Table of contents

1

General introduction

In 2009, a poster at the Annual Meeting of the Human Brain Mapping conference caused a stir. It showed significant brain activity in response to the subject being shown photographs of humans in emotional valence. The subject was an Arctic salmon—and it was dead (Bennett et al. 2009). In the same year, Kriegeskorte et al. (2009) highlighted a disturbing amount of circular reasoning and double-dipping in neuroscience, particularly in fMRI analyses, where the test statistics were not independent of the selection criteria and common analyses produced spurious results. Later research comparing how statistical software designed and used by different manufacturers of fMRI machines handled spatial autocorrelation in the voxels revealed an error rate of up to 70% (Eklund et al. 2016). Yet even when done correctly, effects can be tiny and to reliably detect them requires sample sizes much larger than usually published (Marek et al. 2022). Clearly, fMRI-based science needed improvement. But are the statistical approaches and analyses used in fMRI any different from those used in other scientific fields every day?

All statistical methods make some assumptions about the independence, distributions, representativeness, etc. of the samples and the data collected from them. Most classical statistical methods have been mathematically (analytically) proven to yield "asymptotically unbiased estimates." This phrase means that a statistical estimate will converge to its "true" value (that's the "unbiased" bit) given an infinite number of observations (that's the "asymptotic" bit) of randomly-collected samples (observations or data points) that conform to all the assumptions made in the statistical model. However, no one collects an infinite amount of data and residual assumptions are never 100% met, so it is important to ask how reliable our estimates are for finite or even rather small datasets. As a corollary, we also want to know how small is too small (i.e., how many observations are too few for us to compute unbiased, or at least reliably informative, estimates), how large is large enough, and whether our conclusions are robust to minor violations of model assumptions.

Furthermore, statistical methods generally have been developed as stand-alone tools. For example, general linear models (GLMs, of which the familiar analysis of variance [ANOVA] is a special case) are used to identify the expected (or average) response of one variable as a function of one or more predictor variables. Principal component analysis (PCA) is used to reduce the dimensionality of a large and often unwieldy set of variables (either predictors or responses) to a more manageable, smaller number (usually two or three) of composite variables that are linear combinations of the original data. Used by itself, PCA works well as a descriptive, hypothesis-generating tool, whereas a GLM works well as a predictive, hypothesis-testing tool. But what happens if we combine a PCA and a GLM in workflow? Will the PCA-GLM approach still "work?" And what about more complicated methods and workflows using machine learning or neural networks? Will the resulting estimates still be unbiased or will uncertainty and errors propagate throughout the workflow and create statistical artefacts?

Questions such as these rarely can be answered for even a small number of statistical methods, hardly ever by non-mathematicians, and may even defy analytical solutions. Instead, mathematicians and statisticians rely on simulations to answer them.

This book aims at providing an entry to understanding statistics using simulations. By going through many examples, both conceptually and with code, we hope to lower the bar so that scientists who use statistics can simulate data and associated analyses themselves. Although simulations are a standard tool in statistical research and the development of new statistical methods, we do not present or test general methods. Rather, we illustrate the value of simulations by working through key issues and exemplar datasets in specific contexts representing a broad range of research areas—from sociology and archaeology to ecology and physics, and many in between.

What you will learn from this chapter

- What simulated data are, why they are important, and important ways in which they are used.
- How the book is organized and what you need to know to use this book effectively.
- How to read, interpret, and use the Code Boxes.
- What notational conventions are used for mathematical expressions throughout the book.

1.1 What are simulated data?

Simulated data are data similar to what we plan to collect or have already collected, but are created using computational algorithms.[1] We think it is helpful to think of such data as emerging from a "data-generating model" (DGM). The central challenge addressed in this book is to define such a data-generating model and then to use it effectively. When simulating data, we usually take advantage of random-number generators to include random variation in the simulated dataset that is similar to the random variation that occurs in real data. Examples of such random variation include differences among sampled individuals (i.e., between-subject variance), random positions, and residual errors, among many others.

Computers can simulate data efficiently and rapidly. When we run many simulations (e.g., thousands of them, each starting with a different random "seed"), we end up with replicate datasets for which we know the true underlying parameter values, the sampling design, underlying correlations, etc. When we analyse these simulated data, we can quickly check for and identify biases in our estimates and determine whether our methods are reliable and robust. We also can use simulations to test model fits and generate p-values. Pedagogically, it can be argued that if we cannot simulate our data, we probably do not know what we are doing when we are analysing them. Even if we think we know what we are doing, simulating data opens our eyes to what is simple and what is difficult in statistical analysis.

[1] Other terms you may encounter for simulated data include "synthetic data," "fake data," or "invented data." Fake data and invented data have negative connotations and suggest that we are "cooking the books" by doing something wrong or illegitimate. "Synthetic data" is fine, but in the statistical sciences, "simulated" is used more frequently than "synthetic."

Simulated data can take many forms, from a small number of observations of a single variable to a very large number of observations generated by a sophisticated model of a complex system driven by different underlying stochastic processes and sampled with a different stochastic sampling process (e.g., Zurell et al. 2010). Think, for example, about modelling traffic jams or representing the associated emergency calls as a way of collecting data. Both processes can be simulated at the level of the data (which is the focus of this book) or at the deeper level of the process(es) that could plausibly generate the data. The latter approach is touched upon in Chapter 12.

One more thing before we start. Simulations and reality have a lot in common.[2] In both, strong effects will shine through but weak effects will be hard to discern through the fog of high variability (Silver 2012).

