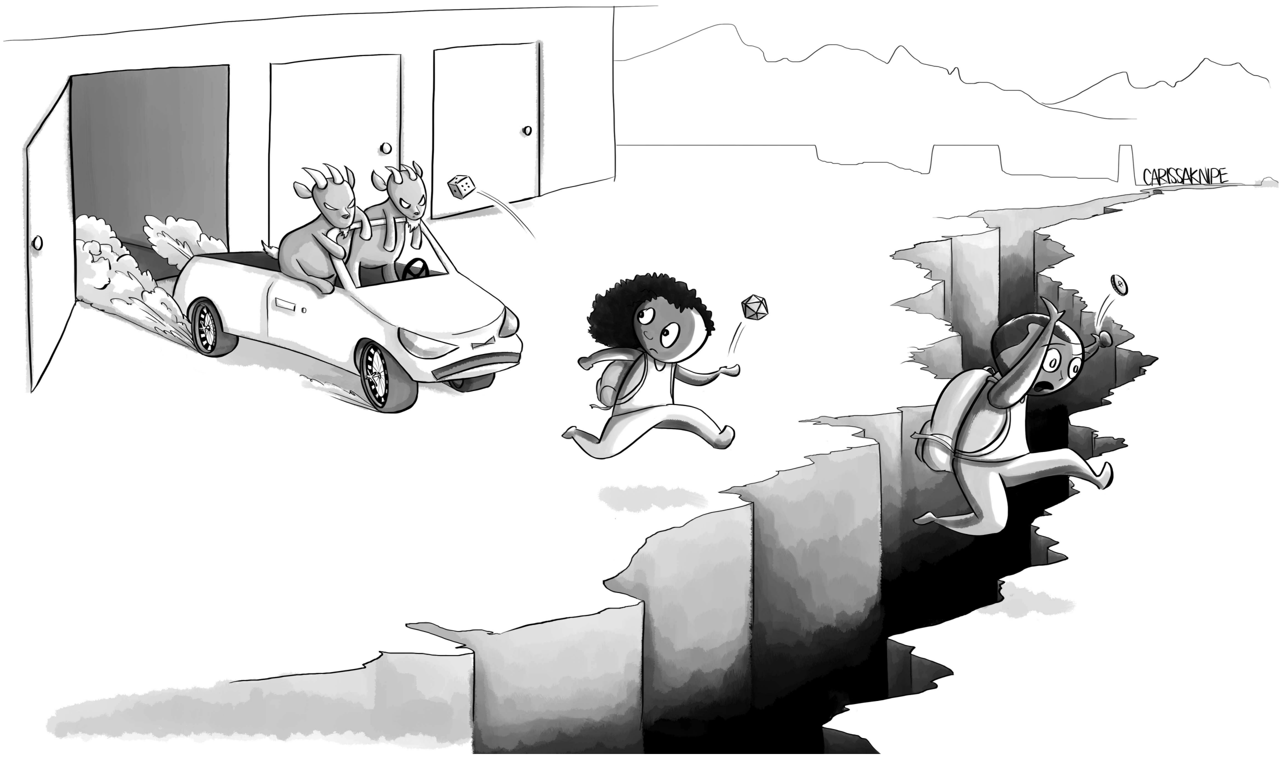
10
Probability



In which our heroes evade threats and conquer their fears by flipping coins, rolling dice, and spinning the wheels of chance.

10.1 Why You Might Care

> Fortune can, for her pleasure, fools advance,
> And toss them on the wheels of Chance.
>
> Juvenal (c. 55-–c. 127)

This chapter introduces *probability,* the study of randomness. Our focus, as will be no surprise by this point of the book, is on building a formal mathematical framework for analyzing random processes. We'll begin with a definition of the basics of probability: defining a random process that chooses one particular *outcome* from a set of possibilities (any one of which occurs some fraction of the time). We'll then analyze the likelihood that a particular *event* occurs—in other words, asking whether the chosen outcome has some particular property that we care about. We then consider *independence* and *dependence* of events, and *conditional probability*: how, if at all, does knowing that the randomly chosen outcome has one particular property change our calculation of the probability that it has a different property? (For example, perhaps 90% of all email is spam. Does knowing that a particular email contains the word `ENLARGE` make that email more than 90% likely to be spam?) Finally, we'll turn to *random variables* and *expectation,* which give quantitative measurements of random processes: for example, if we flip a coin 1000 times, how many heads would we see (on average)? How many runs of 10 or more consecutive heads? Probabilistic questions are surprisingly difficult to have good intuition about; the focus of the chapter will be on the tools required to rigorously settle these questions.

Probability is relevant almost everywhere in computer science. One broad application is in *randomized algorithms* to solve computational problems. In the same way that the best strategy to use in a game of rock–paper–scissors involves randomness (throw rock $\frac{1}{3}$ of the time, throw paper $\frac{1}{3}$ of the time, throw scissors $\frac{1}{3}$ of the time), there are some problems—for example, finding the median element of an unsorted array, or testing whether a given large integer is a prime number—for which the best known algorithm (the fastest, the simplest, the easiest to understand, ...) proceeds *by making random choices.* The same idea occurs in data structures: a *hash table* is an excellent data structure for many applications, and it's best when it assigns elements to (approximately) random cells of a table. (See Section 10.1.1.) Randomization can also be used for *symmetry breaking*: we can ensure that 1000 identical drones do not clog the airwaves by all trying to communicate simultaneously: each drone will choose to try to communicate at a random time. And we can generate more realistic computer graphics of flame or hair or, say, a field of grass by, for each blade, randomly perturbing the shape and configuration of an idealized piece of grass.

As a rough approximation, we can divide probabilistic applications in CS into two broad categories: those uses in which the randomness is internally generated by our algorithms or data structures, and those cases in which the randomness comes "from the outside." The first type we discussed above. In the latter category, consider circumstances in which we wish to build some sort of computational model that addresses some real-world phenomenon. For example, we might wish to model social behavior (a social network of friendships), or traffic on a road network or on the internet, or to build a speech recognition system. Because these applications interact with extremely complex real-world behaviors, we will typically think of them as being generated according to some deterministic (nonrandom) underlying rule, but with hard-to-model variation that is valuably thought of as generated by a random process. In systems for speech recognition, it works well to treat a particular "frame" of the speech stream (perhaps tens of milliseconds in duration) as a noisy version of the sound that the speaker intended to produce, where the noise is essentially a random perturbation of the intended sound.

Finally, you should care about probability because *any* well-educated person must understand something about probability. You need probability to understand political polls, weather forecasting, news reports about medical studies, wagers that you might place (either with real money or by choosing which of two alternatives is a better option), and many other subjects. Probability is everywhere!

10.1.1 Hashing: A Running Example

Throughout this chapter, we will consider a running sequence of examples that are about *hash tables*, a highly useful data structure that also conveniently illustrates a wide variety of probabilistic concepts. So we'll start here with a short primer on hash tables. (See also p. 267, or a good textbook on data structures.)

A *hash table* is a data structure that stores a set of elements in a table $T[1 \ldots m]$—that is, an array of size m. (Remember that, throughout this book, arrays are indexed starting at 1, not 0.) The set of possible elements is called the *universe* or the *keyspace*. We will be asked to store in this table a particular small subset of the keyspace. (For example, the keyspace might be the set of all 8-letter strings; we might be asked to store the user IDs of all students on campus.) We use a *hash function* h to determine in which cell of the table $T[1 \ldots m]$ each element will be stored. The hash function h takes elements of the keyspace as input, and produces as output an index identifying a cell in T. To store an element x in T using hash function h, we compute $h(x)$ and place x into the cell $T[h(x)]$. (We say that the element x *hashes to* the cell $T[h(x)]$.)

We must somehow handle *collisions*, when we're asked to store two different elements that hash to the same cell of T. We will usually consider the simplest solution, where we use a strategy called *chaining* to resolve collisions. To implement chaining, we store all elements that hash to a cell *in that cell*, in an unsorted list. Thus, to find whether an element y is stored in the hash table T, we look one-by-one through the list of elements stored in $T[h(y)]$.

Example 10.1 (A small hash table)
Let the keyspace be $\{1, 2, 3, 4\}$, and consider a 2-cell hash table with the hash function h given by $h(x) = (x \bmod 2) + 1$. (Thus $h(1) = h(3) = 2$ and $h(2) = h(4) = 1$.)

- If we store the elements $\{1, 4\}$, then the table would be $\begin{array}{|c|c|} \multicolumn{1}{c}{T[1]} & \multicolumn{1}{c}{T[2]} \\ \hline [4] & [1] \\ \hline \end{array}$.
- If we store the elements $\{2, 4\}$, then the table would be $\begin{array}{|c|c|} \hline [2, 4] & [] \\ \hline \end{array}$.

More formally, we are given a finite set K called the *keyspace*, and we are also given a positive integer m representing the table size. We will base the data structure on a hash function $h : K \to \{1, \ldots, m\}$. For the purposes of this chapter, we choose h *randomly*, specifically choosing the hash function so that *each function from K to $\{1, \ldots, m\}$ is equally likely to be chosen as h.*

Let's continue our above example with a randomly chosen hash function. For the moment, we'll treat the process of randomly choosing a hash function informally. (The precise definitions of what it means to choose randomly, and what it means for certain "events" to occur, will be defined in the following sections.)

Example 10.2 (A small hash table)
As before, let $K = \{1, 2, 3, 4\}$ and $m = 2$. There are $m^{|K|} = 2^4 = 16$ different functions $h : K \to \{1, 2\}$, and each of these functions is equally likely to be chosen. (The functions are listed in Figure 10.1.) Each of these functions is chosen a $\frac{1}{16}$ fraction of the time. Thus:

- a $\frac{8}{16} = \frac{1}{2}$ fraction of the time, we have $h(4) = h(1)$.
  (These functions are marked with an 'A' in Figure 10.1.)
- a $\frac{6}{16} = \frac{3}{8}$ fraction of the time, the hash function is "perfectly balanced"—that is, hashes an equal share of the keys to each cell.
  (These functions are marked with a 'B' in Figure 10.1.)
- a $\frac{1}{16}$ fraction of the time, the hash function hashes every element of K into cell #2.
  (This one function is marked with a 'C' in Figure 10.1.)

$h(1)$	$h(2)$	$h(3)$	$h(4)$	
1	1	1	1	A
1	1	1	2	
1	1	2	1	A
1	1	2	2	B
1	2	1	1	A
1	2	1	2	B
1	2	2	1	AB
1	2	2	2	
2	1	1	1	
2	1	1	2	AB
2	1	2	1	B
2	1	2	2	A
2	2	1	1	B
2	2	1	2	A
2	2	2	1	
2	2	2	2	A C

Figure 10.1: All functions from $\{1, 2, 3, 4\}$ to $\{1, 2\}$. Each row is a different function h; the ith column records the value of $h(i)$. The letters mark some functions as described in Example 10.2.

Taking it further: In practice, the function h will not be chosen completely at random, for a variety of practical reasons (for example, we'd have to write down the whole function to remember it!), but throughout this chapter we will model hash tables as if h is chosen completely randomly. The assumption that the hash function is chosen randomly, with every function $K \to \{1, 2, \ldots, m\}$ equally likely to be chosen, is called the *simple uniform hashing assumption.* It is very common to make this assumption when analyzing hash tables.

It may be easier to think of choosing a random hash function using an iterative process instead: for every key $x \in K$, we choose a number i_x uniformly at random and independently from $\{1, 2, \ldots, m\}$. (The definitions of "uniformly" and "independently" are coming in the next few sections. Informally, this description means that each number in $\{1, 2, \ldots, m\}$ is equally likely to be chosen as i_x, regardless of what choices were made for previous numbers.) Now define the function h as follows: on input x, output i_x. One can prove that this process is completely identical to the process illustrated in Example 10.2: write down every function from K to $\{1, 2, \ldots, m\}$ (there are $m^{|K|}$ of them), and pick one of these functions at random.

After we've chosen the hash function h, a set of actual keys $\{x_1, \ldots, x_n\} \subseteq K$ will be given to us, and we will store the element x_i in the table slot $T[h(x_i)]$. Notice that the *only* randomly determined quantity is the hash function h. Everything else—the keyspace K, the table size m, and the set of to-be-stored elements—is fixed.

10.2 *Probability, Outcomes, and Events*

> Anyone who does not know how to make the most of his luck has no right to complain if it passes by him.
>
> Miguel de Cervantes (1547–1616)

This section will give formal definitions of the fundamental concepts in probability, giving us a framework to use in thinking about the many computational applications that involve chance. These definitions are somewhat technical, but they'll allow us reason about some fairly sophisticated probabilistic settings fairly quickly.

Warning! It is very rare to have good intuition or instincts about probability questions. Try to hold yourself back from jumping to conclusions too quickly, and instead use the systematic approaches to probabilistic questions that are introduced in this chapter.

10.2.1 *Outcomes and Probability*

Here's the very rough outline of the relevant definitions; we'll give more details in a moment. Imagine a scenario in which some quantity is determined in some random way. We will consider a set S of possible *outcomes.* Each outcome has an associated *probability*, which is a number between 0 and 1. The set S is called the *sample space.* In any particular result of this scenario, one outcome from S is selected randomly (by "nature"); the frequency with which a particular outcome is chosen is given by that outcome's associated probability. (Sometimes we might talk about the *process* by which a sequence of random quantities is selected, and the *realization* as the actual choice made according to this process.) For example, for flipping an unweighted coin we would have $S = \{\text{Heads}, \text{Tails}\}$, where Heads has probability 0.5 and Tails has probability 0.5. Our particular outcome might be Heads.

Here are the formal definitions:

Definition 10.1 (Outcomes and sample space)
An outcome *of a probabilistic process is the sequence of results for all randomly determined quantities. (An outcome can also be called a* realization *of the probabilistic process.) The* sample space *S is the set of all outcomes.*

Definition 10.2 (Probability function)
Let S be a sample space. A probability function $\texttt{Pr} : S \to \mathbb{R}$ *describes, for each outcome $s \in S$, the fraction of the time that s occurs. (We denote probabilities using square brackets, so the probability of $s \in S$ is written $\texttt{Pr}[s]$.) We insist that the following two conditions hold of the probability function* Pr*:*

$$\sum_{s \in S} \texttt{Pr}[s] = 1 \qquad (10.1)$$

$$\texttt{Pr}[s] \geq 0 \text{ for all } s \in S. \qquad (10.2)$$

Intuitively, condition (10.1) says that *something has to happen*: when we flip a coin, then either it comes up heads or it comes up tails. (And so $\texttt{Pr}[\text{Heads}] + \texttt{Pr}[\text{Tails}] = 1$.) The other condition, (10.2), formalizes the idea that $\texttt{Pr}[s]$ denotes the fraction of the time that the outcome s occurs: *the least frequently that an outcome can occur is never.*

The probability function $\mathtt{Pr}$ is also sometimes called a *probability distribution over S*. (This function "distributes" one unit of probability across the set S of all possible outcomes, as in (10.1).)

> **Taking it further:** Bizarrely, in *quantum computation*—an as-yet-theoretical type of computation based on quantum mechanics—we can have outcomes whose probabilities are not restricted to be real numbers between 0 and 1. This model is (very!) difficult to wrap one's mind around, but a computer based on this idea turns out to let us solve interesting problems, and faster than on "normal" computers. For example, we can factor large numbers efficiently on a quantum computer. (Though we don't know how to build quantum computers of any nontrivial size.) See p. 1016 for some discussion.

