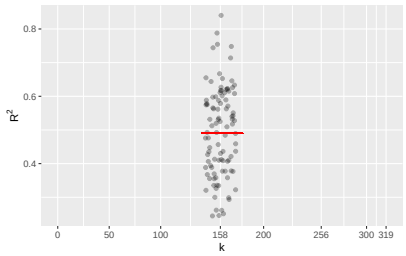


Figure 29.2: 100 trials of R^2 from `list_price ~ shuffled(publisher)`. The theoretical value $k/n = 160/324 = 0.49$ is marked in red.

We can carry out similar trials for the models `list_price ~ shuffle(author)` and `list_price ~ shuffle(isbn_10)`, which have $k = 251$ and $k = 319$ respectively.

The blue diagonal line in Figure **??** shows the theoretical average R^2 as a function of the number of model coefficients when the explanatory variable is randomized. R^2 will always be 1.0 when $k = n$, that is, when the number of coefficients is the same as the sample size.

Figure **??** suggests a way to distinguish between R^2 resulting from the ISBN-effect and R^2 that shows some true explanatory power: Check if R^2 is substantially above the blue diagonal

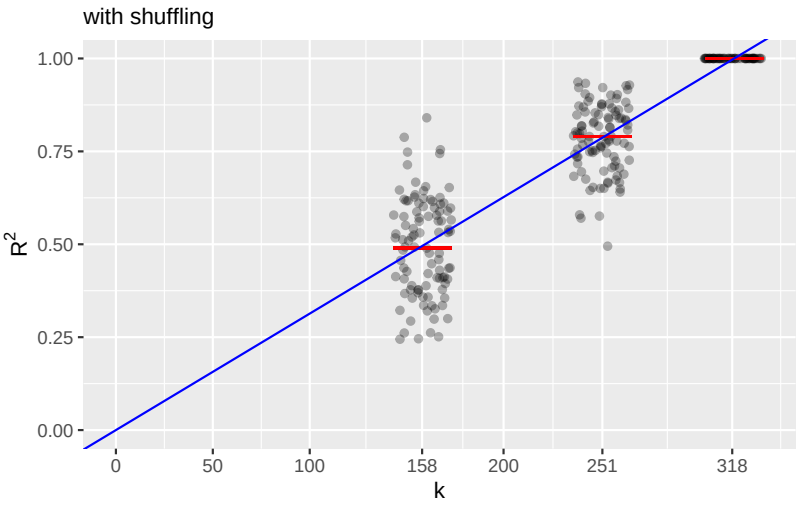


Figure 29.3: R^2 from many trials of three models, `list_price ~ shuffle(publisher)` and `~ shuffle(author)` and `~shuffle(isbn_10)`.

line, that is, check if $R^2 \gg \frac{k}{n-1}$ where k is the number of model coefficients.

29.4 The F statistic

k and n provide the necessary context for proper interpretation of R^2; all three numbers are needed to establish whether $R^2 \gg \frac{k}{n-1}$ to rule out the ISBN effect. The calculation is not difficult; the modeler always knows the size n of the training data and can find k as the number of coefficients in the model (not counting the Intercept term).

Perhaps a little easier than interpreting R^2 is the interpretation of another statistic, named F, which folds in the k, n, and R^2 into a single number:

$$F \equiv \frac{n-k-1}{k} \frac{R^2}{1-R^2}$$

Figure **??** is a remake of Figure **??** but using F instead of R^2. The blue line, which had the formula $R^2 = k/(n-1)$ in Figure **??**, gets translated to the constant value 1.0 in Figure **??**, regardless

of k. To decide when a model points to a connection stronger than the ISBN effect, the threshold F > 3 is a good rule of thumb. (Lesson **??** introduces a more precise calculation for the F threshold, which is built into statistical software and presented as a "**p-value**.")

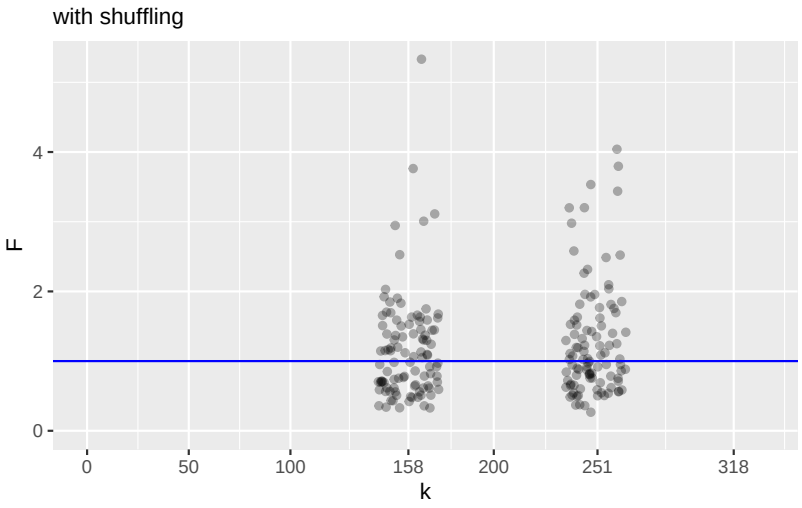


Figure 29.4: Like Figure **??**, but using the F statistic to summarize each trial.

Adjusted R^2

Some fields, notably economics, prefer an alternative to F called "**adjusted R^2**" (or R^2_{adj}). The adjustment comes from moving the raw R^2 downward and leftward, more-or-less in the direction of the blue line in Figure **??**. This movement adjusts a raw R^2 that lies on the blue line to $R^2_{\text{adj}} = 0$.