1.2 Simulated data are specific

Simulation sacrifices generality for ease of use. That is, we typically use simulations to ask questions about a specific dataset or a very specific type of statistical analysis. Unlike an analytical proof, the results of a simulation study rarely are applicable to a wide range of statistical methods, even those that share underlying mathematical structures. Despite its lack of generality, however, simulation is a great tool to use to explore how a particular kind of statistical method works and what it can actually tell us about the kinds of data we can collect or already have on hand.

Any real dataset is but a single instance (and, we hope, a random one) of all the possible datasets that could have been collected from the population of interest. Although we might have sampled more or fewer subjects or sites, used a different method, or collected the data at a different time, we did not. We are usually interested in analysing such specific datasets and using the analyses to make more general inferences about all the datasets we did not collect, i.e., the population.

When we simulate data, we have such a specific dataset in mind, but we do not restrict ourselves to the exact values in our real dataset. Rather, we are asking how much quantitative information can be gleaned from this specific instance of a more general kind or type of dataset.[3] In other words, is the information in the data sufficient for us to extract a signal from the noise? To answer this question using data simulation, we repeatedly generate data with a similar underlying structure and apply our intended analysis to them. Since we defined and set the parameters in our simulations, we can now check: How often do we find the effect we programmed into the simulation? Are our parameter estimates biased? If we are interested in testing our model, we can also ask whether the p-value of the output correctly reflects the number of significant effects across the simulations.

If a statistical method applied to our simulated data yields unbiased estimates or the expected distribution of p-values, then we can be more confident that the application of the method to the analysis of our single and unique real dataset will be similarly unbiased and the p-value will accurately reflect the statistical "significance" of our results. But remember the dead salmon. If the application of our method to our simulated data yields, for example,

[2] In fact, many science-fiction writers and even some dyed-in-the-wool philosophers have asserted that reality may be nothing more than a simulation (e.g., Adams 1979; Bostrom 2003).

[3] We distinguish here between types and instances of a variable or a dataset. A *type* is a more general category, whereas an *instance* is a specific example of the type. For example, the value 10 is an instance of a variable whose type is "integer." An interesting feature of types is that they are hierarchical, so that objects that are usually thought of as types also can be used as instances of other objects.

biased estimates, numerical anomalies, or incorrect standard errors, then we should not trust a similar analysis of the real dataset.

1.3 Yes, scientists really simulate data

Although some mathematicians still frown upon simulations, there are many interesting problems for which closed-form analytical solutions have not yet been identified. Using computers to simulate data and explore a range of possible solutions of analytically insoluble problems now is recognised as a legitimate approach to solving them; one of the first and perhaps most familiar examples is the solution to the four-colour map problem (Appel & Haken 1989; see Meyer & Schmidt 2012 for others). For many statistical methods, simulations are the only way available to validate established and proposed methods.

Not only do many scientists simulate data and analyse them,[4] statisticians and many others teaching both theoretical and applied statistics actively encourage this practice (e.g., Crawley 2007; Jones et al. 2014; Morris et al. 2019; McElreath 2020; Crump et al. 2021; Donovan et al. 2021). Really, there is no reason not to, and we illustrate the power of simulation with a few straightforward examples in the following subsections and in the next chapter. These motivate the more complex problems described in the rest of the book and should whet your appetite for doing simulations yourself.

1.3.1 Geographic biases in reported crime rates

Governments collect and analyse data on crimes to identify where they are committed and the kinds of individuals most likely to commit them. Systematic reviews of these data have suggested that most crimes reported to police occur in a small number of neighbourhoods (Lee et al. 2017). This result has been used to develop strategies that, their proponents assert, will most efficiently deploy people and resources to those areas where crimes are most likely to occur. Critics point to the racial, socioeconomic, and other biases on which these spatially-targeted strategies are based and that they perpetuate. Simulations have been used extensively to support both sides of the debate (Liu & Eck 2008). An important, but rarely asked question, however, is whether the data used to build the crime simulations are themselves biased.

Buil-Gil et al. (2022) used simulation to address this question. They examined potential sources of bias in reported crime statistics in the UK's Crime Survey for England and Wales (CSEW). These researchers simulated three things for the city of Manchester, England: a population of people, the number of crimes experienced by each individual in the population, and the number of those crimes that were actually reported to the police (Figure 1.1). Buil-Gil et al. (2022) then asked whether the bias in the number of reported crimes changed from the smallest spatial scale (a neighbourhood of $\approx$125 households) to larger ones (communities [collections of neighbourhoods], precincts [collections of communities], and wards [collections of precincts]).

The results were striking. First, the simulations accurately reflected the CSEW data that nearly two-thirds (62%) of all committed crimes are not reported to the police. But the variation in non-reporting was largest among neighbourhoods (inset plot in Figure 1.1). That is, different neighbourhoods, each of which tends to be demographically more homogeneous than the larger communities, precincts, or wards, report crimes at different rates. These

[4] Any list of fewer than thousands of references is unfair to those doing simulations, but here are a few more recent examples: Royston & Sauerbrei (2014); Boulesteix et al. (2018); Jayasekera et al. (2018); Boulesteix et al. (2020); Adams & Collyer (2022); Martínez-Santalla et al. (2022); DiRenzo et al. (2023).