A few examples: cards, coins, and words

Here are a few examples of sample spaces with probabilities naturally associated with each outcome:

Example 10.3 (One card from the deck)
We draw one card from a perfectly shuffled deck of 52 cards. Then we can denote the sample space as $S = \{2, 3, \ldots, 10, \mathrm{J}, \mathrm{Q}, \mathrm{K}, \mathrm{A}\} \times \{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}$. Each card $c \in S$ has $\mathtt{Pr}[c] = \frac{1}{52}$. Note that condition (10.1) is satisfied because

$$\textstyle\sum_{c \in S} \mathtt{Pr}[c] = \sum_{c \in S} \frac{1}{52} = 52 \cdot \frac{1}{52} = 1,$$

and (10.2) is obviously satisfied because $\mathtt{Pr}[c] = \frac{1}{52} \geq 0$ for each c.

Example 10.4 (Coin flips)
You flip a quarter and Bill Gates flips a platinum trillion-dollar coin. Assume that both coins are fair (equally likely to come up Heads and Tails) and that flips of the quarter and the platinum coin do not affect each other in any way. Then the four outcomes are—writing the quarter's result first—$\langle \text{Heads}, \text{Heads} \rangle$, $\langle \text{Heads}, \text{Tails} \rangle$, $\langle \text{Tails}, \text{Heads} \rangle$, and $\langle \text{Tails}, \text{Tails} \rangle$. Each of these four outcomes has probability 0.25.

Example 10.5 (A word on the page)
Consider the following sentence, which—excluding spaces—contains a total of 29 different symbols (namely $\mathtt{N}, \mathtt{o}, \mathtt{w}, \mathtt{i}, \mathtt{s}, \mathtt{t}, \ldots, \mathtt{t}$):

```
Now is the winter of our discontent.
```

We are going to select a word from this sentence, according to the following process: choose one of the 29 non-space symbols from the sentence with equal likelihood; the selected word is the one in which the selected symbol appears. (Thus longer words will be chosen more frequently than shorter words, because longer words contain more symbols—and are therefore more likely to be selected.)

The sample space is $S = \{\mathtt{Now}, \mathtt{is}, \mathtt{the}, \mathtt{winter}, \mathtt{of}, \mathtt{our}, \mathtt{discontent}\}$. There are $3 + 2 + 3 + 6 + 2 + 3 + 10 = 29$ total symbols, and thus $\mathtt{Pr}[\mathtt{Now}] = \frac{3}{29}$, $\mathtt{Pr}[\mathtt{is}] = \frac{2}{29}$, and so on, through $\mathtt{Pr}[\mathtt{discontent}] = \frac{10}{29}$. Again, the conditions for being a probability are satisfied: each outcome's probability is nonnegative, and $\sum_{w \in S} \mathtt{Pr}[w] = 1$.

Now is the winter of our discontent/ Made glorious summer by this sun of York;/ And all the clouds that lour'd upon our house/ In the deep bosom of the ocean buried.
— William Shakespeare (1564–1616)
King Richard III

Examples 10.3 and 10.4 are scenarios of *uniform probability*, in which each outcome in the sample space is chosen with equal likelihood. (Specifically, each $s \in S$ has probability $\texttt{Pr}[s] = \frac{1}{|S|}$.) Example 10.5 illustrates *nonuniform probability*, in which some outcomes occur more frequently than others.

Note that for a single sample space S, we can have many different distinct processes by which we choose an outcome from S. For example:

Example 10.6 (Two ways of choosing from $S = \{0, 1, 2, \ldots, 7\}$)
One process for selecting an element of S is to flip three fair coins and treat their results as a binary number (HHH $= 111 \to 7$, HHT $= 110 \to 6, \ldots,$ TTT $= 000 \to 0$). This process gives a uniform distribution over S: each sequence of coin flips occurs with the same probability. For example, $\texttt{Pr}[4] = \frac{1}{8} = 0.125$ and $\texttt{Pr}[7] = \frac{1}{8} = 0.125$.

A second process for selecting an element of S is to flip 7 fair coins and to let the outcome be the number of heads that we see in those 7 flips (HHHHHHH $\to$ 7, HHHHHHT $\to$ 6, HHHHHTH $\to$ 6, $\ldots$, TTTTTTT $\to$ 0). This process gives a *nonuniform* distribution over S, because the number of sequences that have k heads is different for different values of k. For example:

$$\texttt{Pr}[4] = \frac{\binom{7}{4}}{2^7} = \frac{35}{128} \approx 0.2734, \qquad \text{but} \qquad \texttt{Pr}[7] = \frac{\binom{7}{7}}{2^7} = \frac{1}{128} \approx 0.0078.$$

As a word of warning, notice that probabilistic statements *about a particular realization* don't make sense; the only kind of probabilistic statement that makes sense is a statement *about a probabilistic process.* If you happen to be one of the $\approx 10\%$ of the population that's red–green colorblind, and a friend says "what are the odds that you're colorblind!?", the correct answer is: the probability is 1 (because it happened!).

10.2.2 Events

Many of the probabilistic questions that we'll ask are about whether the realization has some particular property, rather than whether a single particular outcome occurs. For example, we might ask for the probability of getting more heads than tails in 1000 flips of a fair coin. Or we might ask for the probability that a hand of seven cards (dealt from a perfectly shuffled deck) contains at least two pairs. There may be many different outcomes in the sample space that have the property in question. Thus, often we will be interested in the probability of a *set* of outcomes, rather than the probability of a *single* outcome. Such a set of outcomes is called an *event*:

Definition 10.3 (Event)
Let S be a sample space with probability function $\texttt{Pr}$*. An* event *is a subset of S. The* probability of an event *E is the sum of the probabilities of the outcomes in E, and it is written* $\texttt{Pr}[E] = \sum_{s \in E} \texttt{Pr}[s]$.

The probability of an event $E \subseteq S$ follows by a probabilistic version of the Sum Rule, from counting: because one (and only one) outcome is chosen in a particular realization, the probability of either outcome x or y occurring is $\texttt{Pr}[x] + \texttt{Pr}[y]$.

Note that the notation in Definition 10.3 generalizes the function Pr by allowing us to write *either* elements of S *or* subsets of S as inputs to Pr. That is, previously we considered a function $\texttt{Pr} : S \to [0,1]$; we have now "extended" our notation so that it's a function $\texttt{Pr} : \mathscr{P}(S) \to [0,1]$. (To be more precise, we're actually extending the notation to be a function $\texttt{Pr} : (S \cup \mathscr{P}(S)) \to [0,1]$, because we're still letting ourselves write outcomes as arguments too.)

Our mixture of $\texttt{Pr}[\text{outcome}]$ and $\texttt{Pr}[\text{event}]$ is an abuse of notation; we're mixing the type of input willy nilly. But, because $\texttt{Pr}[x]$ for an outcome x and $\texttt{Pr}[\{x\}]$ for the singleton event $\{x\}$ are identical, we can write probabilities this way without risk of confusion.

A few examples

Here are a few examples of events and their probabilities:

Example 10.7 (At least one head)
You and Bill Gates each flip fair coins, as in Example 10.4. Define the event $H = \{\langle \text{Heads}, \text{Heads} \rangle, \langle \text{Heads}, \text{Tails} \rangle, \langle \text{Tails}, \text{Heads} \rangle\}$ as "at least one coin comes up heads." Then $\texttt{Pr}[H] = 0.25 + 0.25 + 0.25 = 0.75$.

Example 10.8 (Aces up)
Problem: Suppose that you draw one card from a perfectly shuffled deck, as in Example 10.3. What is the probability that you draw an ace?

Solution: The event in question is $E = \{A\clubsuit, A\diamondsuit, A\heartsuit, A\spadesuit\}$. Each of these four outcomes has a probability of $\frac{1}{52}$, so $\texttt{Pr}[E] = \frac{1}{52} + \frac{1}{52} + \frac{1}{52} + \frac{1}{52} = \frac{4}{52} = \frac{1}{13}$.

Example 10.9 (Full house)
Problem: You're dealt 5 cards from a shuffled deck, so that each set of 5 cards is equally likely to be your hand. A hand is a *full house* if 3 cards share one rank, and the other 2 cards share a second rank. (For example, the hand $3\heartsuit, 3\spadesuit, 9\heartsuit, 9\clubsuit, 3\clubsuit$ is a full house.) What's the probability of being dealt a full house?

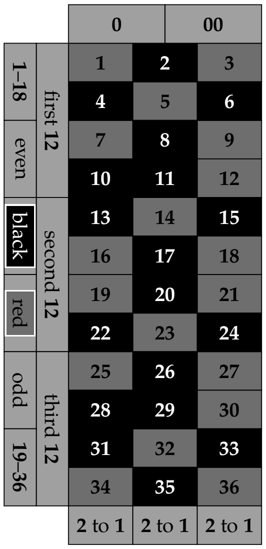
Solution: There are $\binom{52}{5}$ possible hands, each of which is dealt with probability $1/\binom{52}{5}$. Thus the key question is a counting question: *how many full houses are there?* We can compute this number using the Generalized Product Rule; specifically, we can view a full house as the result of the following sequence of selections:

- we choose the rank of which to have three of a kind;
- we choose which 3 of the 4 cards of that rank are in the hand;
- we choose the rank of the pair (any of the 12 remaining ranks); and
- we choose which 2 of the 4 cards of that rank are in the hand.

Thus there are $\binom{13}{1} \cdot \binom{4}{3} \cdot \binom{12}{1} \cdot \binom{4}{2}$ full houses, and the probability of a full house is

$$\frac{\binom{13}{1} \cdot \binom{4}{3} \cdot \binom{12}{1} \cdot \binom{4}{2}}{\binom{52}{5}} = \frac{3744}{2598960} \approx 0.00144.$$

Here's a slightly more complex example, with multiple events of interest:



event name	outcomes	probability
1–18	$\{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18\}$	$\frac{18}{38}$
even	$\{2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36\}$	$\frac{18}{38}$
1st 12	$\{1,2,3,4,5,6,7,8,9,10,11,12\}$	$\frac{12}{38}$
black	$\{2,4,6,8,10,11,13,15,17,20,22,24,26,28,29,31,33,35\}$	$\frac{18}{38}$
red	$\{1,3,5,7,9,12,14,16,18,19,21,23,25,27,30,32,34,36\}$	$\frac{18}{38}$
2nd 12	$\{13,14,15,16,17,18,19,20,21,22,23,24\}$	$\frac{12}{38}$
odd	$\{1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35\}$	$\frac{18}{38}$
19–36	$\{19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36\}$	$\frac{18}{38}$
3rd 12	$\{25,26,27,28,29,30,31,32,33,34,35,36\}$	$\frac{12}{38}$
"2 to 1" A	$\{1,4,7,10,13,16,19,22,25,28,31,34\}$	$\frac{12}{38}$
"2 to 1" B	$\{2,5,8,11,14,17,20,23,26,29,32,35\}$	$\frac{12}{38}$
"2 to 1" C	$\{3,6,9,12,15,18,21,24,27,39,33,36\}$	$\frac{12}{38}$

Figure 10.2: The Roulette board, and the corresponding events.

Example 10.10 (Roulette)

In *roulette*, a wheel is spun, and a metal ball comes to rest in one of the wheel's 38 segments. The segments are numbered 1–36 (each colored red or black, as shown in Figure 10.2), and there are two more segments labeled 0 and 00 (both colored green). Assume that the ball is equally likely to land in each segment, and that the sample space consists of $\{00, 0, 1, 2, \ldots, 36\}$. There are 38 outcomes in the sample space.

In addition to particular outcomes or pairs/triples/quadruples of outcomes whose numbered squares are adjacent in the board, a roulette player can bet on a number of different events defined by the twelve panels along the left-hand and bottom sides of the grid. These events, and their probabilities, are shown in Figure 10.2. (For roulette purposes, the numbers 0 and 00 count as *neither* even nor odd—for reasons related only to casinos' business models, and not to the value of 0 mod 2.)

This version of roulette is the American wheel; the European wheel has only one green zero segment, and there are only 37 outcomes. The player in the European version does better on average.

The details of the particular roulette events in Example 10.10 aren't particularly important, but the distinction between outcomes and events—which this example should make starkly (it's the difference between "the ball stops on number 17" and "the ball stops on an odd number")—is crucial in probability.

It is often useful to visualize a sample space, and the events of interest, using a Venn diagram–like representation. It can be particularly helpful to draw the subsets/events in such a way that their area corresponds to their probability. A small example of this visualization, for some of the roulette events from Example 10.10, is shown in Figure 10.3. This figure also shows a few intersections of pairs of events: because an event is just a subset of the sample space, the intersection of two events is still a subset of the sample space, and therefore is also an event.

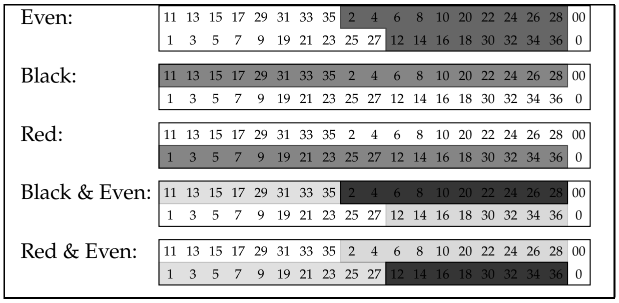


Figure 10.3: A visualization of the sample space and a few events from roulette.

Here are a few useful general properties of the probability of events:

Theorem 10.1 (Some properties of event probabilities)
Let S be a sample space, and let $A \subseteq S$ and $B \subseteq S$ be events. Then, writing $\overline{A} := S - A$ to denote the complement of the event A, we have:

$$\texttt{Pr}\left[S\right] = 1 \tag{10.1.1}$$

$$\texttt{Pr}\left[\varnothing\right] = 0 \tag{10.1.2}$$

$$\texttt{Pr}\left[\overline{A}\right] = 1 - \texttt{Pr}\left[A\right] \tag{10.1.3}$$

$$\texttt{Pr}\left[A \cup B\right] = \texttt{Pr}\left[A\right] + \texttt{Pr}\left[B\right] - \texttt{Pr}\left[A \cap B\right]. \tag{10.1.4}$$

These properties all follow directly from the definition of the probability of an event.

10.2.3 Tree Diagrams in Probability

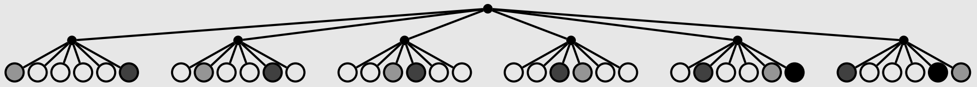
Many probabilistic processes involve a *sequence* of randomly determined quantities, rather than just a single random choice. Much like in counting, we can use a *tree diagram* to represent the sequence of random choices—and then we can look for the probability of a particular outcome as reflected in the sequence of choices in the tree. In a tree diagram for a probabilistic sequence of choices:

Problem-solving tip: Tree diagrams are generally a very good way to solve probability questions; they force you to systematically think about all of the steps of a probabilistic process (and also about all of the steps of solving probability problems!).