We leave the debate on the relative merits of using F or R^2_{adj} their respective boosters. However, before getting wrapped up in such debates, it is worth pointing out that R^2_{adj} is just a rescaling of F.

$$R^2_{\text{adj}} = 1 - \frac{n-1}{k}\frac{R^2}{F} \ .$$

29.5 Comparing models

Modelers are often in the position of having a model that they like but are contemplating adding one or more additional explanatory variables. To illustrate, consider the following models:

- Model 1: `list_price ~ 1`
- Model 2: `list_price ~ 1 + hard_paper`
- Model 3: `list_price ~ 1 + hard_paper + num_pages`
- Model 4: `list_price ~ 1 + hard_paper + num_pages + weigh`

Figure 29.5: Nesting Russian dolls

All the explanatory variables in the smaller models also apply to the bigger models. Such sets are said to be "**nested**" in much the same way as for Russian dolls.

For a nested set of models, R^2 can never decrease when moving from a smaller model to a larger one—almost always, there is an increase in R^2. To demonstrate:

```
amazon_books <- amazon_books %>%
  select(list_price, weight_oz, num_pages, hard_paper) %>%
  filter(complete.cases(.))
model1 <- lm(list_price ~ 1, data=amazon_books)
model2 <- lm(list_price ~ 1 + weight_oz, data = amazon_books)
model3 <- lm(list_price ~ 1 + weight_oz + num_pages, data=amazon_books)
model4 <- lm(list_price ~ 1 + weight_oz + num_pages + hard_paper, data=amazon_books)
```

```
R2(model1)
```

n	k	Rsquared	F	adjR2
309	0	0	NaN	0

```
R2(model2)
```

n	k	Rsquared	F	adjR2
309	1	0.16	57	0.15

```
R2(model3)
```

n	k	Rsquared	F	adjR2
309	2	0.17	30	0.16

```
R2(model4)
```

n	k	Rsquared	F	adjR2
309	3	0.17	21	0.16

When adding explanatory variables to a model, a good question is whether the new variable(s) add to the ability to account for the variability in the response variable. R^2 never goes down when moving from a smaller to a larger model, so we cannot rely on the increase in R^2. A valuable technique called "**Analysis of Variance**" (ANOVA for short) looks at the incremental change in variance explained from a smaller model to a larger one. The increase can be presented as an F statistic. To illustrate:

```
anova_summary(model1, model2, model3, model4)
```

term	df.residual	rss	df	sumsq	statistic
list_price ~ 1	308	54531	NA	NA	NA
list_price ~ 1 + weight_oz	307	46032	1	8499	57.2
list_price ~ 1 + weight_oz + num_pages	306	45466	1	566	3.8
list_price ~ 1 + weight_oz + num_pages + hard_paper	305	45277	1	189	1.3

Focus on the column named `statistic`. This records the F statistic. The move from Model 1 to Model 2 produces F=57, well above the threshold described above and clearly indicating that the `weight_oz` variable accounts for some of the list price. Moving from Model 2 to Model 3 creates a much less impressive F of 3.8. It is as if the added explanatory variable, `num_pages`, is just barely pulling its own "weight." Finally, moving from Model 3 to Model 4 produces a below-threshold F of 1.3. In other words, in the context of `weight_oz` and

`num_pages`, the `hard_paper` variable does not carry additional information about the list price.

The last column of the report, labeled `Pr(>F)`, translates F into a universal 0 to 1 scale called a p-value. A large F produces a small p-value. The rule of thumb for reading p-values is that a value $p < 0.05$ indicates that the added variable brings new information about the response variable. We will return to p-values and the controversy they have entailed in Lessons 36 through 38.

30 Confounding

Many people are concerned that the chemicals used by lawn-greening companies are a source of cancer or other illness. Imagine designing a study that could confirm or refute this concern. The study would sample households, some with a history of using lawn-greening chemicals and others that have never used them. The question for the study designers: What variables to record?

An obvious answer: record both chemical use and a measure of health outcome, say whether anyone in that household has developed cancer in the last five years. We will suppose that the two possible levels of grass treatment are "organic" or "chemicals." As for illness, the levels will be "cancer" or "not."

Here are two very simple DAGs describing possible theories:

illness $\leftarrow$ grass treatment or illness $\rightarrow$ grass treatment

The DAG on the left expresses the belief among people who think chemical grass treatment might cause cancer. But belief is not necessarily reality, so we should consider the right-hand DAG. For example, one way to avoid the possibility of illness $\rightarrow$ grass treatment is to include only households where cancer (if any) started *after* the grass treatment. Note that we are not ignoring the right-hand DAG; we are using the study design to disqualify it.

The statistical thinker knows that covariates are important. But which covariates? Answering that requires knowing a lot about the "domain," that is, how things connect in the world. Such knowledge helps in thinking about the bigger picture and, in particular, possible covariates that connect plausibly to the

response variable and the primary explanatory variable, grass treatment.

For now, suppose that the study designers have not yet become statistical thinkers and have rushed out to gather data on illness and grass treatment. Here are a few rows from the data (which we have simulated for this example):

grass	illness
organic	not
chemicals	not
chemicals	not
chemicals	not
organic	not
organic	not
organic	not
organic	not
chemicals	cancer
organic	not

Analyzing such data is straightforward. First, check the overall cancer rate:

```
# overall cancer rate
lm(zero_one(illness, one="cancer") ~ 1, data = Cancer_data) %>% coef()
```

```
(Intercept)
      0.026
```

In these data, 2.6% of the sampled households had cancer in the last five years. How does the grass treatment affect that rate?

```
mod <- lm(zero_one(illness, one="cancer") ~ grass, data = Cancer_data)
coefficients(mod)
```

```
 (Intercept) grassorganic
  0.01246883   0.02258960
```

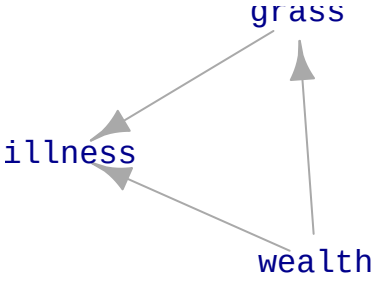
For households whose lawn treatment is "organic," the risk of cancer is higher by 2.3 percentage points compared to households that treat their grass with chemicals. We were expecting the reverse, but it is what the data show. On the other hand, there is sampling variability to take into account. Look at the confidence intervals:

```
conf_interval(mod)
```

term	.lwr	.upr
(Intercept)	-0.0031034	0.028041
grassorganic	0.0024692	0.042710

The confidence interval on `grassorganic` does not include zero, but it comes close. So might the chemical treatment of grass be protective against cancer? Only at this point do the study designers do what they should have from the start: think about covariates.