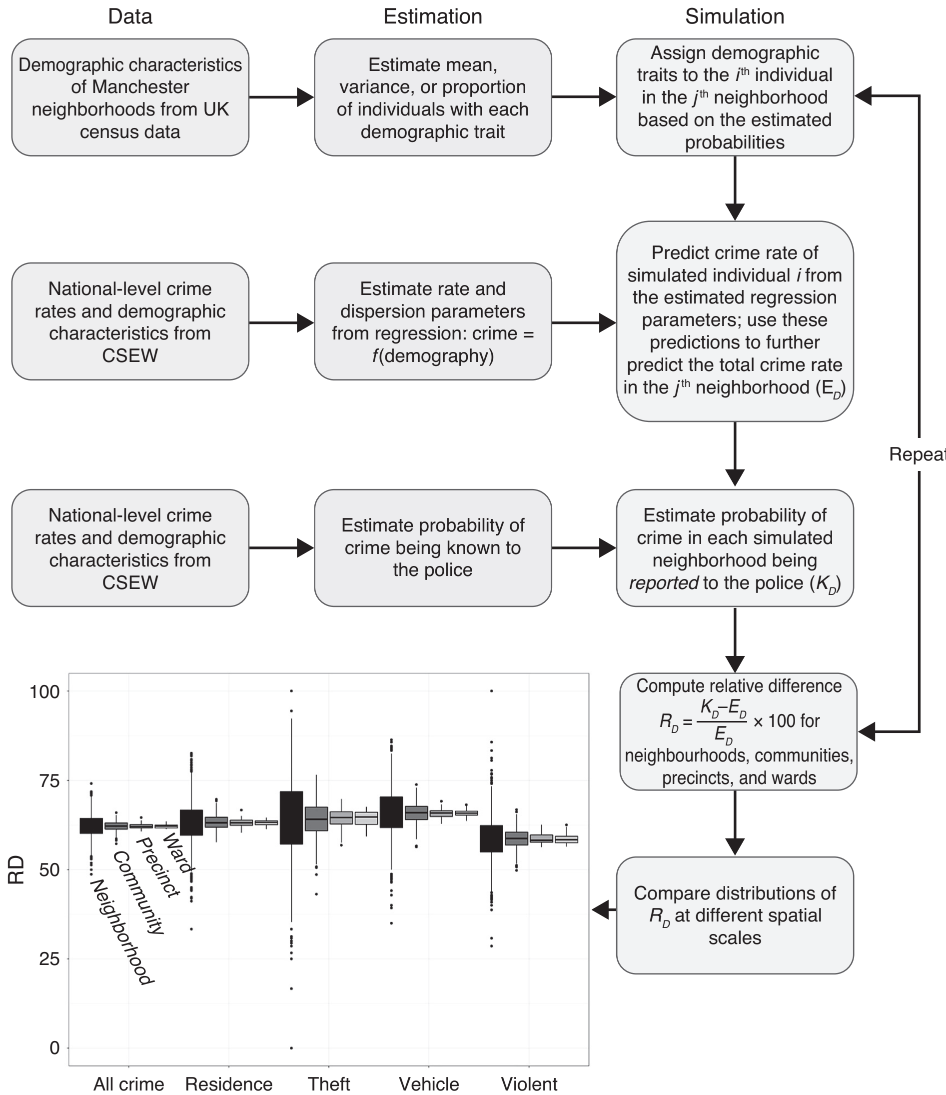

Figure 1.1 Workflow and results of the simulation exploring geographic bias in reported crime rates. The workflow, derived from the description of the simulation in Buil-Gil et al. (2022), includes obtaining data from government sources, estimation using negative binomial and logistic regressions, simulations, and visualisations. The inset figure is reproduced from Figure 1 of Buil-Gil et al. (2022) (CC-BY-4.0).

differences in reporting rates can lead to biases in the data that are used in criminological research and caution against blind acceptance of crime reduction strategies targeted at individual communities ("micro-geographies").

1.3.2 Handling data gaps

FLUXNET[5] is an international network of hundreds of stations that use the eddy-covariance technique to measure the flux of CO_2 and water above vegetation in a standardised way (Baldocchi 2003). The apparatus is very sensitive but requires air movement for measurements;

[5] https://fluxnet.org.