- Every internal node in the tree corresponds to a random decision; every edge leaving that internal node is labeled with the probability of a particular decision. The probability labels of all edges leaving any particular internal node u must add up to 1. (The interpretation is: if the probabilistic process reaches node u, then each branch leaving u is chosen with frequency in proportion to its label.)
- Every leaf corresponds to an outcome. The probability of reaching a particular leaf is precisely equal to the product of the labels on the edges leading from the root to that leaf. As usual, the probability of an event is the sum of the probabilities of the outcomes contained in that event.

Here is a first small example:

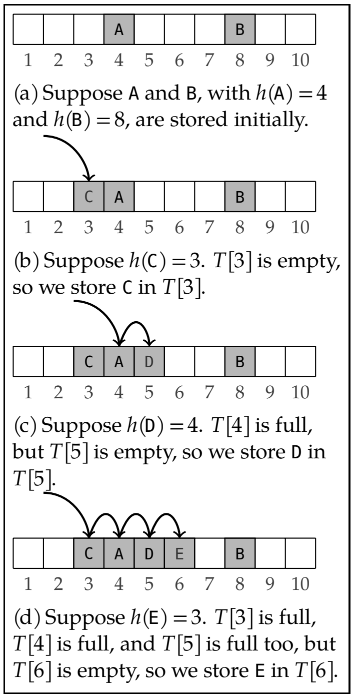
Example 10.11 (Rolling two dice)
Here's the probability tree for rolling two fair dice, one after the other. Outcomes are listed in order from $\langle 1,1 \rangle$ at left to $\langle 6,6 \rangle$ at right; every edge has probability $\frac{1}{6}$.



- All edges have probability $\frac{1}{6}$; thus each outcome's probability is $\frac{1}{6} \cdot \frac{1}{6} = \frac{1}{36}$.
- The event "doubles are rolled" (light-shaded outcomes) has probability $6 \cdot \frac{1}{36} = \frac{1}{6}$.
- The event "a 7 is rolled" (medium-shaded outcomes) has probability $6 \cdot \frac{1}{36} = \frac{1}{6}$.
- The event "an 11 is rolled" (black-shaded outcomes) has probability $2 \cdot \frac{1}{36} = \frac{1}{18}$.

Incidentally, one of the calculations from Example 10.11 can also be rephrased to address a question about hashing. Suppose that we hash two elements into a hash table with 6 slots using a uniform random hash function. (See Section 10.1.1.) What is the probability that we have a collision? In fact, this question is precisely the same as asking for the probability of rolling doubles with two fair dice—that is, $\frac{1}{6}$, by Example 10.11.

When we introduced hash tables in Section 10.1.1, we described resolving collisions by *chaining:* an element x is stored in cell $T[h(x)]$; if that cell is already occupied, then we simply add x to a *list* of elements in cell $T[h(x)]$. But there are several other strategies for resolving collisions in a hash table, the simplest of which is called *linear probing.* In linear probing, when we insert an element x into the table, we put x in *the first unoccupied cell*, moving from left to right, starting at cell $h(x)$. (That is, we try to put x in cell $T[h(x)]$, but if that cell already has an element, then we try to put x into cell $T[h(x)+1]$, and then cell $T[h(x)+2]$, and so on. We wrap around to $T[1]$ after we reach the right edge of T.) See Figure 10.4 for an example.



(a) Suppose A and B, with $h(\texttt{A}) = 4$ and $h(\texttt{B}) = 8$, are stored initially.

(b) Suppose $h(\texttt{C}) = 3$. $T[3]$ is empty, so we store C in $T[3]$.

(c) Suppose $h(\texttt{D}) = 4$. $T[4]$ is full, but $T[5]$ is empty, so we store D in $T[5]$.

(d) Suppose $h(\texttt{E}) = 3$. $T[3]$ is full, $T[4]$ is full, and $T[5]$ is full too, but $T[6]$ is empty, so we store E in $T[6]$.

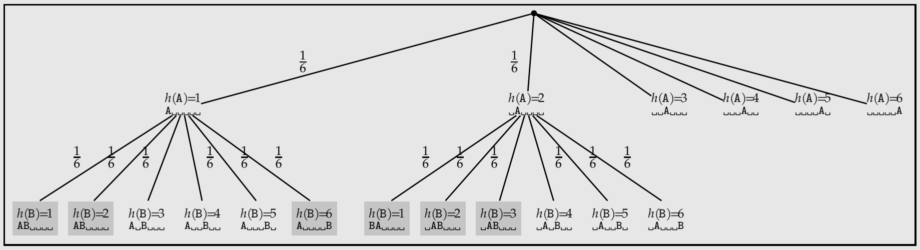
Figure 10.4: Linear probing.

Taking it further: One of the downsides of resolving collisions in a hash table using linear probing is a phenomenon called "clustering": contiguous blocks of filled cells develop, and these filled blocks tend to get longer and longer as more and more elements are added to the table. (This problem is beginning to occur in Figure 10.4.) Other collision-resolution schemes can mitigate this problem; see Exercises 10.45–10.50.

Example 10.12 (Hashing with linear probing)

Problem: Suppose that we hash 2 elements into a hash table with 6 slots using a uniform random hash function h, where we resolve collisions by linear probing. What is the probability that we end up with 2 consecutive slots of the hash table filled?

Solution: The sample space is $S = \{1, 2, \ldots, 6\} \times \{1, 2, \ldots, 6\}$: we first randomly choose a value for $h(\texttt{A})$, and then randomly choose a value for $h(\texttt{B})$. We'll build a tree diagram to represent these choices, as shown (in part) here:



The highlighted outcomes have A and B hashed to adjacent cells. (The remainder of the tree is analogous; it's good practice to try drawing the other branches.)

Each branch of the tree is equally likely, so each outcome occurs with probability $\frac{1}{36}$. How many different outcomes result in A and B being stored in adjacent cells? For each of the 6 possible hash values for A, there are 3 hash values for B that cause A and B to be adjacent, when $h(\texttt{B})$ is one of $h(\texttt{A}) - 1$, $h(\texttt{A})$, and $h(\texttt{A}) + 1$. So the final probability of a cluster forming is $(6 \cdot 3) \cdot \frac{1}{36} = \frac{18}{36} = \frac{1}{2}$.

Here's another (by now famous) example, called the *Monty Hall Problem*, in which using a probability tree helps resolve a potentially confusing probability question:

The Monty Hall Problem is named after Monty Hall, the host of the television game show *Let's Make A Deal* in the 1960s–1980s. The problem became famous after a kerfuffle involving Marilyn vos Savant, in *Parade* magazine.

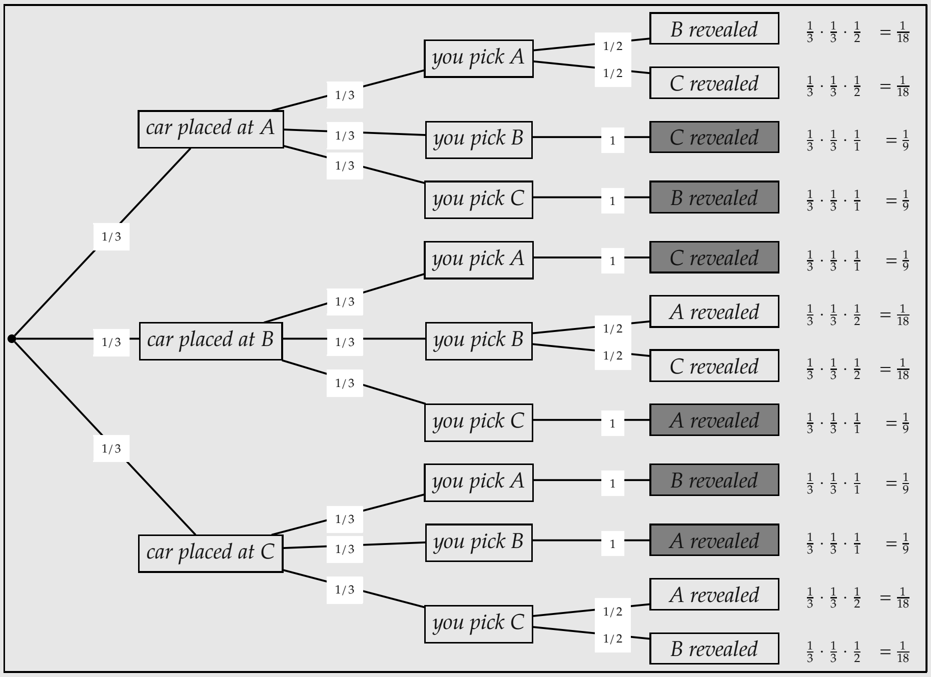
Example 10.13 (Monty Hall Problem)

Problem: Here is the problem (based on the *Let's Make a Deal* setup):

> You are given the choice of three doors, behind which are a car, a goat, and another goat. You choose a door. Monty Hall opens one of the doors that you didn't choose to reveal a goat. He then offers you the chance to switch to the other (unopened) door that you didn't initially choose. Should you switch?

(To make this question concrete, assume that the car is initially placed randomly; you choose an initial door randomly; the host always opens one of the two doors you didn't choose to reveal a goat, choosing a goat at random if there are two unchosen goats; and the host will always give you an opportunity to switch.)

Solution: There are three randomly chosen quantities: where the car is placed, which door you choose, and which goat is revealed (if there are two possibilities). We can express the process using the following probability tree:


1/3
car placed at A
1/3
you pick A
1/2
B revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{18}$
1/2
C revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{18}$
1/3
you pick B
1
C revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{1} = \frac{1}{9}$
1/3
you pick C
1
B revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{1} = \frac{1}{9}$
1/3
car placed at B
1/3
you pick A
1
C revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{1} = \frac{1}{9}$
1/3
you pick B
1/2
A revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{18}$
1/2
C revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{18}$
1/3
you pick C
1
A revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{1} = \frac{1}{9}$
1/3
car placed at C
1/3
you pick A
1
B revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{1} = \frac{1}{9}$
1/3
you pick B
1
A revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{1} = \frac{1}{9}$
1/3
you pick C
1/2
A revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{18}$
1/2
B revealed
$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{18}$

car	your choice	revealed goat	probability
A	A	B	1/18
A	A	C	1/18
A	B	C	1/9
A	C	B	1/9
B	A	C	1/9
B	B	A	1/18
B	B	C	1/18
B	C	A	1/9
C	A	B	1/9
C	B	A	1/9
C	A	B	1/18
C	B	A	1/18

Figure 10.5: The 12 outcomes in the Monty Hall sample space, and their associated probabilities. The shaded outcomes are those where you win by switching.

The shaded outcomes are those in which switching from your initially chosen door causes your new door to hide a car; the unshaded outcomes are those in which not switching causes you to win. These outcomes and their associated probabilities are also shown in Figure 10.5; again, in the shaded outcomes you win by switching.

There are six outcomes in which switching causes you to win the car. Each of these outcomes has probability $\frac{1}{9}$, so the probability of winning a car by switching is $6 \cdot \frac{1}{9} = \frac{2}{3}$. The other six outcomes are those in which *not* switching causes you to get a car (and switching gets you a goat); these outcomes each have probability $\frac{1}{18}$, and so the probability of winning by not switching is $6 \cdot \frac{1}{18} = \frac{1}{3}$. *You should switch.*

Problem-solving tip: It is usually worth the time to make the probabilistic process concrete, and to make explicit any hidden assumptions about the process, before solving the problem. (That's how we began Example 10.13.)

Taking it further: Section 10.2.3 has been devoted to tree diagrams—a systematic way of analyzing probabilistic settings in which a sequence of random choices is made. Typically we think of—or at least model—these random choices as being made "by nature": if you flip a coin, you act as though the universe "chooses" (via microdrafts of wind, the precise topology of the ground where the coin bounces, etc.) whether the coin will come up Heads or Tails.

But, in many scenarios in computer science, we want to generate the randomness *ourselves*, perhaps in a program: choose a random element of the set A; go left with probability $\frac{1}{2}$ and go right with probability $\frac{1}{2}$; generate a random 8-symbol password. The process of *actually generating* a sequence of "random" numbers on a computer is difficult, and (perhaps surprisingly) very closely tied to notions of cryptographic security. A *pseudorandom generator* is an algorithm that produces a sequence of bits that seem to be random, at least to someone examining the sequence of generated bits with limited computational power. It turns out that building a difficult-to-break encryption system is in a sense equivalent to building a difficult-to-distinguish-from-random pseudorandom generator.[1]

10.2.4 Some Common Probability Distributions

We'll end this section by spending a few words on some of the common probabilistic processes (and therefore some common probability distributions) that arise in computer science applications.

Uniform distribution

Under the *uniform distribution*, every outcome is equally likely. We can define a uniform distribution for any finite sample space S:

Definition 10.4 (Uniform distribution)
Let S be a finite sample space. Under the uniform distribution, the probability of any particular outcome $s \in S$ is given by $\texttt{Pr}[s] = \frac{1}{|S|}$.

Some familiar examples of the uniform distribution include:

- flipping a fair coin ($\texttt{Pr}[\text{Heads}] = \texttt{Pr}[\text{Tails}] = \frac{1}{2}$).
- rolling a fair 6-sided die ($\texttt{Pr}[1] = \texttt{Pr}[2] = \texttt{Pr}[3] = \texttt{Pr}[4] = \texttt{Pr}[5] = \texttt{Pr}[6] = \frac{1}{6}$).
- choosing one card from a shuffled deck ($\texttt{Pr}[c] = \frac{1}{52}$ for any card c).

Note that, if outcomes are chosen uniformly at random, then the probability of an event is simply its fraction of the sample space. That is, for any event $E \subseteq S$, we have

$$\texttt{Pr}[E] = \frac{|E|}{|S|}.$$

Taking it further: We often make use of a uniform distribution in randomized algorithms. For example, in randomized quicksort or randomized select applied to an array $A[1 \ldots n]$, a key step is to choose a "pivot" value uniformly at random from A, and then use the chosen value to guide subsequent operation of the algorithm. (See Exercises 10.24–10.27.)

Bernoulli distribution

The next several distributions are related to "flipping coins" in various ways. "Coin flipping" is a common informal way of referring to any probabilistic process is which we have one or more *trials*, where each trial has the same "success probability," also known as "getting heads." We will refer to flipping an actual coin as a coin flip, but we will also refer to other probabilistic processes that succeed with some fixed probability

For more, see:
[1] Oded Goldreich. *Foundations of Cryptography*. Cambridge University Press, 2006.

as a coin flip. We will consider a (possibly) *biased coin*—that is, a coin that comes up heads with probability p, and comes up tails with probability $1 - p$. The coin is called *fair* if $p = \frac{1}{2}$; that is, if the probability distribution is uniform. We can call the coin *p-biased* when $\texttt{Pr}\,[\text{heads}] = p$. It's important that the result of one trial has no effect on the success probability of any subsequent trial. (That is, these flips are *independent*; see Section 10.3.)