One theory—just a theory—is this: Green grass is not a necessity, so the households who treat their lawn with chemicals tend to have money to spare. Wealthier people also tend to have better health, partly because of better access to health care. Another factor is that wealthier people can live in less polluted neighborhoods and are less likely to work in dangerous conditions, such as exposure to toxic chemicals. Such a link between wealth and illness points to a DAG hypothesis where "`wealth`" influences how the household's `grass` is treated and `wealth` similarly influences the risk of developing `cancer`. Like this:



A description of this structure of causality is, "The effect of grass treatment on illness is **confounded** by wealth."

The Oxford Languages dictionary offers two definitions of "confound."

1. *Cause surprise or confusion in someone, especially by acting against their expectations.*
2. *Mix up something with something else so that the individual elements become difficult to distinguish.*

This second definition carries the statistical meaning of "confound."

The first definition seems relevant to our story since the protagonist expected that chemical use would be associated with higher cancer rates and was surprised to find otherwise. But, the statistical thinker does not throw up her hands when dealing with mixed-up causal factors. Instead, she uses modeling techniques to untangle the influences of various factors.

Using covariates in models is one such technique. Our wised-up study designers go back to collect a covariate representing household wealth. Here is a glimpse at the updated data.

wealth	grass	illness
1.4283990	organic	not
0.0628559	chemicals	not
0.4382804	chemicals	not
0.6084487	chemicals	not
0.8033695	organic	not
-0.9367287	organic	not
0.6664468	organic	not
-1.2445977	organic	not
-1.3194594	chemicals	cancer
-1.6162391	organic	not

Having measured `wealth`, we can use it as a covariate in the model of `illness`:

```
lm(zero_one(illness, one="cancer") ~ grass + wealth, data = Cancer_data) %>%
  conf_interval()
```

term	.lwr	.upr
(Intercept)	0.0246811	0.0574819
grassorganic	-0.0450811	-0.0009699
wealth	-0.0568093	-0.0356454

With `wealth` as a covariate, the model shows that (all other things being equal) "organic" lawn treatment reduces cancer risk. However, we do not see this directly from the `grass` and `illness` variables because all other things are not equal: wealthier people are more likely to use chemical lawn treatment. (Keep in mind that this is **simulated data**. Do not conclude from this example anything about the safety of the chemicals used for lawn greening.)

i Example: The flu vaccine

As you know, people are encouraged to get vaccinated before flu season. This recommendation is particularly emphasized for older adults, say, 60 and over.
In 2012, the *Lancet*, a leading medical journal, published a systematic examination and comparison of many previous studies. The *Lancet* article describes a hypothesis that existing flu vaccines may not be as effective as was originally found.

> *A series of observational studies undertaken between 1980 and 2001 attempted to estimate the effect of seasonal influenza vaccine on rates of hospital admission and mortality in [adults 65 and older]. Reduction in all-cause mortality after vaccination in these studies ranged from 27% to 75%. In 2005, these results were questioned after reports that increasing vaccination in people aged 65 years or older did not result in a significant decline in mortality. Five different research groups in three countries have shown that these early observational studies had substantially overestimated the mortality benefits in this age group because of unrecognized confounding. This error has*

Such a study of ea
called a *meta-anal*

been attributed to a healthy vaccine recipient effect: reasonably healthy older adults are more likely to be vaccinated, and a small group of frail, undervaccinated elderly people contribute disproportionately to deaths, including during periods when influenza activity is low or absent.

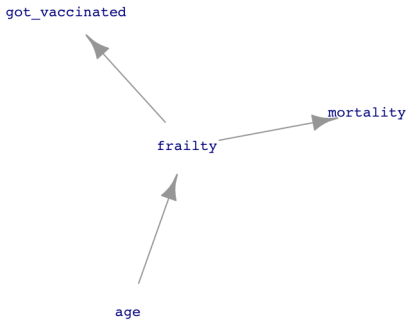
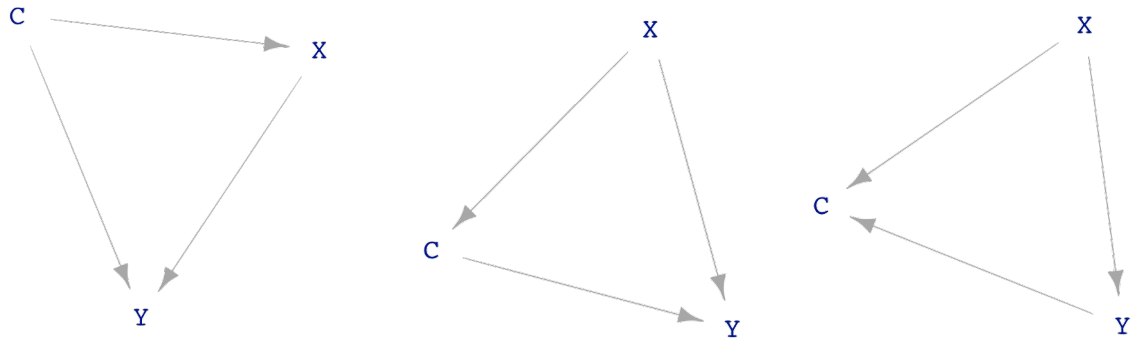


Figure 30.1: A DAG diagramming the "healthy vaccine recipient" effect

Figure **??** presents a network of causal influences that could shape the "healthy vaccine recipient." People are more likely to become frail as they get older. Frail people are *less* likely to get vaccinated, but more likely to die in the next few months. The result is that vaccination is associated with reduced mortality, even if there is no direct link between vaccination and mortality.