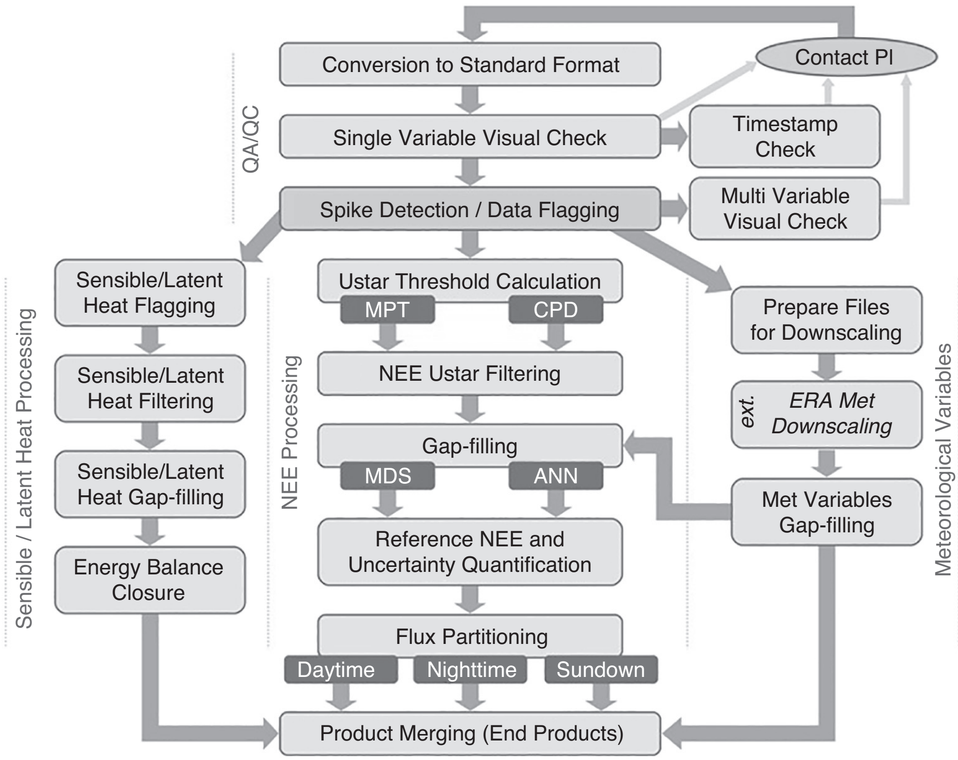

Figure 1.2 Workflow for gap filling of eddy covariance data in the FLUXNET network. Image in the public domain from FLUXNET (nd).

CO_2 flux cannot be measured at very low wind speeds, which happen regularly and especially at night and result in gaps in the data streams. Because these data gaps occur non-randomly, they need to be "filled" before the dataset can be analysed to avoid biased results.

Over the years, and after many, many simulations and comparisons of gap-filling methods, a sophisticated data pre-processing pipeline has been established (Figure 1.2). This is now the default workflow applied to all raw data entering the FLUXNET repository and data-processing chain. However, this workflow and the resulting estimates of the flux of carbon between forests and the atmosphere depends strongly on how the "U-star" ("Ustar" in Figure 1.2) threshold is determined. Simulations have shown that small changes in U-star could change the conclusion that a forest takes up more carbon from the atmosphere than it releases to its opposite—that the forest actually contributes planet-warming CO_2 into the atmosphere (Ellison et al. 2006).

1.3.3 Data augmentation in convolutional neural networks

Automatic classification of images is used widely—and often inappropriately or to ill effect—by many people, organisations, and government agencies. Classification, also called the "tyranny of the discontinuous mind" by Richard Dawkins (2011), can be improved by providing more and more data to the classifying algorithm. In lieu of genuine new data, computer scientists training convolutional neural networks (CNNs) on images use data augmentation: simulation of new data points based on old ones. Typical methods of data augmentation are flipping or rotating images, re-sizing or re-colouring them, adding noise to them, or increasing their grain. More recently, generative adversarial networks (GANs) have introduced more sophisticated image mimicry but rely similarly on the variability in the training data. If each training image is altered in all combinations of such manipulations, sample size can be increased ten-fold with only marginal computational cost and at no cost to data collectors (e.g., Gopal et al. 2020; Nandhini Abirami et al. 2021; Li et al. 2023).

The second step of these simulations is to see whether data augmentation using GANs actually improves the classifier. Classifying the same images is uninformative; only classification of new, previously analysed, and validated images can tell us whether the method of data augmentation actually works. Indeed, it does improve classification accuracy for new images (e.g., Perez & Wang 2017; Mikołajczyk & Grochowski 2018), but not for the reasons initially suggested. Rather, the main reason seems to be that data augmentation using GANs adds noise, making the classifier avoid focusing on the specifics of an image, such as the stark contrast of the light reflection on the wet nose of a dog. As a result, data augmentation is now seen as a regularising or shrinkage approach rather than as a way to simulate new data (Steiner et al. 2022).

1.3.4 Null models in network analysis

When the value of an index depends in a complicated and non-linear way on its inputs, its distribution can be difficult to derive mathematically. One relevant case is the structure of bipartite (two-part) networks, which show the relationships between, for example, actors and the movies in which they appear, directors and the boards on which they serve, or pollinators and the flowers they visit. A description of what network structure we could expect based on how many movies and popular actors there are (i.e., in how many movies an actor has appeared) requires simulating the networks.

For example, when trying to identify modules ("cliques") of actors (or pollinators), we use a "null model": an algorithm to simulate a reference baseline. For either Boolean networks (e.g., an actor is in a movie or not) or networks with variable (but integer) number of interactions (e.g., a pollinator visits a flower N times), a modularity algorithm uses the sum of observations for each combination of actors and movies or pollinators and flowers, respectively, to simulate thousands of random networks against which the observed network is compared to determine if it is "structured" or random (e.g., Newman & Girvan 2004; Newman 2006; Barber 2007).

1.4 There are many good reasons to simulate data

Data simulation often is motivated by a specific problem with real data, such as violation of assumptions (e.g., normality, homoscedasticity), missing data, or unbalanced designs. However, there are many other benefits to simulating data besides demonstrating the validity or soundness of a particular method applied to a specific dataset. Perhaps most importantly, data simulation helps us to work through new problems that turn up while we are planning our research or analysing our data, and assess the validity of our proposed solutions when we are unsure of them.

While we were writing this book, ChatGPT, CoPilot, Llama, and many other large language models (LLMs) masquerading as artificial intelligence (AI) came online (cf. Searle 1982). Although many pundits (and not a few of our students and colleagues) have asserted that Big Data spelled the end of data analysis (e.g., Anderson 2008) and that AI and LLMs will make it unnecessary to learn how to write code (Hutson 2022), statisticians and programmers are still in high demand. Why? Even the best LLMs "hallucinate": although they may confidently provide an answer to a question (a "prompt"), the answer may be inaccurate or completely wrong (e.g., Lehr et al. 2024). Hallucinations are most common when the data used to train the LLM had errors or did not include enough information to address a particular prompt, when the prompt does not have sufficient context, or if the prompt concerns methods, packages, or programming languages that are not commonly used. Thus, it is incumbent on us to be sceptical of responses given by any AI or LLM. Answers should always be checked for accuracy and code should be tested carefully to determine whether it works the way we expect (e.g., Cooper et al., 2024). Independent data simulations are one of many useful tools to check the responses of an LLM and to use it effectively (see also Lubiana et al. 2023; van Dis et al. 2023).