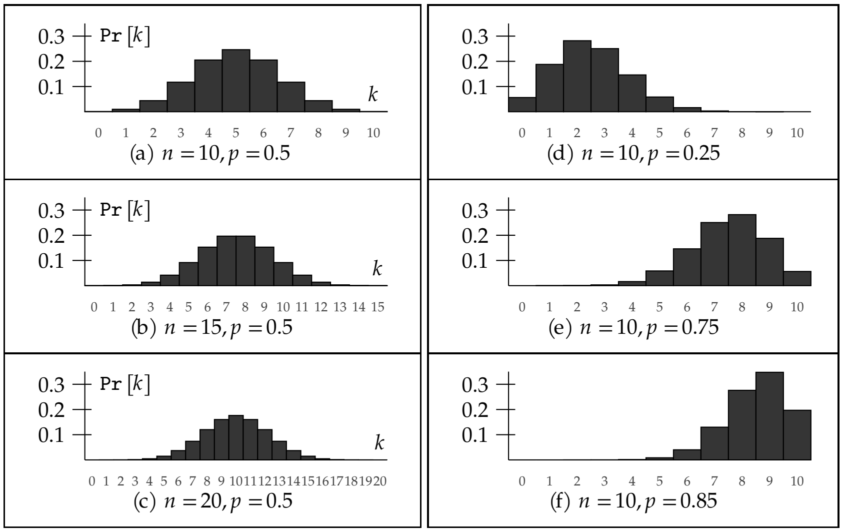
The first coin-related distribution is simply the one associated with a single trial:

Definition 10.5 (Bernoulli distribution)
The Bernoulli distribution with parameter *p is the probability distribution that results from flipping one p-biased coin. Thus the sample space is* $\{\text{H}, \text{T}\}$*, where* $\texttt{Pr}\,[\text{H}] = p$ *and* $\texttt{Pr}\,[\text{T}] = 1 - p$*.*

The Bernoulli distribution is named after Jacob Bernoulli, a 17th-century Swiss mathematician.

Taking it further: Imagine a sequence of Bernoulli trials performed with $p = 0.01$, and another sequence of Bernoulli trials performed with $p = 0.48$. The former sequence will consist almost entirely of zeros; the latter will be about half zeros and about half ones. There's a precise technical sense in which the second sequence *contains more information* than the first, measured in terms of the *entropy* of the sequence. See p. 1017 for some discussion.

Binomial distribution



(a) $n = 10, p = 0.5$
(d) $n = 10, p = 0.25$
(b) $n = 15, p = 0.5$
(e) $n = 10, p = 0.75$
(c) $n = 20, p = 0.5$
(f) $n = 10, p = 0.85$

Figure 10.6: Several binomial distributions, for different values of n and p.

A somewhat more interesting distribution results from considering a *sequence* of flips of a biased coin. Consider the following probabilistic process: perform n flips of a p-biased coin, and then count the number of heads in those flips. The *binomial distribution with parameters n and p* is a distribution over the sample space $\{0, 1, \ldots, n\}$, where $\texttt{Pr}\,[k]$ denotes the probability of getting precisely k heads in those flips. Figure 10.6 shows several examples of binomial distributions, for different settings of the parameters n and p. Each panel of Figure 10.6 shows the probability $P[k]$ of getting precisely k heads in n flips of a p-biased coin, for each k in the sample space.

If we flip a p-biased coin n times, what is the probability of the event of getting exactly k heads? For example, consider the outcome

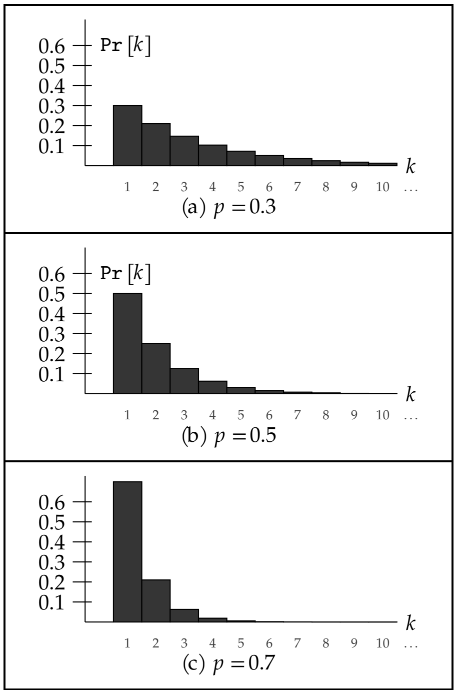
$$\underbrace{\text{HH}\cdots\text{H}}_{k \text{ times}}\ \underbrace{\text{TT}\cdots\text{T}}_{n-k \text{ times}}.$$

The probability of this outcome is $p^k \cdot (1-p)^{n-k}$: the first k flips must come up heads, and the next $n - k$ flips must come up tails. In fact, *any* ordering of k heads and $n - k$ tails has probability $p^k \cdot (1-p)^{n-k}$. One way to see this fact is by imagining the probability tree, which is a binary tree with left branches (heads) having probability p and right branches (tails) having probability $1 - p$. The outcomes in question have k left branches and $n - k$ right branches, and thus have probability $p^k \cdot (1-p)^{n-k}$. There are $\binom{n}{k}$ different outcomes with k heads—a sequence of n flips, out of which we choose which k come up heads. Therefore:

Definition 10.6 (Binomial distribution)
The binomial distribution with parameters n and p *is a distribution over the sample space* $\{0, 1, \ldots, n\}$*, where for each* $k \in \{0, 1, \ldots, n\}$ *we have*

$$\texttt{Pr}\left[k\right] = \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k}.$$

For an unbiased coin, when $p = \frac{1}{2}$, the expression for $\texttt{Pr}\left[k\right]$ from Definition 10.6 simplifies to $\texttt{Pr}\left[k\right] = \binom{n}{k} / 2^n$, because $(\frac{1}{2})^k \cdot (1 - \frac{1}{2})^{n-k} = (\frac{1}{2})^k \cdot (\frac{1}{2})^{n-k} = (\frac{1}{2})^n$.



(a) $p = 0.3$

(b) $p = 0.5$

(c) $p = 0.7$

Figure 10.7: Several geometric distributions, for different values of p. Although these plots are truncated at $k = 10$, the distribution continues infinitely: $\texttt{Pr}\left[k\right] > 0$ for all positive integers k.

Geometric distribution

Another interesting coin-derived distribution comes from the "waiting time" before we see heads for the first time. Consider a p-biased coin, and continue to flip it until we get a heads. The output of this probabilistic process is the number of flips that were required, and the *geometric distribution with parameter p* is defined by this process. (The name "geometric" comes from the fact that the probability of needing k flips looks a lot like a geometric series, from Chapter 5.) See Figure 10.7 for a few such distributions.

What is the probability of needing precisely k flips to get heads for the first time? We would have to have $k - 1$ initial flips come up tails, and then one flip come up heads. As with the binomial distribution, one nice way to think about the probability of this outcome uses the probability tree. This tree has left branches (heads) having probability p and right branches (tails) having probability $1 - p$; the outcome k follows $k - 1$ right branches and one left branch, and thus has probability $(1-p)^{k-1} \cdot p$. Therefore:

Definition 10.7 (Geometric distribution)
Let p *be a real number satisfying* $0 < p \leq 1$*. The* geometric distribution with parameter p *is a distribution over the sample space* $\mathbb{Z}^{\geq 1} = \{1, 2, 3, \ldots\}$*, where for each* k *we have*

$$\texttt{Pr}\left[k\right] = (1-p)^{k-1} \cdot p.$$

Notice that the geometric distribution is our first example of an *infinite* sample space: every positive integer is a possible result.

Computer Science Connections

Quantum Computing

As the 20th-century revolution in physics brought about by the discovery of quantum mechanics unfolded, some researchers working at the boundary of physics and computer science developed a new model of computation based on these quantum ideas. This model of *quantum computation* relies deeply on some very deep physics, far too deep for one page, but here is a brief summary—without any of the details of the physics.

"Anyone who is not shocked by quantum theory has not understood it."
— attributed to Niels Bohr (1885–1962)

The most basic element of data in a quantum computer is a *quantum bit,* or *qubit.* Like a bit (the basic element of data on a *classical* computer), a qubit can be in one of two basic states. These two states are written as $|0\rangle$ and $|1\rangle$. (A classical bit is in state 0 or 1). The quantum magic is that a qubit can *be in both states simultaneously,* in what's called a *superposition* of these basic states. A qubit will be in a state $\alpha|0\rangle + \beta|1\rangle$, where α and β are "weights" where $|\alpha|^2 + |\beta|^2 = 1$. (Actually, the weights α and β are *complex* numbers, but the basic idea will come across if we think of them as real numbers—possibly negative!—instead.) Thus, while there are only two states of a bit, there are infinitely many states that a qubit can be in. So a qubit's state contains a huge amount of information. *But,* by the laws of quantum physics, we are limited in how we can extract that information from a qubit. Specifically, we can *measure* a qubit, but we only see 0 or 1 as the output. When we measure a qubit $\alpha|0\rangle + \beta|1\rangle$, the probability that we see 0 is $|\alpha|^2$; the probability that we see 1 is $|\beta|^2$. For example, we might have a qubit in the state

$$\tfrac{1}{2}|0\rangle + \tfrac{\sqrt{3}}{2}|1\rangle. \qquad \textit{(Note } \left(\tfrac{1}{2}\right)^2 + \left(\tfrac{\sqrt{3}}{2}\right)^2 = \tfrac{1}{4} + \tfrac{3}{4} = 1.)$$

When we measure this qubit, 25% of the time we'd see a 0, and 75% of the time we'd see a 1.

There are two more crucial points. First, when there are multiple qubits—say n of them—the qubits' state is a superposition of 2^n basic states. (For example, two qubits are in a state $\alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle$.) Second, even though we only see one value when we measure qubits, there can be "cancellation" (or *interference*) among coefficients. There are notable restrictions on how we can operate on qubits, based on constraints of physics, but at a very rough level, we can run an operation on an n-qubit quantum computer in parallel in each of the 2^n basic states and, if the process is designed properly, still read something useful from our single measurement.[2]

This cursory description of qubits and quantum computation is nowhere close to a full accounting of how qubits work, or what a quantum computer might do. For much more, see the wonderful text

Why does anyone care about any of this? The main interest in quantum computation stems from a major breakthrough, *Shor's algorithm* (named after its discoverer, Peter Shor): an algorithm that solves the factoring problem—given a large integer n, determine n's prime factorization—efficiently on a quantum computer. An efficient factoring problem is deeply problematic for most currently deployed cryptographic systems (see Chapter 7), so a functional quantum computer would be a big deal. *But,* at least as of this writing, no one has been able to build a quantum computer of any appreciable size. So at the moment, at least, it's a theoretical device—but there's active research both on the physics side (can we actually build one?) and on the algorithmic side (what *else* could we do if we did build one?).

[2] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

Computer Science Connections

Information, Charles Dickens, and the Entropy of English

Consider the following two (identical-length) sequences of letters and spaces—one from Charles Dickens's *A Tale of Two Cities* and one generated by uniformly randomly choosing a sequence of elements of $\{\texttt{A}, \ldots, \texttt{Z}, ␣\}$:

```
IT WAS THE BEST OF TIMES, IT WAS THE WORST OF TIMES, IT WAS THE AGE OF
WISDOM, IT WAS THE AGE OF FOOLISHNESS, IT WAS THE EPOCH OF BELIEF, IT
WAS THE EPOCH OF INCREDULITY.
```

```
TUYSSUWWYVOZULF XZQBSFS AFNBMAOOGWZPAHGREAYC SUSCMBOWDCNCYEJBHPVCRO
MLVTGVHTVCZXHSCQFULCMBO CDIWTXOCUPKTFZVNBHRGDWAKZSZPFTZKEWKWIH O
QFIUWTCDKUBTQSPLXSYXGQZA DLXBHKFILFPZ.
```

Which sequence contains more information? It is very tempting to choose the first (information about contrast, and irony, and the opposition of ideas!)—but, in a precise technical sense, Random contains far more information than Dickens. The basic reason is that, in Dickens, certain letters occur far more frequently than others—E occurs 17 times and there are six letters that don't appear at all. (In Random, all 26 letters appear.) With such a lopsided distribution, you already know a lot about what letter is (probably) going to come next, and so there's less new information conveyed by a typical letter.

Formally, the *entropy* of a sequence of letters (or bits, or whatever) is a measure of "how surprising" each element of the sequence is, averaged over the sequence. We'll convert the "unit of surprise" into a real number between zero and one, where zero corresponds to *the next letter is* 100% *predictable* and one corresponds to *we have absolutely no idea what the next letter will be.* Formally, the entropy H of a probability distribution over S is given by

$$-\sum_{x \in S} \texttt{Pr}\left[x\right] \cdot \log(\texttt{Pr}\left[x\right]).$$

For example, if we produce a sequence of coin flips where each flip comes up heads with probability p (see Figure 10.8), then the entropy of the sequence will be $-\big(p \log p + (1-p)\log(1-p)\big)$, as shown in Figure 10.9.

This definition of entropy comes from the 1940s, in a paper by Claude Shannon,[3] and has found all sorts of useful applications since. Here is one example: the entropy of a sequence of bits expresses a theoretical limit on the *compressibility* of that sequence. (And that theoretical limit is, in fact, achievable.) That is, if the entropy of a string of n bits is very low—say around 0.25—then with some clever algorithms we can represent that string (without any error) using only about $\frac{n}{4}$ bits. But we can't represent it in fewer bits with perfect fidelity ("lossless" compression; see p. 938).

There is significant redundancy in English text, as we've already mentioned, based on the nonuniformity in the probability distribution of individual letters. But there's even more redundancy based on the fact that the probability that the ith character of an English document is an H is affected by whether the $(i-1)$st character was a T. (In the language of Section 10.3, these events are not independent.) If you've seen the letters ␣TH in succession, you can make a very good bet that E is coming next. Compression schemes for English make use of this phenomenon.[4]

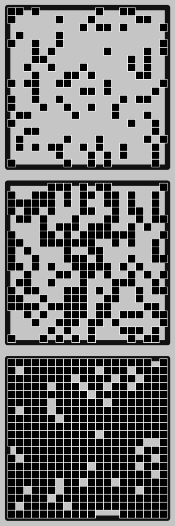


Figure 10.8: A sequence of bits, produced independently at random with probability $p = 0.25$ (top), $p = 0.5$ (middle), and $p = 0.9$ (bottom) of a one. Their entropies are, respectively, 0.8113, 1.0000, and 0.4690.

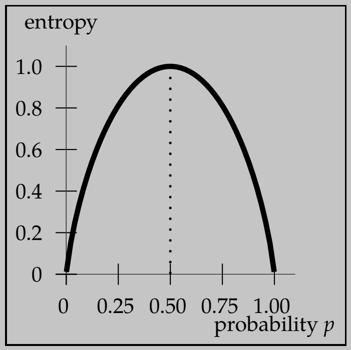


Figure 10.9: The entropy of a biased coin whose heads probability is p.

[3] Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 1948.

For more about entropy, compressibility, and information generally, see a textbook about information theory. A great classic reference is:

[4] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 1991.

10.2.5 Exercises

Philippe flips a fair coin 100 *times. Let the outcome be the number of heads that he sees.*

10.1 What is the sample space?

10.2 What is $\texttt{Pr}\left[0\right]$?

10.3 What is $\texttt{Pr}\left[50\right]$?

10.4 What is $\texttt{Pr}\left[64\right]$?

Philippe now flips his fair coin n times. He is interested in the event "there are (strictly) more heads than tails." What's the probability of this event for the following values of n?