30.1 Block that path!

Let us look more generally at the possible causal connections among three variables, which we will call X, Y, and C. We will stipulate that X points causally toward Y and that C is a possible covariate. Like all DAGs, there cannot be a cycle of causation. These conditions leave three distinct DAGs that do not have a cycle, shown in Figure **??**.



(a) C is a confounder. (b) C is a mechanism. (c) C is a consequence.

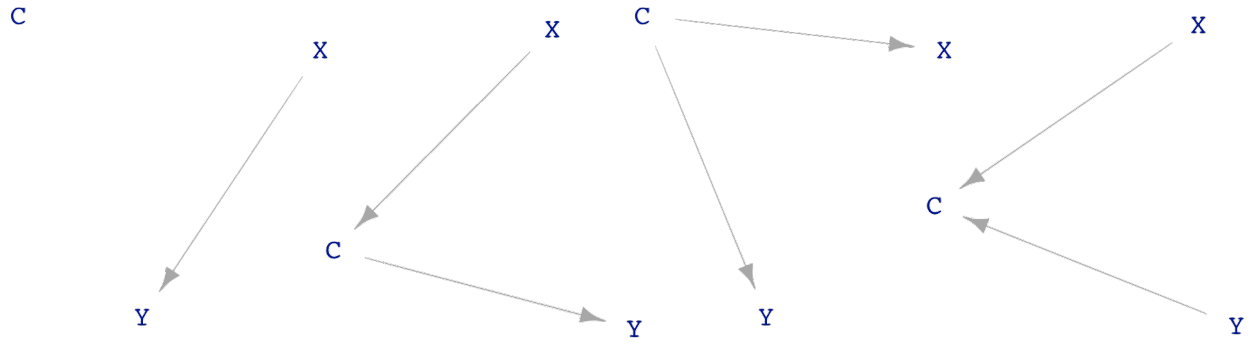
Figure 30.2: Three different DAGs connecting X, Y, and C.

C plays a different role in each of the three dags. In sub-figure (a), C causes both X and Y. In (b), part of the way that X influences Y is *through* C. We say, in this case, "C is a mechanism by which X causes Y. In sub-figure (c), C does not cause either X or Y. Instead, C is a consequence of both X and Y.[1]

To understand how a DAG informs whether or not to include a covariate, It will help to give general names to some of the sub-structures seen in the Figure **??** DAGs. **?@fig-dag-paths** shows some of these sub-structures, removing other links that are not part of the structure.

- A "**direct causal link**" between X and Y. There are no intermediate nodes.
- A "**causal path**" from C to X and on to Y. A causal path is one where, starting at the originating node, flow along the arrows can get to the terminal node, passing through all intermediate nodes.
- A "**correlating path**" from Y through X to C. Correlating paths are distinct from causal paths because, in a correlating path, there is no way to get from one end to the other by following the flows.

[1] In any given real-world context, good practice calls for considering each possible DAG structure and concocting a story behind it. Such stories will sometimes be implausible, but there can also be surprises that give the modeler new insight.



(a) Direct causal link from X to Y (b) Causal path from X through C to Y (c) Correlating path connecting X and Y via C (d) C as a consequence of X and Y

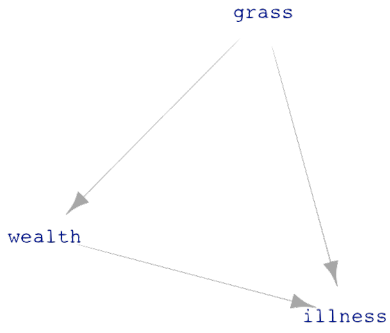
Figure 30.3: Sub-structures seen in Figure **??**.

- A "**collider**" `wealth`. In other words, both X and Y are causes of C.

Look back to Figure **??**(a), where `wealth` is a confounder. A confounder is always an intermediate node in a *correlating path*.

Including a covariate either blocks or opens the pathway on which that covariate lies. Which it will be depends on the kind of pathway. A causal path, as in Figure **??**(b), is blocked by including the covariate. Otherwise, it is open. A correlating path (**?@fig-dags-path**(c)) is similar: the path is open unless the covariate is included in the model. A colliding path, as in Figure **??**(d), is blocked *unless* the covariate is included—the opposite of a causal path.

Often, covariates are selected to block all paths except the direct link between the explanatory and response variable. This means *do* include the covariate if it is on a correlating path and *do not* include it if the covariate is at the collision point.



As for a causal path, the choice depends on what is to be studied. Consider the DAG drawn in Figure **??**(b), reproduced here for convenience:

`grass` influences `illness` through two distinct paths:

i. the direct link from `grass` to `illness`.

ii. the causal pathway from `grass` through `wealth` to `illness`.

Admittedly, it is far-fetched that choosing to green the grass makes a household wealthier, but focus on the topology of the DAG and not the unlikeliness of this specific causal scenario.

There is no way to block a direct link from an explanatory variable to a response. If there were a reason to do this, the modeler probably selected the wrong explanatory variable.

But there is a genuine choice to be made about whether to block pathway (ii). If the interest is the purely biochemical link between grass-greening chemicals and illness, then block pathway (ii). However, if the interest is in the *total* effect of `grass` and `illness`, including both biochemistry and the sociological reasons why `wealth` influences `illness`, then leave the pathway open.