In this book, our main focus is on using simulations to better understand what statistical models and analyses do, and in doing so, to improve our statistical competence. However, simulations increasingly are being used as part of scientific publications. In recent years, scientific societies and journal publishers have begun to develop guidelines for reporting the results of simulations so that readers can trust and maximally benefit from them (Morris et al. 2019; DiRenzo et al. 2023).

1.5 Useful background knowledge to use this book most effectively

This book is for anybody doing statistical analysis who already is familiar with basic concepts such as probability and likelihood and has skills with standard hypothesis-testing and model-fitting methods such as t-tests, χ^2-statistics, and GLMs. Our previous books (Gotelli & Ellison 2012; Dormann 2020) can be used as refreshers. If you are starting from scratch, the `OpenIntro` project[6] has excellent open-source resources for teaching and learning basic statistics. Its latest book, the second edition of *Introduction to Modern Statistics* (Çetinkaya-Rundel & Hardin 2024), covers basic concepts in modelling, hypothesis testing, and statistical inference; includes detailed exercises and scripts in the `R` programming language, and introduces simulation and randomisation tests with its discussion of the foundations of statistical inference. Fieberg (2024) focuses more tightly on alternatives to ordinary linear regression for analysing observational data collected by ecologists. His text includes many good examples of using data simulation and, like us, takes a pragmatic approach to the utility of both frequentist and Bayesian approaches.

We also assume familiarity with at least one statistical programming language, such as `R`, `Python`, `Julia`, `Matlab`, or `Mathematica`, and the ability to construct scripts in it from descriptive algorithms and pseudocode. We provide code for all our examples written in the `R` programming language (R Core Team 2023) and presented in a standard form for all the algorithms and examples in the text (Code Box 1.1).[7]

All the code may be downloaded from the online code repository associated with this book (Dormann & Ellison 2024). With only a few exceptions, we eschew the `tidyverse` and pipes (Wickham et al. 2019); all data wrangling and functions are written in "standard" `R` and will run in either the classic `R` GUI (Venables et al. 2023) or `RStudio` (RStudio Team 2020). Similarly, with the exception of the graphics in the Appendix, all plots are generated using `ggplot` (Wickham 2016). Unless the primary goal of a particular code snippet is generating a specific plot, we do not include the plotting commands in the Code Boxes. However, the code to generate all the plots is included in the online code repository. Finally, the online repository also includes additional examples and associated code that we have developed since writing the book, and we encourage our readers and others interested in data simulation to contribute their own examples and code to the repository.

[6] https://openintro.org

[7] There are packages in `R` or `Python` that simulate data for you (e.g., `conjurer` and `simstudy` in `R`; see Appendix A for others), but which hide what actually is going on. So, we use them sparingly.

Code Box 1.1: How to read the code in the Code Boxes

```
# Code is presented in the same way throughout the text

# Lines or phrases beginning with a "hash tag" or "pound sign" (#) are comments
    that describe the subsequent or associated executable commands. If these
    lines are pasted into an R script, they will not run. For example, the line

# Load required libraries

# tells you that the subsequent lines will include the contributed packages
    (R "libraries") necessary for the executable code to run successfully.

# Lines indented with one or more spaces and lacking a leading character (such
    as a # or a |) are executable code. If these are pasted into an R script,
    they will run. For example, the line

 mean(x)

# will return the average of the variable "x".

# Lines beginning with a vertical bar (|) include the output (returned to the
    console) if executable code is executed. These lines should not be pasted
    into an R script, as they are neither comments nor executable code. For
    example,

| [1] 13.75

# would be the output from running the previous command (mean(x)).
```

1.6 Notational conventions

We have endeavoured to use consistent notation for mathematical expressions throughout the book, following Casella & Berger (2002). An upper-case letter (e.g., X) indicates a random variable, whereas its corresponding lower-case letter (e.g., x,) is a specific value from it. The set of all N values of $\{X_1, X_2, \ldots, X_N\}$ is denoted as X or $\{X\}$. We capitalise the sample size, N, throughout the main text, but in Code Boxes, it may appear in either lower case or upper case, depending on how it is defined in existing R functions.[8] We do not use vector notation (e.g., $\mathbf{X}$) for a matrix X; the difference between scalars and vectors should be clear from context.

Population-level ("true" and usually unknown) parameters are written with Greek letters (e.g., the population mean μ and its variance σ^2), whereas sample parameters normally are written with Roman letters (e.g., the sample mean, $\bar{x}$, and its variance, s^2). Modelled estimates are indicated with "hats" (e.g., the estimated population variance is written as $\hat{\sigma}^2$ and the estimated sample variance is written as $\hat{s}^2$).

[8] Note that commands, object names, etc. in R are case-sensitive: "Bird" and "bird" would be two different kinds of flying animals.

The names of libraries, functions, and variables used in `R` (or other programming languages) are set in `monospaced font`; function names always include parentheses, as in `t.test()`, to serve as a reminder that they need to have objects, parameters, or options provided to them (e.g., `t.test(x, y, type="paired")`).