10.5 $n = 2$

10.6 $n = 3$

10.7 $n = 1001$ (*Hint:* $\texttt{Pr}\left[k\right] = \texttt{Pr}\left[1001 - k\right]$.)

10.8 an arbitrary positive integer n

Bridget plays Bridge. Bridge is a card game played with a standard 52*-card deck. Each player is initially dealt a hand of* 13 *cards; assume a fair deal in which each of the* $\binom{52}{13}$ *hands is equally likely.*

10.9 What is the probability of being dealt both A♣ and A♢?

10.10 Suppose Bridget receives a uniformly drawn hand of 13 cards, in a uniformly random order. Because your ex-friend Peter was trying to cheat at poker with this deck, the A♣ card is marked. You observe that the card the fourth-from-the-right position in Bridget's hand is A♣. What is the probability that Bridget also has the A♢ in her hand?

Most casual bridge players sort their hands by suit (♠, ♡, ♣, ♢ *from left to right), and decreasing from left to right by rank within each suit.* (*So one might have a hand like* ♠AK4 ♡983 ♣AKQ ♢AJ98, *reading from left to right.*) *Professional players are taught* not *to sort their hands, because doing so causes which card they play to leak information about the rest of their hand to the other players. Suppose Bridget receives a uniformly drawn hand of* 13 *cards, and* sorts the cards in her hand. *Peter's card marking is still present, and you observe the* A♣ *in a particular position in Bridget's hand. In the following scenarios, what is the probability that Bridget also has the* A♢ *in her hand?* (That is: out of all hands for which A♣ is highest/ lowest/ etc. card, what fraction also have the A♢?)

10.11 A♣ is the fourth-from-the-right (that is, fourth-from-the-lowest) card

10.12 A♣ is the *rightmost* (that is, lowest) card

10.13 A♣ is the *leftmost* (that is, highest) card

Chrissie plays Cribbage. Cribbage is a card game played with a standard 52*-card deck. For the purposes of these questions, assume that a player is dealt one of the* $\binom{52}{4}$ *different* 4*-card hands, chosen uniformly at random. Cribbage hands are awarded points for having a variety of special configurations:*

- *A* flush *is a hand with all four cards from the same suit.*
- *A* run *is a set of at least* 3 *cards with consecutive rank.* (*For example, the hand* 3♡, 9♣, 10♢, J♣ *contains a run.*)
- *A* pair *is a set of two cards with identical rank.*

Aces are low in Cribbage, so A, 2, 3 *is a valid run, but* Q, K, A *is not.*

10.14 What's the probability that Chrissie is dealt a flush?

10.15 What's the probability that Chrissie is dealt a run of length 4?

10.16 What's the probability of getting *two* runs of length 3 that is not a run of 4? (For example, the hand 9♡, 9♣, 10♢, J♣ contains two runs of length 3: the first is 9♡, 10♢, J♣ and the second is 9♣, 10♢, J♣.)

10.17 What's the probability of getting *one* (and only one) run of length 3 (and not a run of length 4)?

10.18 What's the probability of getting at least one pair? (*Hint:* $\texttt{Pr}\left[\textit{getting a pair}\right] = 1 - \texttt{Pr}\left[\textit{getting no pair}\right]$.)

10.19 What's the probability of getting two or more pairs? (In cribbage, any two cards with the same rank count as a pair; for example, the hand 2♡2♢2♠8♣ has *three* pairs: 2♡2♢ and 2♡2♠ and 2♢2♠.)

10.20 (*programming required*) Write a program to approximately verify your calculations from these Cribbage exercises, as follows: generate 1,000,000 random hands from a standard deck, and count the number of those samples in which there's a flush, run (of the three flavors), pair, or multiple pairs.

10.21 (*programming required*) Modify your program to exactly verify your calculations: exhaustively generate *all* 4-card hands, and count the number of hands with the various features (flushes, runs, pairs).

10.22 A *fifteen* is a subset of cards whose ranks sum to 15, where an A counts as 1 and each of {10, J, Q, K} counts as 10. (For example, the hand 3♡, 2♣, 5♢, J♣ contains two fifteens: 3♡ + 2♣ + J♣ = 15 and 5♢ + J♣ = 15.) What's the probability a 4-card hand contains at least one fifteen? (*Hint: use a program.*)

10.23 A bitstring $x \in \{0,1\}^5$ is stored in vulnerable memory, subject to corruption—for example, on a spacecraft. An α-ray strikes the memory and resets one bit to a random value (both the new value and which bit is affected are chosen uniformly at random). A second α-ray strikes the memory and resets one bit (again chosen uniformly at random). What's the probability that the resulting bitstring is identical to x?

```
quickSort($A[1 \ldots n]$):
1: if $n \leq 1$ then
2:     return $A$
3: else
4:     choose $pivot \in \{1, \ldots, n\}$, somehow.
5:     $L$ := list of all $A[i]$ where $A[i] < A[pivot]$.
6:     $R$ := list of all $A[i]$ where $A[i] > A[pivot]$.
7:     return quickSort($L$) + $\langle A[pivot] \rangle$ + quickSort($R$)
```

Figure 10.10: Quick Sort, briefly. (See Figure 5.20(a) for more detail.) Assume that the elements of A are all distinct.

Recall the quick sort *algorithm for sorting an array A: we choose a "pivot" value x; we partition A into those elements less than x and those greater than x; and we return x and those two sublists, recursively sorted, in the correct order. (See Figure 10.10.) This algorithm is efficient if the two sublists are close to equal in size. There are many ways to choose the pivot value, but one common (and good!) strategy is to choose x* randomly *from A.*

Assume that the elements of A are all distinct. If we select pivot *in Line 4 by choosing* uniformly at random *from the set $\{1, \ldots, n\}$:*

10.24 As a function of n, what is the probability that $|L| \leq 3n/4$ *and* $|R| \leq 3n/4$? (You may assume that n is divisible by 4.)

10.25 As a function of n and $\alpha \in [0, 1]$, what is the probability $|L| \leq \alpha n$ *and* $|R| \leq \alpha n$? (You may neglect issues of integrality: assume αn is an integer.)

Suppose that we choose pivot *in Line 4 by choosing* three elements p_1, p_2, p_3 uniformly at random *from the set $\{1, \ldots, n\}$, and taking as* pivot *the p_i whose corresponding element of A is the median of the three. (Assume that the same index can be chosen as both p_1 and p_3, for example.) For example, for the array $A = \langle 94, 32, 29, 85, 64, 8, 12, 99 \rangle$, we might randomly choose $p_1 = 1$, $p_2 = 7$, and $p_3 = 2$. Then the pivot will be p_3 because $A[p_3] = 32$ is between $A[p_2] = 12$ and $A[p_1] = 94$. Under this "median of three" strategy:*

10.26 What is the probability that $|L| \leq 3n/4$ *and* $|R| \leq 3n/4$? Assume n is large; for ease, you may neglect issues of integrality in your answer.

10.27 As a function of $\alpha \in [0, 1]$, what is the probability $|L| \leq \alpha n$ *and* $|R| \leq \alpha n$? Again, you may assume that n is large, and you may neglect issues of integrality in your answer.

Suppose that Team Emacs and Team VI play a best-of-five series of softball games. Emacs, being better than VI, wins each game with probability 60%.

10.28 Use a tree diagram to compute the probability that Team Emacs wins the series.

10.29 What is the probability that the series goes five games? (That is, what is the probability that neither team wins 3 of the first 4 games?)

10.30 Update your last two answers if Team Emacs wins each game with probability 70%.

"Emacs" rhymes with "ski wax"; "VI" rhymes with "knee-high." The teams are named after two text editors frequently used by computer scientists to write programs or emails or textbooks.

(Calculus required.) *Now assume that Team Emacs wins each game with probability p, for an arbitrary value $p \in [0, 1]$. For the following questions, write down a formula expressing the probability of the listed event. Also find the value of p that maximizes the probability, and the probability of the specified event for this maximizing p.*

10.31 There is a fifth game in the series.

10.32 There is a fourth game of the series.

10.33 There is a fourth game of the series *and* Team Emacs wins that fourth game.

Let S be a sample space, and let $\texttt{Pr} : S \to [0, 1]$ be an arbitrary function satisfying the requirements of being a probability function (Definition 10.2). That is, we have

$$\sum_{s \in S} \texttt{Pr}[s] = 1 \qquad \textit{and} \qquad \texttt{Pr}[s] \geq 0 \textit{ for all } s \in S.$$

Argue briefly that the following properties hold.

10.34 For any outcome $s \in S$, we have $\texttt{Pr}[s] \leq 1$.

10.35 For any event $A \subseteq S$, we have $\texttt{Pr}\left[\overline{A}\right] = 1 - \texttt{Pr}[A]$. (Recall that $\overline{A} = S - A$.)

10.36 For any events $A, B \subseteq S$, we have $\texttt{Pr}[A \cup B] = \texttt{Pr}[A] + \texttt{Pr}[B] - \texttt{Pr}[A \cap B]$.

10.37 The *Union Bound:* for any events $A_1, A_2, \ldots, A_n$, we have $\texttt{Pr}[\bigcup_i A_i] \leq \sum_i \texttt{Pr}[A_i]$.

Imagine n identical computers that share a single radio frequency for use as a network connection. Each of the n computers would like to send a packet of information out across the network, but if two or more different computers simultaneously try to send a message, no message gets through. Here you'll explore another use of randomization: using randomness for symmetry breaking.

10.38 Suppose that each computer flips a coin that comes up heads with probability p. What is the probability that *exactly* one of the n machines' coins comes up heads (and thus that machine can send its message)? Your answer should be a formula that's in terms of n and p.

(The next two exercises require calculus.)

10.39 Given the formula you found in Exercise 10.38, what p should you choose to maximize the probability of a message being successfully sent?

10.40 What is the probability of success if you choose p as in Exercise 10.39? What is the limit of this quantity as n grows large? *(You may use the following fact: $(1 - \frac{1}{m})^m \to e^{-1}$ as $m \to \infty$.)*

We hash items into a 10*-slot hash table using a hash function h that uniformly assigns elements to* $\{1, \ldots, 10\}$*. Compute the probability of the following events if we hash* 3 *elements into the* 10*-slot table:*

10.41 no collisions occur

10.42 all 3 elements have the same hash value

Suppose that we resolve collisions by linear probing*, wherein an element x that hashes to an occupied cell $h(x)$ is placed in the first unoccupied cell after $h(x)$. (That is, we try to put x into $h(x)$, then $h(x)+1$, then $h(x)+2$, and so forth—wrapping back around to the beginning of the table after the* 10*th slot. See Figure 10.11.) If we hash* 3 *elements into the* 10*-slot table, what is the probability that ...*

10.43 at least 2 adjacent slots are filled. (Count slot #10 as adjacent to #1.)

10.44 3 adjacent slots are filled.

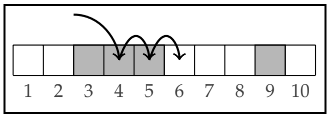


Figure 10.11: A reminder of linear probing. If $h(x) = 4$, then we try to store x in slot 4, then 5, then 6. Because slot 6 is empty, x is placed into that slot.

One issue with resolving collisions by linear probing is called clustering*: if there's a large block of occupied slots in the hash table, then there's a relatively high chance that the next element placed into the table extends that block.*

10.45 Suppose that we currently have a single block of k adjacent slots full in an n-slot hash table, and all other slots are empty. What's the probability that the next element inserted into the hash table extends that block (that is, leaves $k+1$ adjacent slots full).

10.46 (*programming required*) Write a program to hash 5000 elements into a 10,007-slot hash table using linear probing. Record which cell x_{5000} ends up occupying—that is, how many hops from $h(x_{5000})$ is x_{5000}? Run your program 2048 times, and report how far, on average, x_{5000} moved from $h(x_{5000})$. Also report the *maximum* distance that x_{5000} moved.

Because linear probing suffers from this clustering issue, other mechanisms for resolving collisions are sometimes used. Another choice is called quadratic probing*: we change the cell number we try by an increasing step size at every stage, instead of by one every time. Specifically, to hash x into an n-slot table, first try to store x in $h(x)$; if that cell is full, try putting x into $h(x)+i^2$, wrapping back around to the beginning of the table as usual, for $i = 1, 2, \ldots$. (Linear probing tried slot $h(x)+i$ instead.)*

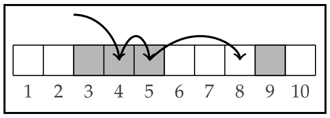


Figure 10.12: Quadratic probing. We try to store x in slot $h(x)$, then $h(x)+1^2$, then $h(x)+2^2$, etc.

10.47 (*programming required*) Modify your program from Exercise 10.46 to use quadratic probing instead, and report the same statistics: the mean and maximum number of cells probed for x_{5000}.

10.48 In about one paragraph, explain the differences that you observed between linear and quadratic probing. A concern called *secondary clustering* arises in quadratic probing: if $h(x) = h(y)$ for two elements x and y, then the sequence of cells probed for x and y is identical. These sequences were also identical for linear probing. In your answer, explain why secondary clustering from quadratic probing is less of a concern than the clustering from linear probing.

A fourth way of handling collisions in hash tables (after chaining, linear probing, and quadratic probing) is what's called double hashing*: we move forward by the same number of slots at every stage, but that number is randomly chosen, as the output of a different hash function. Specifically, to hash x into an n-slot table, first try to store x in $h(x)$; if that cell is full, try putting x into $h(x) + i \cdot g(x)$, wrapping back around to the beginning of the table as usual, for $i = 1, 2, \ldots$. (Here g is a* different *hash function, crucially one whose output is never zero.) See Figure 10.13.*

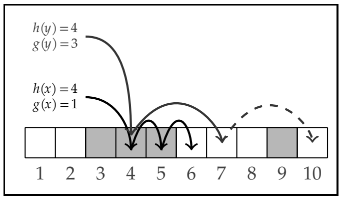


Figure 10.13: Double hashing. We try to store x in slot $h(x)$, then $h(x)+g(x)$, then $h(x)+2g(x)$, etc. (wrapping around the table as necessary).

10.49 (*programming required*) Modify your program from Exercises 10.46 and 10.47 to use double hashing. Again report the mean and maximum number of cells probed for x_{5000}.

10.50 In about one paragraph, explain the differences you observe between chaining, linear probing, quadratic probing, and double hashing. Is there any reason you wouldn't always use double hashing?

Consider a randomized algorithm that solves a problem on a particular input correctly with probability p, and it's wrong with probability $1-p$. Assume that each run of the algorithm is independent of every other run, so that we can think of each run as being an (independent) coin flip of a p-biased coin (where heads means "correct answer").

10.51 (*Requires calculus.*) Suppose that the probability p is unknown to you. You observe that exactly k out of n trials gave the correct answer. Then the number k of correct answers follows a binomial distribution with parameters n and p: that is, the probability that exactly k runs give the correct answer is

$$\binom{n}{k} \cdot p^k \cdot (1-p)^{n-k}. \qquad (*)$$

Prove that the *maximum likelihood estimate* of p is $p = \frac{k}{n}$—that is, prove that $(*)$ is maximized by $p = \frac{k}{n}$.

10.52 (*Requires calculus.*) Suppose that the probability p is unknown to you. You observe that it takes n trials before the first time you get a correct answer. Then n follows a geometric distribution with parameter p: that is, the probability that n runs were required is given by

$$(1-p)^{n-1}p. \qquad (\dagger)$$

Prove that the maximum likelihood estimate of p is $p = \frac{1}{n}$—that is, prove that $(\dagger)$ is maximized by $p = \frac{1}{n}$.