> ⚠ In draft: Some resources
>
> https://towardsdatascience.com/causal-effects-via-dags-801df31da794
> https://towardsdatascience.com/causal-effects-via-the-do-operator-5415aefc834a

31 Spurious correlation

Google NGram provides a quick way to track word usage in books over the decades. Figure **??** shows the NGram for three statistical words: coefficient, correlation, and regression.

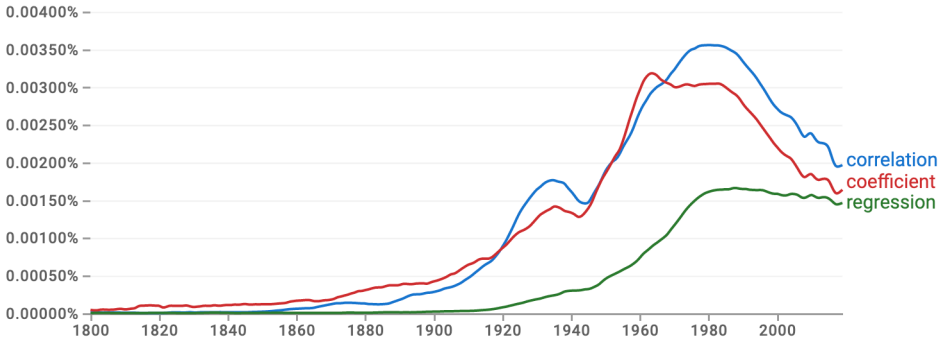


Figure 31.1: Google NGram for "coefficient," "correlation," and "regression."

The use of "correlation" started in the mid to late 1800s, reached an early peak in the 1930s, then peaked again around 1980. "Correlation" is tracked closely by "coefficient." This parallel track might seem evident to historians of statistics; the quantitative measure called the "**correlation coefficient**" was introduced by Francis Galton in 1888 and quickly became a staple of statistics textbooks.

In contrast to mainstream statistics textbooks, "correlation" barely appears in these lessons (until this chapter). There is a good reason for this. Although the correlation coefficient measures the "strength" of the relationship between two variables, it is a special case of a more general and powerful method that appears throughout these Lessons: regression modeling.

Figure **??** shows that "regression" got a later start than correlation. That is likely because it took 30-40 years before it was appreciated that correlation could be generalized. Furthermore, regression is more mathematically complicated than cor-

relation, so practical use of regression relied on computing, and computers started to become available only around 1950.

31.1 Correlation

A dictionary is a starting point for understanding the use of a word. Here are four definitions of "correlation" from general-purpose dictionaries.

> "*A relation existing between phenomena or things or between mathematical or statistical variables which tend to vary, be associated, or occur together in a way not expected on the basis of chance alone*" Source: Merriam-Webster Dictionary

> "*A connection between two things in which one thing changes as the other does*" Source: Oxford Learner's Dictionary

> "*A connection or relationship between two or more things that is not caused by chance. A positive correlation means that two things are likely to exist together; a negative correlation means that they are not.*" Source: Macmillan dictionary

> "A mutual relationship or connection between two or more things," "interdependence of variable quantities." Source: [Oxford Languages]

All four definitions use "connection" or "relation/relationship." That is at the core of "correlation." Indeed, "relation" is part of the word "correlation." One of the definitions uses "causes" explicitly, and the everyday meaning of "connection" and "relation" tend to point in this direction. The phrase "one thing changes as the other does" is close to the idea of causality, as is "interdependence.:

Three of the definitions use the words "vary," "variable," or "changes." The emphasis on variation also appears directly in a close statistical synonym for correlation: "covariance."

Two of the definitions refer to "chance," that correlation "is not caused by chance," or "not expected on the basis of chance alone." These phrases suggest to a general reader that correlation, since not based on chance, must be a matter of fate: pre-determination and the action of causal mechanisms.

We can put the above definitions in the context of four major themes of these Lessons:

- Quantitative description of relationships
- Variation
- Sampling variation
- Causality

Correlation is about relationships; the "correlation coefficient" is a way to describe a straight-line relationship quantitatively. The correlation coefficient addresses the tandem variation of quantities, or, more simply stated, how "one thing changes as the other does."

To a statistical thinker, the concern about "chance" in the definitions is not about fate but reliability. Sampling variation can lead to the appearance of a pattern in some samples of a process that is not seen in other samples of that same process. Reliability means that the pattern will appear in a large majority of samples.

Note

One of the better explanations of "correlation" appears in an 1890 article by Francis Galton, who invented the correlation coefficient. Since the explanation is more than a century old, some words will be unfamiliar to the modern reader. For example, a "clerk" is an office worker. An "omnibus" is merely a means of public transportation today.

> *Two clerks leave their office together and travel homewards in the same and somewhat unpunctual omnibus every day. They both get out of the omnibus at the same halting-place, and thence both walk by their several ways to their*

> *respective homes. … The upshot is that when either clerk arrives at his home later than his average time, there is some reason to expect that the other clerk will be late also, because the retardation of the first clerk may have been wholly or partly due to slowness of the omnibus on that day, which would equally have retarded the second clerk. Hence their unpunctualities are related. If the omnibus took them both very near to their homes, the relation would be very close. If they lodged in the same house and the omnibus dropped them at its door, the relation would become identity.*
>
> *The problems of … correlation deal wholly with departures or variations ; they pay no direct regard to the central form from which the departures or variations are measured. If we were measuring statures, and had made a mark on our rule at a height equal to the average height of the race of persons whom we were considering, then it would be the distance of the top of each man's head from that mark, upward or downward as the case might be, that is wanted for our use, and not its distance upward from the ground.*[a]
>
> [a] Francis Galton (1890) "Kinship and Correlation" *The North American Review* 150(401) URL

31.2 Spurious causation

The "Spurious correlations" website http://www.tylervigen.com/spurious-correlations provides entertaining examples of correlations gone wrong. The running gag is that the two correlated variables have no reasonable association, yet the correlation coefficient is very close to its theoretical maximum of 1.0. Typically, one of the variables is morbid, as in Figure **??**.