1.7 Structure, organisation, and flow

The book is constructed in such a way that the two chapters in this Part I set the tone. After that, we proceed through topics following the same path that we would normally use to develop a research project: ask a question and develop hypotheses; come up with a sampling protocol or experimental design; collect, visualise, summarise, and analyse the data; fit and test models and (re)check their assumptions; and state our conclusions together with our degree of certainty about them. We show that using data simulation helps us do all these steps better, improves our science, and give us more confidence in the conclusions we draw from our research.

In the two chapters in Part II of this book, we show how to use simulations to plan surveys, observations and experiments from the time we come up with an idea until we actually start the study. For example, we can investigate how a proposed sample size will affect the probability of detecting weak effects. Simulation might also suggest that we abandon a proposed study if we cannot muster the necessary resources and logistics to reliably test our hypotheses. Thinking longer about a study *before* we do it is always beneficial. Data simulation encourages us to think first and act later, and forces us to be explicit about our assumptions and expectations.

The three chapters in Part III get us ready to analyse our hard-earned data. Before we analyse the data we have collected, we should check to see if our data satisfy key distributional assumptions (Chapter 5). We also can use simulations to check whether the method or model we plan to use to analyse our data makes sense or is it simply folklore: a rule-of-thumb that "everybody knows" or has "always been done" (Chapter 6). In a similar vein, individual models may work fine by themselves but exhibit strange behaviour when linked together in a workflow or analytical pipeline. Data simulations can help us discover that before we get so far into a tunnel that we cannot turn around and extricate ourselves (Chapter 7).

The two chapters in Part IV showcase ways to use data simulations to evaluate the fit of our models and what to do if they exhibit strange behaviour or yield results that are not statistically significant. If our models work as expected, all is well; simulations let us evaluate the fit of models we know to be reliable (Chapter 8). If they do not and our model yields really strange or unexpected results when we analyse our real data, we also can use simulations to probe the method or model more deeply, identify the origin of the weird results, and have firmer grounds from which to judge whether the results are strange but true or only an odd artefact of the specific sample or method (Chapter 8). Finally, if—despite all our careful planning, scrupulous data collection, and robust analysis—our results are not statistically significant, simulation shows us that we should not seek absolution in retrospective power analysis (Chapter 9). At the same time, we can use retrospective design analysis (Gelman & Carlin 2014) to test whether our statistically significant results are as meaningful as we think they are (Chapter 9).

The three chapters in Part V close the circle. We can use all the approaches presented in Parts II–IV to synthesise existing work using meta-analysis or federated analyses (Chapter 10), or to design and test new statistical methods or create new indices (Chapter 11). Using simulations offers an efficient and effective way to demonstrate the appropriateness of a new procedure or detect its flaws. Finally, Chapter 12 returns to where we started this

Table 1.1: Where to find detailed discussion of key statistical topics that recur throughout this book.

Topic	Chapter.Section
Assumptions:	
– Normally-distributed data	2.3; 5.1; 5.6; 8.4
– Homoscedasticity	2.2; 5.4; 8.4
– Independence	2.4; 4.2; 5.5–5.6; 6.5–6.6
– Stationarity	5.7; 12.2
Bayesian inference:	
– Bayes' Theorem	3.3; 6.4
– Prior vs. posterior distributions	8.1; 8.6
– MCMC and its relatives	7.7; 8.1; 11.6
Categorical vs. continuous predictors	6.3; 8.5
Collinearity	6.2; 7.5; 7.8
Confidence intervals	2.3; 3.4; 8.5; 9.3
Effect size	3.4; 4.1; 7.6; 9.1–9.3; 10.3–10.7
Imputation of missing data	6.1; 7.3–7.4; 7.8
Machine learning vs. classical statistics	6.2; 6.6; 7.5; 8.4–8.5; 11.4–11.5
Meta-analysis	3.4; 10.3–10.4
Model selection	6.1; 7.8; 10.2; 11.3
Power analysis	
– a priori (prospective)	3.4; 4 [entire chapter]
– post hoc (retrospective)	9 [entire chapter]; 10.3
Time-series analysis	2.4; 4.4; 6.4–6.5; 11.5

book: using simulations for planning large and very complex research programs that are addressing some of the most difficult questions scientists have posed to date.

We encourage you to read and work through the chapters in order. However, each chapter is mostly self-contained and you should be able to jump to, and work with, any part or chapter of the book in which you are most interested. Some key statistical topics and concepts pertain to more than one method or model, or recur in different contexts. Table 1.1 provides an overview of the most important of these topics and points to where we present them in detail. For more fine-grained direction, please consult the table of contents and the index.

1.8 Summary

- Simulated data are created using computational algorithms.
- Simulated data are similar, but not identical, to data we have already collected or that we plan to collect.
- Simulating data requires a "data-generating model" that normally implies a specific question, dataset, or type of statistical analysis.
- One of the most important uses of data simulation is to determine how much information about a large population can be gleaned from the smaller sample of it that has been collected.

- Other common uses of data simulation include testing assumptions, working through unanticipated or unexpected problems that came up while planning a research project or analysing the data, assessing the validity of proposed solutions, and determining whether the results of using large language models are real or hallucinatory.
- To use this book most effectively, you should already be familiar with basic statistical concepts such as probability and likelihood, and have skills with standard hypothesis-testing and model-fitting methods such as t-tests, χ^2-statistics, and GLMs.

References

Adams, D. (1979). *The Hitchhiker's Guide to the Galaxy*. London, UK: Pan/Macmillan.