10.3 Independence and Conditional Probability

> If your parents never had children, chances are you won't, either.
>
> Dick Cavett (b. 1936)

Imagine that you're interviewing to be a consultant for Premier Passenger Pigeon Purveyors, a company that pitches its products to prospective pigeon purchasers using online advertising—specifically, by displaying ads to users of a particular search engine on the web. PPPP makes \$50 profit from each sale, and, from historical data, they have determined that 0.02% of searchers who see an ad buy a pigeon. The interviewer asks you how much PPPP should be willing to pay to advertise to a searcher. A good answer is \$0.01: on average, PPPP earns $\$50 \cdot 0.0002 = \$0.01$ per ad, so paying anything up to a penny per ad yields a profit, on average. But you realize that there's a better answer (and, by giving it, you get the job): *it depends on what the user is searching for!* A user who searches for `BIRD` or `PIGEON` or `BUYING A PET TO COMBAT LONELINESS` is far more likely to respond to a PPPP ad than an average user, while a user who searches for `ORNITHOPHOBIA` is much less likely to respond to an ad.

It is a general phenomenon in probability that *knowing that event A has occurred* may tell you that *an event B is much more likely* (*or much less likely*) *to occur* than you'd previously known. In this section, we'll discuss when knowing that an event A has occurred does or does not affect the probability that B occurs (that is, whether A and B are *dependent* or *independent*, respectively). We'll then introduce *conditional probability*, which allows us to state and manipulate quantities like "the probability that B happens *given that A happens*."

10.3.1 Independence and Dependence of Events

We'll start with *independence* and *dependence* of events. Intuitively, two events A and B are dependent if A's occurrence/nonoccurrence gives us some information about whether B occurs; in contrast, A and B are independent when A occurs with the same probability when B occurs as it does when B does not occur. More formally:

Definition 10.8 (Independent and dependent events)
Two events A and B are independent *if and only if* $\texttt{Pr}[A \cap B] = \texttt{Pr}[A] \cdot \texttt{Pr}[B]$*. The events A and B are called* dependent *if they are not independent.*

If A and B are dependent events, then we can also say that A and B are *correlated*; independent events are said to be *uncorrelated.*

This definition is phrased a bit differently from the intuition above, but a little manipulation of the equation from Definition 10.8 may help to make the connection clearer. Assume for the moment that $\texttt{Pr}[B] \neq 0$. (Exercise 10.70 addresses the case of $\texttt{Pr}[B] = 0$.) Dividing both sides of the equality $\texttt{Pr}[A] \cdot \texttt{Pr}[B] = \texttt{Pr}[A \cap B]$ by $\texttt{Pr}[B]$, we see that the events A and B are independent if and only if

$$\texttt{Pr}[A] = \frac{\texttt{Pr}[A \cap B]}{\texttt{Pr}[B]}.$$

The left-hand side ($\texttt{Pr}\left[A\right]$) denotes the fraction of the time that A occurs. The right-hand side ($\texttt{Pr}\left[A \cap B\right] / \texttt{Pr}\left[B\right]$) denotes the fraction of the time *when B occurs* that A occurs too. If these two fractions are equal, then A occurs with the same probability when B occurs as it does when B does not occur. (And if these two fractions are equal, then *both* when B occurs and when B does not occur, A occurs with probability $\texttt{Pr}\left[A\right]$—that is, the probability of A without reference to B.)

Examples of independent and dependent events

To establish that two events A and B are independent, we can simply compute $\texttt{Pr}\left[A\right]$, $\texttt{Pr}\left[B\right]$, and $\texttt{Pr}\left[A \cap B\right]$, and show that the product of the first two quantities is equal to the third. Here are a few examples:

Example 10.14 (Some independent events)
The following pairs of events are independent:

1. I flip a fair penny and a fair nickel. Define the following events:
   - *Event A:* The penny is heads.
   - *Event B:* The nickel is heads.

   Then $\texttt{Pr}\left[A\right] = 0.5$ and $\texttt{Pr}\left[B\right] = 0.5$ and $\texttt{Pr}\left[A \cap B\right] = 0.25 = 0.5 \cdot 0.5$.

2. I draw a card from a randomly shuffled deck. Define the following events:
   - *Event A:* I draw an ace.
   - *Event B:* I draw a heart.

   For these events, we have
$$\begin{aligned} \texttt{Pr}\left[A\right] &= \texttt{Pr}\left[\{\mathrm{A}\clubsuit, \mathrm{A}\diamondsuit, \mathrm{A}\heartsuit, \mathrm{A}\spadesuit\}\right] &&= \tfrac{1}{13} \\ \texttt{Pr}\left[B\right] &= \texttt{Pr}\left[\{\mathrm{A}\heartsuit, 2\heartsuit, \ldots, \mathrm{K}\heartsuit\}\right] &&= \tfrac{1}{4} \\ \texttt{Pr}\left[A \cap B\right] &= \texttt{Pr}\left[\{\mathrm{A}\heartsuit\}\right] &&= \tfrac{1}{52} = \tfrac{1}{4} \cdot \tfrac{1}{13}. \end{aligned}$$

3. I roll a fair red die and a fair blue die. Define the following events:
   - *Event A:* The red die is odd.
   - *Event B:* The sum of the rolled numbers is odd.

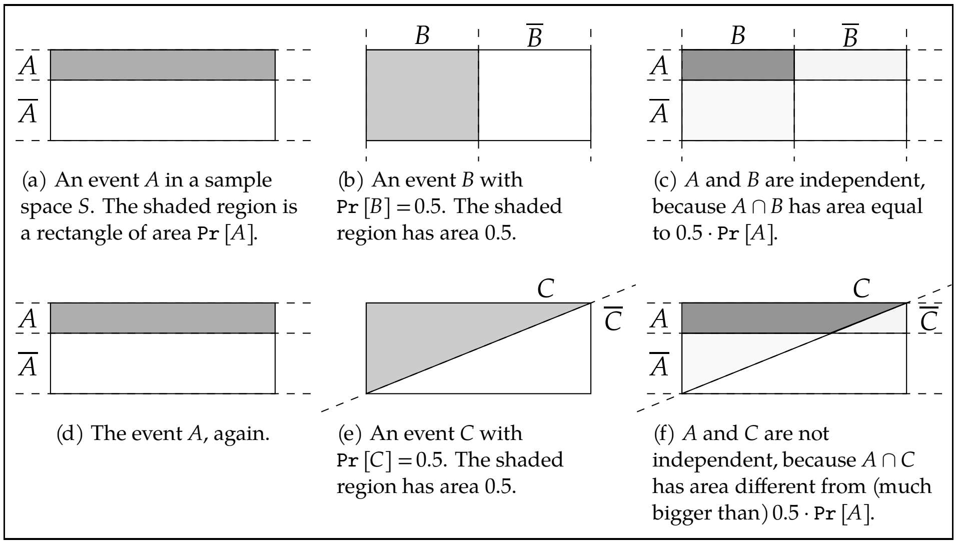
   Then, writing outcomes as $\langle$the red roll, the blue roll$\rangle$, we have
$$\begin{aligned} \texttt{Pr}\left[A\right] &= \texttt{Pr}\left[\{1,3,5\} \times \{1,2,3,4,5,6\}\right] &&= \tfrac{18}{36} = 0.5 \\ \texttt{Pr}\left[B\right] &= \texttt{Pr}\left[\underbrace{\{1,3,5\} \times \{2,4,6\}}_{\text{red odd, blue even}} \cup \underbrace{\{2,4,6\} \times \{1,3,5\}}_{\text{red even, blue odd}}\right] &&= \tfrac{18}{36} = 0.5 \end{aligned}$$

   Observe that $A \cap B = \{1,3,5\} \times \{2,4,6\}$, and so $\texttt{Pr}\left[A \cap B\right] = \frac{9}{36} = (0.5) \cdot (0.5)$.

Any time the processes by which A and B come to happen are completely unrelated, it's certainly true that A and B are independent. But events can also be independent in other circumstances, as we saw in Example 10.14.3: both events in this example in some way incorporated the result of the value rolled on the red die, but the stated events themselves are independent anyway.

A visual representation of independent and dependent events is shown in Figure 10.14.

In Example 10.14, we showed that a few pairs of events are independent by showing that $\texttt{Pr}[A \cap B] = \texttt{Pr}[A] \cdot \texttt{Pr}[B]$. By contrast, we can establish that two events are not independent—that is, they are *dependent*—directly from the definition by showing that $\texttt{Pr}[A \cap B] \neq \texttt{Pr}[A] \cdot \texttt{Pr}[B]$. Here are a few examples:



(a) An event A in a sample space S. The shaded region is a rectangle of area $\texttt{Pr}[A]$.

(b) An event B with $\texttt{Pr}[B] = 0.5$. The shaded region has area 0.5.

(c) A and B are independent, because $A \cap B$ has area equal to $0.5 \cdot \texttt{Pr}[A]$.

(d) The event A, again.

(e) An event C with $\texttt{Pr}[C] = 0.5$. The shaded region has area 0.5.

(f) A and C are not independent, because $A \cap C$ has area different from (much bigger than) $0.5 \cdot \texttt{Pr}[A]$.

Figure 10.14: Pairs of independent and dependent events, represented visually. Think of the area of a region as its probability; the area of the sample space (the enclosing rectangle) is 1.

Example 10.15 (Some dependent events)

The following pairs of events are *not* independent:

1. I roll a fair die. Define the following events:

   - *Event A:* I roll an odd number.
   - *Event B:* I roll a prime number.

   $$\begin{array}{rclcl} \texttt{Pr}[A] & = & \texttt{Pr}\left[\{1,3,5\}\right] & = & \frac{1}{2} \\ \texttt{Pr}[B] & = & \texttt{Pr}\left[\{2,3,5\}\right] & = & \frac{1}{2} \\ \texttt{Pr}[A \cap B] & = & \texttt{Pr}\left[\{3,5\}\right] & = & \frac{2}{6}. \end{array}$$

   Because the probability of $A \cap B$ is $\frac{2}{6} = \frac{1}{3}$, but $\texttt{Pr}[A] \cdot \texttt{Pr}[B] = \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \neq \frac{1}{3}$, the events A and B are dependent.

   Similarly, define *Event C* as "I roll an even number." Because $\texttt{Pr}[B] = \texttt{Pr}[C] = \frac{1}{2}$ and $\texttt{Pr}[C \cap B] = \texttt{Pr}\left[\{2\}\right] = \frac{1}{6} \neq \frac{1}{2} \cdot \frac{1}{2}$, the events B and C are dependent too.

2. I draw a card from a randomly shuffled deck. Define the following events:

   - *Event A:* I draw a heart.
   - *Event B:* I draw a spade.

   $$\begin{array}{rclcl} \texttt{Pr}[A] & = & \texttt{Pr}\left[\{\text{A}\heartsuit, 2\heartsuit, \ldots, \text{K}\heartsuit\}\right] & = & \frac{1}{4} \\ \texttt{Pr}[B] & = & \texttt{Pr}\left[\{\text{A}\spadesuit, 2\spadesuit, \ldots, \text{K}\spadesuit\}\right] & = & \frac{1}{4} \\ \texttt{Pr}[A \cap B] & = & \texttt{Pr}[\varnothing] & = & 0, \end{array}$$

   where $A \cap B = \varnothing$ because no cards are both a heart *and* a spade. Because $0 \neq \frac{1}{16} = \frac{1}{4} \cdot \frac{1}{4}$, we have $\texttt{Pr}[A \cap B] \neq \texttt{Pr}[A] \cdot \texttt{Pr}[B]$. These events are dependent.

3. I flip a fair penny and a fair nickel. Define the following events:

   - *Event A:* The penny is heads.
   - *Event B:* Both coins are heads.

   Then $\texttt{Pr}[A] = 0.5$ and $\texttt{Pr}[B] = 0.25$ and $\texttt{Pr}[A \cap B] = 0.25 = \texttt{Pr}[B] \neq \texttt{Pr}[A] \cdot \texttt{Pr}[B]$.

Correlation of events

The pairs of dependent events from Example 10.15 are of two different qualitative types. Knowing that the first event occurred can make the second event more likely to occur ("rolling an odd number" and "rolling a prime number" for the dice) or less likely to occur ("rolling an even number" and "rolling a prime number"):

Definition 10.9 (Positive and negative correlation)
When two events A and B satisfy $\texttt{Pr}[A \cap B] > \texttt{Pr}[A] \cdot \texttt{Pr}[B]$, we say that A and B are positively correlated. *When $\texttt{Pr}[A \cap B] < \texttt{Pr}[A] \cdot \texttt{Pr}[B]$, we say that A and B are* negatively correlated. (*If $\texttt{Pr}[A \cap B] = \texttt{Pr}[A] \cdot \texttt{Pr}[B]$, then A and B are* uncorrelated.)

At the extreme, knowing that the first event occurred can ensure that the second event definitely does not occur ("drawing a heart" and "drawing a spade" from Example 10.15) or can ensure that the second event definitely does occur ("both coins are heads" and "the first coin is heads" from Example 10.15).

Here are some further examples in which you're asked to figure out whether certain pairs of events are independent or dependent:

Example 10.16 (Encryption by random substitution)

Problem: One simple form of encryption for text is a *substitution cipher*, in which (in the simplest version) we choose a permutation of the alphabet, and then replace each letter with its permuted variant. (For example, we might permute the letters as $\texttt{ABCDE}\cdots \to \texttt{XENBG}\cdots$; thus `DECADE` would be written as `BGNXBG`.) Suppose we choose a random permutation for this mapping, so that each of the 26! orderings of the alphabet is equally likely. Are the following events Q and Z independent or dependent?

- $Q =$ "the letter `Q` is mapped to itself (that is, `Q` is 'rewritten' as `Q`)."
- $Z =$ "the letter `Z` is mapped to itself."

Solution: We must compute $\texttt{Pr}[Q]$, $\texttt{Pr}[Z]$, and $\texttt{Pr}[Q \cap Z]$. Because each permutation is equally likely to be chosen, we have

$$\texttt{Pr}[Q] = \frac{\text{\# permutations } \pi_{1,2,\ldots,26} \text{ where } \pi_{17} = 17}{\text{\# permutations } \pi_{1,2,\ldots,26}} = \frac{25!}{26!} = \frac{1}{26}$$

because we can choose any of 25! orderings of all non-`Q` letters. Similarly,

$$\texttt{Pr}[Z] = \frac{\text{\# permutations } \pi_{1,2,\ldots,26} \text{ where } \pi_{26} = 26}{\text{\# permutations } \pi_{1,2,\ldots,26}} = \frac{25!}{26!} = \frac{1}{26}.$$

To compute $\texttt{Pr}\left[Q \cap Z\right]$, we need to count the number of permutations $\pi_{1\ldots26}$ with both $\pi_{17} = 17$ and $\pi_{26} = 26$. Any of the 24 other letters can go into any of the remaining 24 slots of the permutation, so there are 24! such permutations. Thus

$$\texttt{Pr}\left[Q \cap Z\right] = \frac{\#\text{ permutations } \pi_{1,2,\ldots,26} \text{ where } \pi_{17} = 17 \text{ and } \pi_{26} = 26}{\#\text{ permutations } \pi_{1,2,\ldots,26}} = \frac{24!}{26!} = \frac{1}{25 \cdot 26}.$$

Thus we have

$$\texttt{Pr}\left[Q \cap Z\right] = \tfrac{1}{25 \cdot 26} \qquad \text{and} \qquad \texttt{Pr}\left[Q\right] \cdot \texttt{Pr}\left[Z\right] = \tfrac{1}{26} \cdot \tfrac{1}{26} = \tfrac{1}{26 \cdot 26}.$$

There's only a small difference between $\frac{1}{26 \cdot 26} \approx 0.00148$ and $\frac{1}{25 \cdot 26} \approx 0.00154$, but they're indubitably different, and thus Q and Z are *not independent.*

(Incidentally, substitution ciphers are susceptible to *frequency analysis*: the most common letters in English-language texts are `ETAOIN`—almost universally in texts of reasonable length—and the frequencies of various letters is surprisingly consistent. See Exercises 10.72–10.76.)