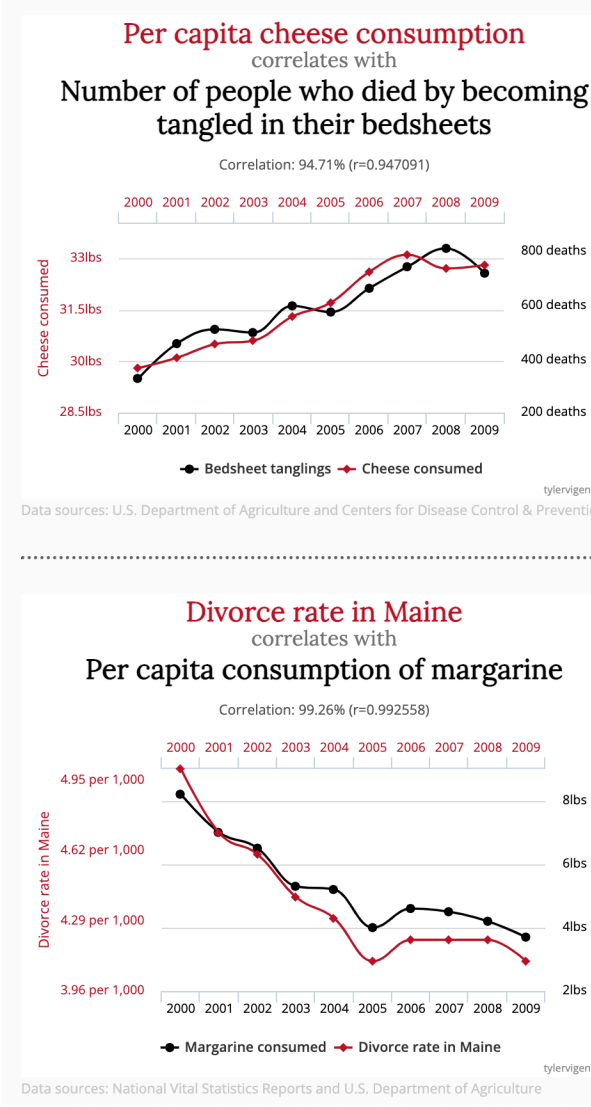
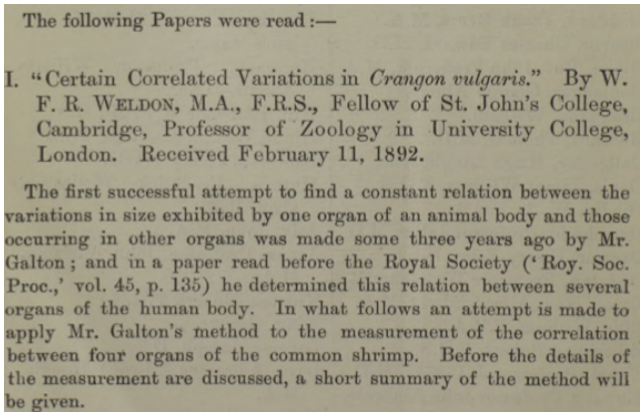


Figure 31.2: Two examples from the Spurious correlations website

According to Aldrich (1995)^[John Aldrich (1994) "Correlations Genuine and Spurious in Pearson and Yule" *Statistical Science* 10(4) URL the idea of **spurious correlations** appears first in an 1897 paper by statistical pioneer and philosopher of science Karl Pearson. The correlation coefficient method was published only in 1888, and, understandably, early users encountered pitfalls. One very early user, W.F.R. Weldon, published a study in 1892 on the correlations between the sizes of organs, such as the tergum and telson in shrimp. (See Figure **??**.)



The following Papers were read :—

I. "Certain Correlated Variations in *Crangon vulgaris*." By W. F. R. Weldon, M.A., F.R.S., Fellow of St. John's College, Cambridge, Professor of Zoology in University College, London. Received February 11, 1892.

The first successful attempt to find a constant relation between the variations in size exhibited by one organ of an animal body and those occurring in other organs was made some three years ago by Mr. Galton; and in a paper read before the Royal Society ('Roy. Soc. Proc.,' vol. 45, p. 135) he determined this relation between several organs of the human body. In what follows an attempt is made to apply Mr. Galton's method to the measurement of the correlation between four organs of the common shrimp. Before the details of the measurement are discussed, a short summary of the method will be given.

Pearson noticed a distinctive feature of Weldon's method. Weldon measured the tergum and telson as a fraction of the overall body length.

Figure **??** shows one possible DAG interpretation where `telson` and `tergum` are *not* connected by any causal path. Similarly, `length` is exogenous with no causal path between it and either `telson` or `tergum`.

```
shrimp_dag <- dag_make(
  tergum ~ unif(min=2, max=3),
  telson ~ unif(min=4, max=5),
  length ~ unif(min=40, max=80),
  x ~ tergum/length + exo(.01),
  y ~ telson/length + exo(.01)
)
# dag_draw(shrimp_dag, seed=101, vertex.label.cex=1)
knitr::include_graphics("www/telson-tergum.png")
```

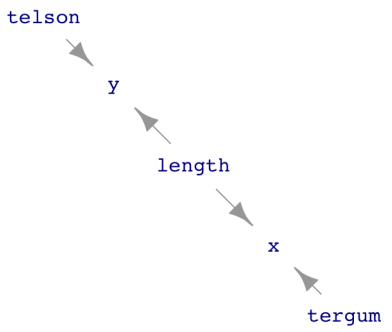


Figure 31.4: DAG for the shrimp measurements.