Adams, D. C. & Collyer, M. L. (2022). Consilience of methods for phylogenetic analysis of variance. *Evolution*, 76(7), 1406–1419. doi: 10.1111/evo.14512.

Anderson, C. (2008). The end of theory: The data deluge makes the scientific method obsolete. *WIRED*, 16(7). https://www.wired.com/2008/06/pb-theory/.

Appel, K. & Haken, W. (1989). *Every Planar Map Is Four Colorable*. Providence, RI, USA: American Mathematical Society.

Baldocchi, D. D. (2003). Assessing the eddy covariance technique for evaluating carbon dioxide exchange rates of ecosystems: Past, present and future. *Global Change Biology*, 9(4), 479–492. doi: 10.1046/j.1365-2486.2003.00629.x.

Barber, M. (2007). Modularity and community detection in bipartite networks. *Physical Review E*, 76(6), 1–9. doi: 10.1103/PhysRevE.76.066102.

Bennett, C. M., Miller, M. B., & Wolford, G. L. (2009). Neural correlates of interspecies perspective taking in the post-mortem Atlantic Salmon: An argument for multiple comparisons correction. *NeuroImage*, 47, S125. doi: 10.1016/S1053-8119(09)71202-9.

Bostrom, N. (2003). Are you living in a computer simulation? *Philosophical Quarterly*, 53(211), 243–255.

Boulesteix, A.-L., Binder, H., Abrahamowicz, M., et al. (2018). On the necessity and design of studies comparing statistical methods. *Biometrical Journal*, 60(1), 216–218. doi: 10.1002/bimj.201700129.

Boulesteix, A.-L., Groenwold, R. H. H., Abrahamowicz, M., et al. (2020). Introduction to statistical simulations in health research. *British Medical Journal Open*, 10(12), e039921. doi: 10.1136/bmjopen-2020-039921.

Buil-Gil, D., Moretti, A., & Langton, S. H. (2022). The accuracy of crime statistics: Assessing the impact of police data bias on geographic crime analysis. *Journal of Experimental Criminology*, 18, 515–541. doi: 10.1007/s11292-021-09457-y.

Casella, G. & Berger, R. L. (2002). *Statistical Inference*. 2nd edition. Duxbury Press/Thomson Learning.

Çetinkaya-Rundel, M. & Hardin, J. (2024). *Introduction to Modern Statistics*. https://openintro.org: OpenIntro, 2nd edition.

Cooper, N., Clark, A. T., Lecompte, N., et al. (2024). Harnessing large language models for coding, teaching and inclusion to empower research in ecology and evolution. *Methods in Ecology and Evolution*, 15(10), 1757–1763. doi: 10.1111/2041-210X.14325.

Crawley, M. J. (2007). *The R Book*. Chichester, West Sussex, UK: John Wiley & Sons.

Crump, M. J. C., Navarro, D., & Suzuki, J. (2021). Answering questions with data: Introductory statistics for psychology students. DOI: 10.17605/OSF.IO/JZE52. Accessed 1 February 2024.

Dawkins, R. (2011). The tyranny of the discontinuous mind. *New Statesman*, 100(5084/5085), 54–57.

DiRenzo, G. V., Hanks, E., & Miller, D. A. W. (2023). A practical guide to understanding and validating complex models using data simulations. *Methods in Ecology and Evolution*, 14(1), 203–217. doi: 10.1111/2041-210X.14030.

Donovan, T. M., Brown, M., & Katz, J. E. (2021). R for fledglings. https://www.uvm.edu/˜tdonovan/RforFledglings. Accessed 1 February 2024.

Dormann, C. & Ellison, A. M. (2024). Data ex machina. https://osf.io/3skv4/.

Dormann, C. F. (2020). *Environmental Data Analysis: An Introduction with Examples in R*. Heidelberg, Germany: Springer.

Eklund, A., Nichols, T. E., & Knutsson, H. (2016). Cluster failure: Why fMRI inferences for spatial extent have inflated false-positive rates. *Proceedings of the National Academy of Sciences, USA*, 113, 7900–7905. doi: 10.1073/pnas.1612033113.

Ellison, A. M., Osterweil, L. J., Hadley, J. L., et al. (2006). Analytic webs support the synthesis of ecological datasets. *Ecology*, 87(6), 1345–1358. doi: 10.1890/0012-9658(2006)87[1345:AWSTSO]2.0.CO;2.

Fieberg, J. R. (2024). *Statistics for Ecologists: A Frequentist and Bayesian Treatment of Modern Regression Models*. Minneapolis, MN, USA: University of Minnesota Libraries Publishing.

FLUXNET (n.d.). Data Processing 101: Pipeline and Procedures. https://fluxnet.org/data/aboutdata/data-processing-101-pipeline-and-procedures/. Accessed 16 January 2024.

Gelman, A. & Carlin, J. (2014). Beyond power calculations: Assessing type S (sign) and type M (magnitude) errors. *Perspectives on Psychological Science*, 9(6), 641–651. doi: 10.1177/1745691614551642.

Gopal, A., Gandhimaruthian, L., & Ali, J. (2020). Role of general adversarial networks in mammogram analysis: A review. *Current Medical Imaging*, 16(7), 863–877. doi: 10.2174/1573405614666191115102318.

Gotelli, N. J. & Ellison, A. M. (2012). *A Primer of Ecological Statistics*, 2nd edition. New York, NY, USA: Oxford University Press.