Example 10.17 (Matched flips of two fair coins)

Problem: I flip two fair coins (independently). Consider the following events:

- *Event A:* the first flip comes up heads.
- *Event B:* the second flip comes up heads.
- *Event C:* the two flips match (are both heads or are both tails).

Which pairs of these events are independent, if any?

Solution: The sample space is {HH, HT, TH, TT}, and the events from the problem statement are given by $A = \{\text{HH}, \text{HT}\}$, $B = \{\text{HH}, \text{TH}\}$, and $C = \{\text{HH}, \text{TT}\}$. Thus $A \cap B = A \cap C = B \cap C = \{\text{HH}\}$—that is, HH is the only outcome that results in more than one of these events being true. (See Figure 10.15.)

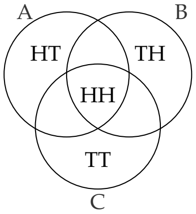


Figure 10.15: Two coin flips and three events.

Because the coins are fair, every outcome in this sample space has probability $\frac{1}{4}$. Focusing on the events A and B, we have

$$\begin{aligned}
\texttt{Pr}\left[A\right] &= \texttt{Pr}\left[\{\text{HH}, \text{HT}\}\right] &= \tfrac{1}{2} \\
\texttt{Pr}\left[B\right] &= \texttt{Pr}\left[\{\text{HH}, \text{TH}\}\right] &= \tfrac{1}{2} \\
\texttt{Pr}\left[A \cap B\right] &= \texttt{Pr}\left[\{\text{HH}\}\right] &= \tfrac{1}{4}.
\end{aligned}$$

Thus $\texttt{Pr}\left[A\right] \cdot \texttt{Pr}\left[B\right] = \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4}$, and $\texttt{Pr}\left[A \cap B\right] = \frac{1}{4}$. Because $\texttt{Pr}\left[A\right] \cdot \texttt{Pr}\left[B\right] = \texttt{Pr}\left[A \cap B\right]$, the two events are independent.

The calculation is identical for the other two pairs of events, and so A and B are independent; A and C are independent; and B and C are independent.

Example 10.18 (Matched flips of two biased coins)

Problem: How would your answers to Example 10.17 change if the coins are p-biased instead of fair?

Solution: The sample space and events remain as in Example 10.17 (see Figure 10.16), but the outcomes now have different probabilities:

outcome	HH	HT	TH	TT
probability	$p \cdot p$	$p \cdot (1-p)$	$(1-p) \cdot p$	$(1-p) \cdot (1-p)$

Using these outcome probabilities, we compute the event probabilities as follows:

$$\begin{aligned}
\Pr[A] &= \Pr[\{\mathrm{HH}, \mathrm{HT}\}] &&= p \cdot p + p \cdot (1-p) &&= p && (1)\\
\Pr[B] &= \Pr[\{\mathrm{HH}, \mathrm{TH}\}] &&= p \cdot p + (1-p) \cdot p &&= p && (2)\\
\Pr[C] &= \Pr[\{\mathrm{HH}, \mathrm{TT}\}] &&= p \cdot p + (1-p) \cdot (1-p) &&= p^2 + (1-p)^2. && (3)
\end{aligned}$$

Because $A \cap B = B \cap C = A \cap C = \{\mathrm{HH}\}$, we also have

$$\Pr[A \cap B] = \Pr[B \cap C] = \Pr[A \cap C] = \Pr[\mathrm{HH}] = p^2. \qquad (4)$$

Thus A and B are still independent, because $\Pr[A] \cdot \Pr[B] = p \cdot p = p^2 = \Pr[A \cap B]$ by (1), (2), and (4). But what about the events A and C? By (1), (3), and (4), we have

$$\Pr[A] \cdot \Pr[C] = p \cdot \left[p^2 + (1-p)^2\right] \qquad \text{and} \qquad \Pr[A \cap C] = p^2.$$

By a bit of algebra, we see that $\Pr[A \cap C] = \Pr[A] \cdot \Pr[C]$ if and only if

$$\begin{aligned}
p^2 = p(p^2 + (1-p)^2) &\Leftrightarrow 0 = p(p^2 + (1-p)^2) - p^2\\
&\Leftrightarrow 0 = 2p^3 - 3p^2 + p\\
&\Leftrightarrow 0 = p(2p-1)(p-1).
\end{aligned}$$

So the events A and C are independent—that is, $\Pr[A \cap C] = \Pr[A] \cdot \Pr[C]$—if and only if $p \in \{0, \frac{1}{2}, 1\}$.

Thus events A and B are independent for any value of p, while events A and C (and similarly B and C) are independent if and only if $p \in \{0, \frac{1}{2}, 1\}$.

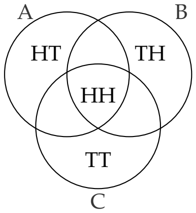


Figure 10.16: The flips and events, again. Recall the events: A: 1st flip heads. B: 2nd flip heads. C: flips match.

Taking it further: While *any two of the events* from Example 10.17 (or Example 10.18 with $p = \frac{1}{2}$) are independent, *the third event is not independent of the other two.* Another way to describe this situation is that the events A and $B \cap C$ are *not* independent: in particular, $\Pr[A \cap (B \cap C)] / \Pr[B \cap C] = 1 \neq \Pr[A]$. A set of events $A_1, A_2, \ldots, A_n$ is said to be *pairwise independent* if, for any two indices i and $j \neq i$, the events A_i and A_j are independent. More generally, these events are said to be *k-wise independent* if, for any subset S of up to k of these events, the events in S are all independent. (And we say that the set of events is *fully independent* if every subset of any size satisfies this property.)

Sometimes it will turn out that we "really" care only about pairwise independence. For example, if we think about a hash table that uses a "random" hash function, we're usually only concerned with the question "do elements x and y collide?"—which is a question about just one pair of events. Generally, we can create a pairwise-independent random hash function more cheaply than creating a fully independent random hash function. If we view random bits as a scarce resource (like time and space, in the style of Chapter 6), then this savings is valuable.

10.3.2 Conditional Probability

In Section 10.3.1, we discussed the black-and-white distinction between pairs of independent events and dependent events: if A and B are independent, then knowing whether or not B happened gives you no information about whether A happened; if A and B are dependent, then the probability that A happens if B happened is different from the probability that A happens if B did not happen. But *how* does knowing that B occurred change your estimate of the probability of A? Think about events like "the sky is clear" and "it is very windy" and "it will rain today": sometimes B means that A is less likely or even impossible; sometimes B means that A is more likely or even certain. Here we will discuss *quantitatively* how one event's probability is affected by the knowledge of another event.

The *conditional probability of A given B* represents the probability of A occurring *if we know that B occurred*:

Definition 10.10 (Conditional probability)
The conditional probability of A given B, *written* $\texttt{Pr}\left[A|B\right]$, *is given by*

$$\texttt{Pr}\left[A|B\right] = \frac{\texttt{Pr}\left[A \cap B\right]}{\texttt{Pr}\left[B\right]}.$$

(The quantity $\texttt{Pr}\left[A|B\right]$ *is also sometimes called the* probability of A conditioned on B.*)* *We will treat* $\texttt{Pr}\left[A|B\right]$ *as undefined when* $\texttt{Pr}\left[B\right] = 0$.

Here are a few simple examples:

Example 10.19 (Odds and primes)
I choose a number uniformly at random from $\{1, 2, \ldots, 10\}$. Define these two events:

- *Event A:* The chosen number is odd.
- *Event B:* The chosen number is prime.

For these events, we have
$$\texttt{Pr}\left[A|B\right] = \frac{\texttt{Pr}\left[A \cap B\right]}{\texttt{Pr}\left[B\right]} = \frac{\texttt{Pr}\left[\{3,5,7\}\right]}{\texttt{Pr}\left[\{2,3,5,7\}\right]} = \frac{3}{4}$$
and
$$\texttt{Pr}\left[B|A\right] = \frac{\texttt{Pr}\left[A \cap B\right]}{\texttt{Pr}\left[A\right]} = \frac{\texttt{Pr}\left[\{3,5,7\}\right]}{\texttt{Pr}\left[\{1,3,5,7,9\}\right]} = \frac{3}{5}.$$

Example 10.20 (Dominoes)
Problem: Shuffle the dominoes in Figure 10.17, and draw one uniformly at random.

1. What is the probability that you drew a domino with a 2 (⚁) on it?
2. You make a draw and see the domino ⚀▨. (Imagine the shaded side of the domino is covered by your hand.) What's the probability your domino has a 2?
3. You make a draw and see that the domino is ⚄▨. What is the probability that you drew a domino with a 2?

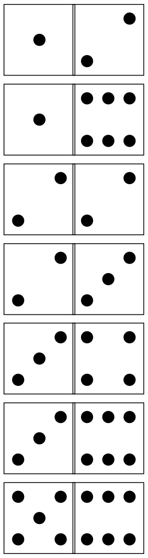
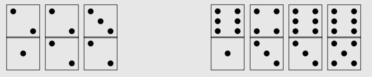


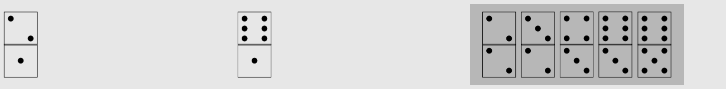
Figure 10.17: Some dominoes.

Solution: 1. We are asked for the probability of drawing a domino with a 2:



Thus 3 of the 7 dominoes have a 2, so $\texttt{Pr}[\text{⚁}] = \frac{3}{7}$.

2. We observe ⚀ on our drawn domino. We're asked for the probability of a 2:



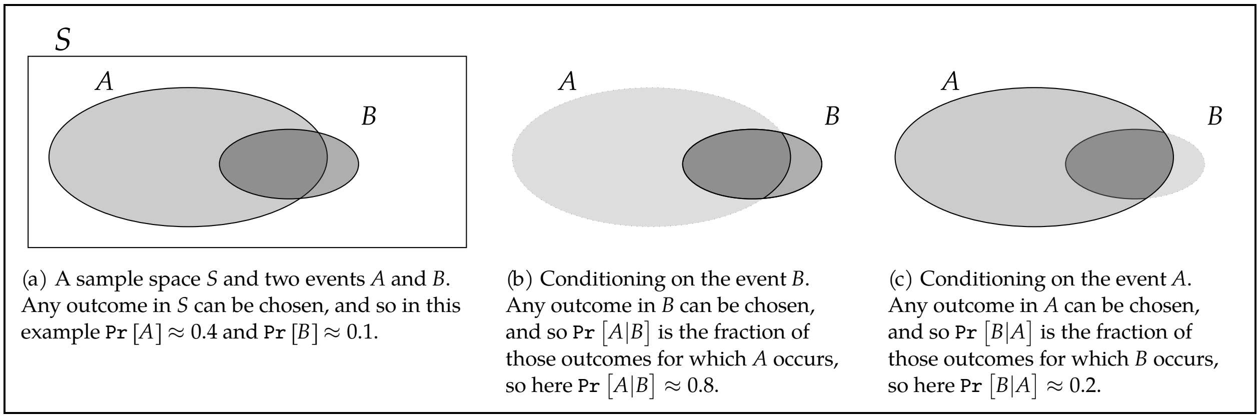
We know that the domino you drew must have been either ⚀⚁ or ⚀⚅. These two dominoes were equally likely to be drawn, and 1 of these 2 has a ⚁, so there's a $\frac{1}{2}$ probability that you drew a ⚁. Using conditional probability notation, we can write this quantity as $\texttt{Pr}[\text{⚁}|\text{⚀}] = \frac{1}{2}$.

3. We are computing $\texttt{Pr}[\text{⚁}|\text{⚄}]$, the probability of a ⚁ *given that* we observed a ⚄. By the definition of conditional probability, we have

$$\texttt{Pr}[\text{⚁}|\text{⚄}] = \frac{\texttt{Pr}[\text{⚁} \cap \text{⚄}]}{\texttt{Pr}[\text{⚄}]} = \frac{0}{\frac{1}{7}} = 0.$$

(A less notationally heavy way of writing this argument: because we see a ⚄, we know that the domino you drew must have been ⚄⚅. This domino doesn't have a ⚁ and so there's zero chance that we observe a ⚁.)

Conditional probability as "zooming in" (and another example)



(a) A sample space S and two events A and B. Any outcome in S can be chosen, and so in this example $\texttt{Pr}[A] \approx 0.4$ and $\texttt{Pr}[B] \approx 0.1$.

(b) Conditioning on the event B. Any outcome in B can be chosen, and so $\texttt{Pr}[A|B]$ is the fraction of those outcomes for which A occurs, so here $\texttt{Pr}[A|B] \approx 0.8$.

(c) Conditioning on the event A. Any outcome in A can be chosen, and so $\texttt{Pr}[B|A]$ is the fraction of those outcomes for which B occurs, so here $\texttt{Pr}[B|A] \approx 0.2$.

Figure 10.18: A view of conditional probability.

Intuitively, we can think of $\texttt{Pr}[A|B]$ as "zooming" the universe down to the set B. The basic idea that we used in Example 10.20 was to narrow the set of possible outcomes to those consistent with the observed partial data about the drawn domino, and then compute the fraction of the narrowed sample space for which A occurs. This view of conditional probability is illustrated in Figure 10.18.

Here's one more example, where we condition on slightly more complex events.

Example 10.21 (Coin flips)

Problem: Flip a fair coin 10 times (with all flips independent: the ith flip has no effect on the jth flip for $j \neq i$). Write H to denote the event of getting at least 9 heads.

1. What is $\Pr[H]$?
2. Let A be the event "the first flip comes up heads." What is $\Pr\left[H|A\right]$?
3. Let B be the event "the first flip comes up tails." What is $\Pr\left[H|B\right]$?
4. Let C be the event "the first three flips come up heads." What is $\Pr\left[H|C\right]$?
5. Let D be the event "we get at least 8 heads." What is $\Pr\left[H|D\right]$?

Solution: 1. Observe that every outcome—every element of $\{\text{H}, \text{T}\}^{10}$—is equally likely, each with probability $1/2^{10}$. The number of sequences of 10 flips with 9 or 10 heads is $\binom{10}{9} + \binom{10}{10} = 10 + 1 = 11$, so $\Pr[H] = 11/2^{10} \approx 0.0107$.