The Figure ?? shows a hypothesis where there is no causal relationship between telson and tergum. Pearson wondered whether dividing those quantities by `length` to produce variables `x` and `y`, might induce a correlation. Weldon had found a correlation coefficient between `x` and `y` of about 0.6. Pearson estimated that dividing by `length` would induce a correlation between `x` and `y` of about 0.4-0.5, even if telson and tergum are not causally connected.

We can confirm Pearson's estimate by sampling from the DAG and modeling `y` by `x`. The confidence interval on `x` shows a relationship between `x` and `y`. In 1892, before the invention of regression, the correlation coefficient would have been used. In retrospect, we know the correlation coefficient is a simple scaling of the `x` coefficient.

```
Sample <- sample(shrimp_dag, size=1000)
lm(y ~ x, data=Sample) %>% conf_interval()
```

term	.lwr	.upr
(Intercept)	0.0457665	0.0522715
x	0.6147549	0.7566114

```
cor(y ~ x, data=Sample)
```

```
[1] 0.514812
```

Pearson's 1897 work precedes the earliest conception of DAGs by three decades. An entire century would pass before DAGs came into widespread use. However, from the DAG of Figure ??] in front of us, we can see that `length` is a common cause of `x` and `y`.

Within 20 years of Pearson's publication, a mathematical technique called "**partial correlation**" was in use that could deal with this particular problem of spurious correlation. The key is that the model should include `length` as a covariate. The covariate correctly blocks the path from `x` to `y` via `length`.

```
lm(y ~ x + length, data=Sample) %>% conf_interval()
```

term	.lwr	.upr
(Intercept)	0.1507687	0.1635108
x	-0.0362598	0.0833543
length	-0.0013975	-0.0012508

The confidence interval on the `x` coefficient includes zero once `length` is included in the model. So the data, properly analyzed, show no correlation between telson and tergum.

In this case, "spurious correlation" stems from using an inappropriate method. This situation, identified 130 years ago and addressed a century ago, is still a problem for those who use the correlation coefficient. Although regression allows the incorporation of covariates, the correlation coefficient does not.

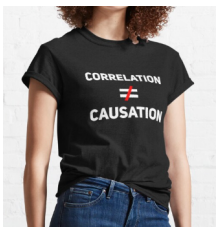
Time series analysis

Some spurious correlations, such as those presented on the eponymous website, can also be attributed to methodological error.
One source of error was identified in 1904 by F.E. Cave-Browne-Cave in her paper "On the influence of the time factor on the correlation between the barometric heights at stations more than 1000 miles apart," published in the Proceedings of the Royal Society. "Miss Cave," as she was referred to in 1917 and 1921, respectively by eminent statisticians William Sealy Gosset (who published under the name "Student") and George Udny Yule, also offered a solution to the problem. Her solution is very much in the tradition of "**time-series analysis**," a contemporary specialized area of statistics.
The unlikeliness of the correlations on the website is another clue to their origin as methodological. Nobody woke up one morning with the hypothesis that cheese consumption and bedsheet mortality are related. Instead, the correlation is the product of a search among many miscellaneous records. Imagine that data were available on 10,000

annually tabulated variables for the last decade. These 10,000 variables create the opportunity for 50 million pairs of variables. Even if none of these 50 million pairs have a genuine relationship, sampling variation will lead to some of them having a strong correlation coefficient.
In statistics, such a blind search is called the "multiple comparisons problem." Ways to address the problem have been available since the 1950s. (We will return to this topic under the label "false discovery" in Lesson **??**.) Multiple comparisons can be used as a trick, as with the website. However, multiple comparisons also arise naturally in some fields. For example, in molecular genetics, "microarrays" make a hundred thousand simultaneous measurements of gene expression. Correlations in the expression of two genes give a clue to cellular function and disease. With so many pairs available, multiple comparisons will be an issue.

31.3 "Correlation implies causation."

Francis Galton's 1890 example of the clerks on the bus introduces "correlation" as a causality story. The bus trip causes variation in commute times. Two clerks riding the same bus will have correlated commute times. In the dictionary definitions of "correlation" at the start of the Lesson, the words "connection," "relationship," and "interdependence" suggests causal connections.



Insofar as the dictionary definitions of correlation suggest a causal relationship, they are at odds with the statistical mainstream, which famously holds that "correlation does not imply causation." This view is so entrenched that it appears on tee shirts, one style of which is available for sale by the American Statistical Association.

The statement "A is not B" can be valid only if we know what A and B are. We have a handle on the meaning of "correlation." So what is the meaning of "causation?"

Dictionaries define "causation" using the word "cause." So we look there for guidance.

> A person or thing that gives rise to an action, phenomenon, or condition. Source: Oxford Languages
>
> An event, thing, or person that makes something happen. Source: Macmillan Dictionary
>
> A person or thing that acts, happens, or exists in such a way that some specific thing happens as a result; the producer of an effect. Source: Dictionary.com

Interpreting these definitions requires making sense of "give rise to," "makes happen," or "happens as a result." All of them are synonyms for "cause."

This circularity produces a muddle. Centuries of philosophical debate have yet to clarify things much.

Still, we can do something. The point of view of these Lessons is to support decision-making. Causation is a valuable concept for decision-making, particularly in cases where the decision-maker is considering an *intervention*. With this as an anchor, a pragmatic definition of "causation" is available:

> Causation describes a class of hypotheses that DAGs can represent. In that representation, a causal relationship between two nodes X and Y is marked by a causal path connecting X to Y. In Lesson **??**, we defined "causal path" in terms of the directions of arrows in a DAG.[1] A definitive demonstration of a causal relationship between X and Y is that intervening to change X results reliably in a change in Y, *all other nodes not on the causal path being held constant.* (Lesson **??** treats the methodology behind this definitive sign.)