Hutson, M. (2022). AI learns to write computer code in 'stunning' advance. DOI: 10.1126/science.adg2088. Accessed 26 May 2024.

Jayasekera, J., Li, Y., Schechter, C. B., et al. (2018). Simulation modeling of cancer clinical trials: Application to omitting radiotherapy in low-risk breast cancer. *JNCI: Journal of the National Cancer Institute*, 110(12), 1360–1369. doi: 10.1093/jnci/djy059.

Jones, O., Maillardet, R., & Robinson, A. (2014). *Introduction to Scientific Programming and Simulation Using R*, 2nd edition. Boca Raton, FL, USA: Chapman & Hall/CRC.

Kriegeskorte, N., Simmons, W. K., Bellgowan, P. S. F., & Baker, C. I. (2009). Circular analysis in systems neuroscience: The dangers of double dipping. *Nature Neuroscience*, 12(55), 535–540. doi: 10.1038/nn.2303.

Lee, Y., Eck, J. E., O, S., & Martinez, N. N. (2017). How concentrated is crime at places? A systematic review from 1970 to 2015. *Crime Science*, 6, 6. doi: 10.1186/s40163-017-0069-x.

Lehr, S. A., Caliskan, A., Liyanage, S., & Banaji, M. R. (2024). ChatGPT as research scientist: probing GPT's capabilities as a research librarian, research ethicist, data generator, and data predictor. *Proceedings of the National Academy of Sciences USA*, 121(35), e2404328121.

Li, J., Yu, Z., Yu, L., et al. (2023). A comprehensive survey on SAR ATR in deep-learning era. *Remote Sensing*, 15(5), 1454. doi: 10.3390/rs15051454.

Liu, L. & Eck, J., Eds. (2008). *Artificial Crime Analysis Systems: Using Computer Simulations and Geographic Information Systems*. Hershey, PA, USA: Information Science Reference.

Lubiana, T., Lopes, R., Medeiros, P., et al. (2023). Ten quick tips for harnessing the power of ChatGPT in computational biology. *PLoS Computational Biology*, 19(8), 1–9. doi: 10.1371/journal.pcbi.1011319.

Marek, S., Tervo-Clemmens, B., Calabro, F. J., et al. (2022). Reproducible brain-wide association studies require thousands of individuals. *Nature*, 603(7902), 654–660. doi: 10.1038/s41586-022-04492-9.

Martínez-Santalla, S., Martín-Devasa, R., Gómez-Rodríguez, C., et al. (2022). Assessing the nonlinear decay of community similarity: Permutation and site-block resampling significance tests. *Journal of Biogeography*, 49(5), 968–978. doi: 10.1111/jbi.14351.

McElreath, R. (2020). *Statistical Rethinking: A Bayesian Course with Examples in R and STAN*, 2nd edition. Boca Raton, FL, USA: Chapman & Hall/CRC.

Meyer, K. R. & Schmidt, D. S., Eds. (2012). *Computer Aided Proofs in Analysis*. New York, NY, USA: Springer.

Mikołajczyk, A. & Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. In *2018 International Interdisciplinary PhD Workshop (IIPhDW)* (pp. 117–122). Świnouście, Poland.

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

Nandhini Abirami, R., Durai Raj Vincent, P. M., Srinivasan, K., et al. (2021). Deep CNN and deep GAN in computational visual perception-driven image analysis. *Complexity*, 2021, 5541134. doi: 10.1155/2021/5541134.

Newman, M. E. J. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences, USA*, 103(23), 8577–8582. doi: 10.1073/pnas.0601602103.

Newman, M. E. J. & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E*, 69(2), 1–15. doi: 10.1103/PhysRevE.69.026113.

Perez, L. & Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. *arXiv*, 1712.04621. doi: 10.48550/arXiv.1712.04621.

R Core Team (2023). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Royston, P. & Sauerbrei, W. (2014). Interaction of treatment with a continuous variable: Simulation study of power for several methods of analysis. *Statistics in Medicine*, 33(27), 4695–4708. doi: 10.1002/sim.6308.

RStudio Team (2020). *RStudio: Integrated Development Environment for R*. RStudio, PBC., Boston, MA.

Searle, J. (1982). The myth of the computer. *The New York Review of Books*, 29 (April 29), 3–7. https://www.nybooks.com/articles/1982/04/29/the-myth-of-the-computer/.

Silver, N. (2012). *The Signal and the Noise: Why So Many Predictions Fail—But Some Don't*. New York, NY, USA: Penguin Press.

Steiner, A., Kolesnikov, A., Zhai, X., et al. (2022). How to train your ViT? Data, augmentation, and regularization in vision transformers. *arXiv*, 2106.10270. doi: 10.48550/arXiv.2106.10270.

van Dis, E. A. M., Bollen, J., van Rooij, R., et al. (2023). ChatGPT: Five priorities for research. *Nature*, 614(7947), 224–226. doi: 10.1038/d41586-023-00288-7.

Venables, W. N., Smith, D. M., & The R Core Team (2023). An Introduction to R. https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf. Accessed 11 February 2024.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. New York, NY, USA: Springer-Verlag, 2nd edition.

Wickham, H., Averick, M., Bryan, J., et al. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43), 1686. doi: 10.21105/joss.01686.

Zurell, D., Berger, U., Cabral, J. S., et al. (2010). The virtual ecologist approach: Simulating data and observers. *Oikos*, 119(4), 622–635. doi: 10.1111/j.1600-0706.2009.18284.x.

Index

Italic pagination indicates figures and tables, "n" indicates footnotes