Whether or not a definitive demonstration is feasible is not directly relevant to the decision-maker. A decision-maker acts under the guidance of one or more hypotheses. A good rule

[1] We will consider a "direct causal link" to be a form of causal path.

of thumb for decision-makers is to be guided only by plausible hypotheses. Whether a hypothesis is plausible is a matter of informed belief. A definitive demonstration should sharpen that belief. If no such definitive demonstration is available, the decision-maker must rely on alternative sources for belief. Austin Bradford Hill (1898-1991), an epidemiologist and eminent statistician, famously published a list of nine criteria that support belief in a causal hypothesis.

Using my definition of causation, and in marked disagreement with many statisticians, I submit that

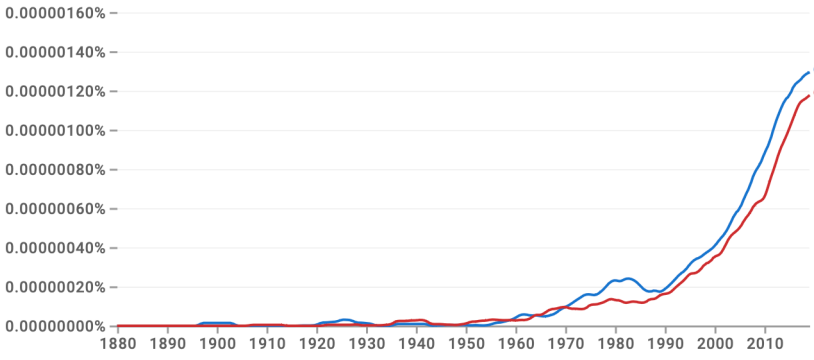
Correlation implies causation.

"Correlation implies causation" is not the same as saying, "A correlation between A and B implies that A causes B." That statement is false. For instance, it might be instead that B causes A. Alternatively, there might be a common cause C for both A and B. Or, C might be a collider between A and B.

There is no mechanism to produce correlation that I am aware of, other than the sources of spurious correlation described previously, that does not involve causation in some way.

::: {.callout-note} ## So why do many statisticians say different?

Historically, the rise of the expression "correlation does not imply causation"—Figure **??** shows the ngram since the 1888 invention of the correlation coefficient—comes *after* the peak in the use of the word "correlation."



31.5: Google NGram showing in the use of the phrases "cor- does not imply causation," rrelation is not causation" in lecades.

The first documented use of the phrase is from 1900. It comes in a review of the second edition of a book, *The Grammar of Science*, by Karl Pearson (whom we have met before in this Lesson).

The Grammar of Science is a metaphysically oriented prescription for a new type of science. It posited that sciences such as physics or chemistry unnecessarily drew on metaphors for causation, such as "force." Instead, the book advocated another framework as more appropriate, eschewing causation in favor of descriptions of "perceptions" with probability.

Pearson illustrates his antipathy toward causation with an example of an ash tree in his garden:

> [T]he causes of its growth might be widened out into a description of the various past stages of the universe. One of the causes of its growth is the existence of my garden, which is conditioned by the existence of the metropolis [London]; another cause is the nature of the soil, gravel approaching the edge of the clay, which again is conditioned by the geological structure and past history of the earth. The causes of any *individual* thing thus widen out into the unmanageable history of the universe. *The Grammar of Science*, 2/e, p. 131

It should not be surprising that the field of statistics, which uses probability very extensively as a description, and that developed correlation as a measure of probability, would advocate for more general use of its approach. In this spirit, I read "correlation does not imply causation" as "our new science framework of probability and correlation replaces the antiquated framework of causation." Outside of statistics, however, probability is merely a tool; causation does indeed have practical use. All the more so for decision-makers.

32 Experiment and random assignment

In its everyday meaning, the word "experiment" is similar in meaning to the word "experience." As a verb, to experiment means to "try out new concepts or ways of doing things." As a noun, an experiment is a "course of action tentatively adopted without being sure of the outcome: the farm is an ongoing experiment in sustainable living." Both quotes are from the Oxford Languages, which provides examples of each: "the designers experimented with new ideas in lighting" or "the farm is an ongoing experiment in sustainable living."

From movies and other experiences, people associate experiments with science. Indeed, one of the dictionary definitions of "experiment" is: "a scientific procedure undertaken to make a discovery, test a hypothesis, or demonstrate a known fact."

Almost all the knowledge needed to perform a scientific experiment relates to the science itself: what reagents to use, how to measure the concentration of a neurotransmitter, how to administer a drug safely, and so on. This is why people who carry out scientific procedures are trained primarily in their area of science.

> **Example: Malaria and bed nets**
>
> In many parts of the world, malaria is a major cause of disability and death. Economists who study ways to relieve poverty have a simple, plausible theory: reducing the effect of illnesses such as malaria will have an impact on poverty rates, since healthier people are more productive and reduced uncertainty can help them amass capital to invest to increase production further.
> There are many possible ways to reduce the burden of

malaria. Vaccination (although effective vaccines have been hard to develop), insect control using pesticides (which can cause environmental problems), etc. One simple intervention is the use of bed nets; screen nets deployed at night by draping over the bed and its occupant. Still, there are reasons why distributing bed nets may not be effective; people might use them incorrectly or for other purposes such as fishing. People might not be able to afford them, but giving them away might signal that they have no value.
To find out, try it: do an experiment. For instance, run a trial program where nets are given away to everyone in an area and observed whether and to what extent rates of malarial illness go down.
Such a trial is certainly an experiment. But it may not be the best way to get meaningful information.

32.1 Replication

To understand some of the contribution that statistical thinking can make to experiment, recall our earlier definition:

> *Statistic thinking is the explanation/description of variation* in the context of *what remains unexplained/undescribed.*

A key concept that statistical thinking brings to experiment is the idea of **variation**. Simply put, a good experiment should involve some variation. The simplest way to create variation is to repeat each experimental trial multiple times. This is called "**replication**."

32.2 Example: Replicated bed net trials

One way to improve the simple experiment bed net described above is to carry out many trials. One reason is that the results