For the conditional probabilities, we will compute $\Pr[H \cap X]$ and $\Pr[X]$ for each of the stated events X. The final answer is their ratio. Because each outcome is equally likely, we only have to compute the cardinality of the given events (and the cardinality of their intersection with H) to answer the questions.

2. For A (the first flip comes up H), we have $|A \cap H| = 10$: there are 9 outcomes with one Tails that start with a Heads (HTHHHHHHHH, HHTHHHHHHH, ..., HHHHHHHHHT) and 1 outcome with zero Tails (HHHHHHHHHH). Thus $\Pr[A \cap H] = 10/2^{10}$. Obviously $\Pr[A] = \frac{1}{2}$. Thus

$$\Pr\left[H|A\right] = \frac{\Pr[A \cap H]}{\Pr[A]} = \frac{10/2^{10}}{1/2} = \frac{10}{2^9} \approx 0.01953.$$

3. For B (the first flip comes up T), we've already "used up" the single permitted non-heads in the first flip, so there's only one outcome in $B \cap H$, namely THHHHHHHHH. And, again, obviously $\Pr[B] = \frac{1}{2}$. Therefore we have

$$\Pr\left[H|B\right] = \frac{\Pr[B \cap H]}{\Pr[B]} = \frac{1/2^{10}}{1/2} = \frac{1}{2^9} \approx 0.00195.$$

4. For C (the first three flips come up H), we have $\Pr[C] = \frac{1}{8}$. The outcomes in $C \cap H$ are exactly those that start with HHH followed by 6+ heads in the last 7 flips. There are $\binom{7}{7} + \binom{7}{6} = 8$ such outcomes. Thus

$$\Pr\left[H|C\right] = \frac{\Pr[C \cap H]}{\Pr[C]} = \frac{8/2^{10}}{1/8} = \frac{64}{2^{10}} \approx 0.0625.$$

5. For D (there are at least 8 heads), we have $\Pr[H \cap D] = \Pr[H] = 11/2^{10}$. (There are no outcomes in which we get 9+ heads but fail to get 8+ heads!) The probability of getting 8+ heads in 10 fair flips is

$$\Pr[D] = \frac{\binom{10}{8} + \binom{10}{9} + \binom{10}{10}}{2^{10}} = \frac{45 + 10 + 1}{2^{10}} = \frac{56}{2^{10}}.$$

And therefore

$$\Pr\left[H|D\right] = \frac{\Pr[D \cap H]}{\Pr[D]} = \frac{11/2^{10}}{56/2^{10}} = \frac{11}{56} \approx 0.1964.$$

To repeat the word of warning from early in this chapter: *it can be very difficult to have good intuition about probability questions.* For example, the last problem in Example 10.21 asked for the probability of getting 9+ heads in 10 flips *conditioned on getting* 8+ *heads.* It may be easy to talk yourself into believing that, of the times that we get 8+ heads, there's a $\approx 50\%$ chance of getting 9 or more heads. ("Put aside the first 8 heads, and look at one of the other flips—it's heads with probability $\frac{1}{2}$, so we get a 9th heads with probability $\frac{1}{2}$.") But this intuition is blatantly wrong. Another way of thinking about the calculation in the last part of Example 10.21 is to observe that there are 56 outcomes with 8, 9, or 10 heads. Only 11 of these outcomes have 9 or 10 heads. Each outcome is equally likely. So if we're promised that one of the 56 outcomes occurred, then there's an $\frac{11}{56}$ chance that one of the 11 occurred.

Taking it further: So far, we have considered only random processes in which each outcome that can occur does so with probability $\varepsilon > 0$—that is, there have been no infinitesimal probabilities. But we can imagine scenarios in which infinitesimal probabilities make sense.

For example, imagine a probabilistic process that chooses a real number x between 0 and 1, where each element of the sample space $S = \{x : 0 \leq x \leq 1\}$ is equally likely to be chosen. We can make probabilistic statements like $\texttt{Pr}\,[x \leq 0.5] = \frac{1}{2}$—half the time, we end up with $x \leq 0.5$, half the time we end up with $x \geq 0.5$—but for *any* particular value c, *the probability that* $x = c$ *is zero!* (Perhaps bizarrely, $\texttt{Pr}\,[x \leq 0.5] = \texttt{Pr}\,[x < 0.5]$. Indeed, $\texttt{Pr}\,[x = 0.5]$ *cannot be* $\varepsilon > 0$, for any ε. Every possible outcome has to have that same probability ε of occurring, and for any $\varepsilon > 0$ there are more than $\frac{1}{\varepsilon}$ real numbers between 0 and 1. So we'd violate (10.1) if we had $\texttt{Pr}\,[x = 0.5] > 0$.)

To handle infinitesimal probabilities, we need calculus. We can describe the above circumstance with a *probability density function* $p : S \to [0, 1]$, so that, in place of (10.1), we require

$$\int_{x \in S} p(x)dx = 1.$$

(For a uniformly chosen $x \in [0, 1]$, we have $p(x) = 1$; for a uniformly chosen $x \in [0, 100]$, we have $p(x) = \frac{1}{100}$.) Some of the statements that we've made in this chapter don't apply in the infinitesimal case. For example, the "zooming in" view of conditional probability from Figure 10.18 doesn't quite work in the infinitesimal case. In fact, we can consider questions about $\texttt{Pr}\left[A|B\right]$ even when $\texttt{Pr}\,[B] = 0$, like *what is the probability that a uniformly chosen* $x \in [0, 100]$ *is an integer, conditioned on x being a rational number?*. (And Exercise 10.70—if $\texttt{Pr}\,[B] = 0$, then A and B are independent—isn't true with infinitesimal probabilities.) But details of this infinitesimal version of probability theory are generally outside of our concern here, and are best left to a calculus-based/analysis-based textbook on probability.

The restriction to non-infinitesimal probabilities is generally a reasonable one to make for CS applications, but it *is* a genuine restriction. (It's worth noting that we *have* encountered an infinite sample space before—just one that didn't have any infinitesimal probabilities. In a geometric distribution with parameter $\frac{1}{2}$, for example, any positive integer k is a possible outcome, with $\texttt{Pr}\,[k] = 1/2^k$, which is a finite, albeit very small, probability for any positive integer k.)

10.3.3 *Bayes' Rule and Calculating with Conditional Probability*

Here, we'll briefly introduce a few simple but useful ways of thinking about conditional probability: the connection between independence of events and conditional probability; a few ways of thinking about plain (unconditional) probability using conditional probability; and, finally, *Bayes' Rule*, a tremendously useful formula that relates $\texttt{Pr}\left[A|B\right]$ and $\texttt{Pr}\left[B|A\right]$.

Relating independence of events and conditional probability

Consider two events A and B for which $\texttt{Pr}\,[B] \neq 0$. Observe that A and B are independent if and only if $\texttt{Pr}\left[A|B\right] = \texttt{Pr}\left[A\right]$:

$$\begin{aligned}
A \text{ and } B \text{ are independent} &\Leftrightarrow \texttt{Pr}\left[A\right] \cdot \texttt{Pr}\left[B\right] = \texttt{Pr}\left[A \cap B\right] && \textit{definition of independence} \\
&\Leftrightarrow \texttt{Pr}\left[A\right] = \frac{\texttt{Pr}\left[A \cap B\right]}{\texttt{Pr}\left[B\right]} && \textit{dividing by } \texttt{Pr}\left[B\right] \\
&\Leftrightarrow \texttt{Pr}\left[A\right] = \texttt{Pr}\left[A|B\right]. && \textit{definition of } \texttt{Pr}\left[A|B\right]
\end{aligned}$$

(Note that this calculation doesn't work when $\texttt{Pr}\left[B\right] = 0$—we can't divide by 0, and $\texttt{Pr}\left[A|B\right]$ is undefined—but see Exercise 10.70.) Notice again that this relationship is an if-and-only-if relationship: when A and B are not independent, then $\texttt{Pr}\left[A\right]$ and $\texttt{Pr}\left[A|B\right]$ *must* be different. Here is a small example:

Example 10.22 (Self-mapped letters in substitution ciphers)
In Example 10.16, we showed that, for a random permutation π of the alphabet, the events Q (`Q` is mapped to itself by π) and Z (`Z` is mapped to itself by π) were not independent: specifically, $\texttt{Pr}\left[Q\right] = \frac{1}{26}$, $\texttt{Pr}\left[Z\right] = \frac{1}{26}$, and $\texttt{Pr}\left[Q \cap Z\right] = \frac{1}{25 \cdot 26}$. Thus

$$\texttt{Pr}\left[Q|Z\right] = \frac{\texttt{Pr}\left[Q \cap Z\right]}{\texttt{Pr}\left[Z\right]} = \frac{1/(25 \cdot 26)}{1/26} = \frac{1}{25}.$$

Compare $\texttt{Pr}\left[Q|Z\right] = \frac{1}{25}$ to $\texttt{Pr}\left[Q\right] = \frac{1}{26}$: thus, knowing that `Z` is mapped to itself makes it *slightly more likely* that `Q` is also mapped to itself. The reason that Z makes Q slightly more probable is that, when Z occurs, `Z` cannot be mapped to `Q`, so there are only 25 letters "competing" to be mapped to `Q` instead of 26.

Problem-solving tip: Often it is easier to get intuition about a probabilistic statement by imagining an absurdly small variant of the problem. Here, for example, imagine a 2-letter alphabet `Q`,`Z`. Then if `Z` is mapped to itself *then* `Q` *must also be mapped to itself.* So $\texttt{Pr}\left[Q\right] = \frac{1}{2}$, but $\texttt{Pr}\left[Q|Z\right] = 1$.

Intersections and conditional probability

The definition of conditional probability (Definition 10.10) states that

$$\texttt{Pr}\left[A|B\right] = \frac{\texttt{Pr}\left[A \cap B\right]}{\texttt{Pr}\left[B\right]}.$$

Multiplying both sides of this equality by $\texttt{Pr}\left[B\right]$ yields a useful way of thinking about the probability of intersections:

Theorem 10.2 (The Chain Rule)
Let A and B be arbitrary events. Then

$$\texttt{Pr}\left[A \cap B\right] = \texttt{Pr}\left[B\right] \cdot \texttt{Pr}\left[A|B\right].$$

And, more generally, for events $A_1, A_2, \ldots, A_k$, we have

$$\begin{aligned}
&\texttt{Pr}\left[A_1 \cap A_2 \cap A_3 \cap \cdots \cap A_k\right] \\
&= \texttt{Pr}\left[A_1\right] \cdot \texttt{Pr}\left[A_2|A_1\right] \cdot \texttt{Pr}\left[A_3|A_1 \cap A_2\right] \cdot \cdots \cdot \texttt{Pr}\left[A_k|A_1 \cap \cdots \cap A_{k-1}\right].
\end{aligned}$$

If we're interested in the probability that A and B occur, then we need it to be the case that A occurs—and, *conditioned on A occurring*, B occurs too.

Example 10.23 (Drawing a heart flush in poker)

Problem: A *flush* in poker is a 5-card hand, all of which are the same suit. What is the probability of drawing a heart flush from a randomly shuffled deck?

Solution: We can draw any heart first. We have to keep drawing hearts to get a flush, so for $2 \leq k \leq 5$, the kth card we draw must be one of the remaining $14 - k$ hearts from the $53 - k$ cards left in the deck. That is, writing H_i to denote the event that the ith card drawn is a heart:

$$\begin{aligned}
&\texttt{Pr}\left[H_1 \cap H_2 \cap H_3 \cap H_4 \cap H_5\right] \\
&= \texttt{Pr}\left[H_1\right] \cdot \texttt{Pr}\left[H_2|H_1\right] \cdot \texttt{Pr}\left[H_3|H_{1,2}\right] \cdot \texttt{Pr}\left[H_4|H_{1,2,3}\right] \cdot \texttt{Pr}\left[H_5|H_{1,2,3,4}\right] \\
&= \frac{13}{52} \cdot \frac{12}{51} \cdot \frac{11}{50} \cdot \frac{10}{49} \cdot \frac{9}{48} \\
&= \frac{154440}{311875200} \approx 0.00049519807.
\end{aligned}$$

(We could also have directly computed this quantity via counting: there are $\binom{13}{5}$ hands of 5 hearts, and $\binom{52}{5}$ total hands. Thus the fraction of all hands that are heart flushes is

$$\frac{\binom{13}{5}}{\binom{52}{5}} = \frac{\frac{13!}{5!\cdot 8!}}{\frac{52!}{5!\cdot 47!}} = \frac{13! \cdot 47!}{8! \cdot 52!} = \frac{13 \cdot 12 \cdot 11 \cdot 10 \cdot 9}{52 \cdot 51 \cdot 50 \cdot 49 \cdot 48},$$

which is the same quantity that we found above.)

We can use the chain rule to compute the probability of an event A by making the (obvious!) observation that another event B either occurs or doesn't occur:

Theorem 10.3 (The Law of Total Probability)

Let A and B be arbitrary events. Then

$$\texttt{Pr}\left[A\right] = \texttt{Pr}\left[A|B\right] \cdot \texttt{Pr}\left[B\right] + \texttt{Pr}\left[A|\,\overline{B}\,\right] \cdot \texttt{Pr}\left[\,\overline{B}\,\right].$$

Proof. We'll proceed by splitting A into two disjoint subsets, $A \cap B$ and $A - B$ (which is otherwise known as $A \cap \overline{B}$):

$$\begin{aligned}
\texttt{Pr}\left[A\right] &= \texttt{Pr}\left[(A \cap B) \cup (A \cap \overline{B})\right] && A = (A \cap B) \cup (A \cap \overline{B}) \\
&= \texttt{Pr}\left[A \cap B\right] + \texttt{Pr}\left[A \cap \overline{B}\right] && A \cap B \text{ and } A \cap \overline{B} \text{ are disjoint} \\
&= \texttt{Pr}\left[A|B\right] \cdot \texttt{Pr}\left[B\right] + \texttt{Pr}\left[A|\,\overline{B}\,\right] \cdot \texttt{Pr}\left[\,\overline{B}\,\right]. && \text{the chain rule}
\end{aligned}$$

Thus the theorem follows. □

Here's a simple example of using the law of total probability:

Example 10.24 (Binary Symmetric Channel)

We wish to transmit a 1-bit message from a sender to a receiver. The sender's message is 0 with probability 0.3, and it's 1 with probability 0.7. The sender sends this data using a communication channel that corrupts (that is, flips) every transmitted bit with probability 0.25. Then the probability that the receiver receives a "1" message is

$$\begin{aligned}
\texttt{Pr}\left[\text{receive } 1\right] &= \texttt{Pr}\left[\text{receive } 1 \middle| \text{send } 1\right] \cdot \texttt{Pr}\left[\text{send } 1\right] \;+ \texttt{Pr}\left[\text{receive } 1 \middle| \text{send } 0\right] \cdot \texttt{Pr}\left[\text{send } 0\right] \\
&= (0.75 \cdot 0.7) \,+\, (0.25 \cdot 0.3) \\
&= 0.525 + 0.075 = 0.6.
\end{aligned}$$

Taking it further: The *binary symmetric channel* is given this name because it transmits a bit (it's *binary*) and it corrupts a 0 with the same probability as it corrupts a 1 (it's *symmetric*). (See Figure 10.19; view each arrow in the channel as transforming a particular input bit to a particular output bit, with the indicated probability.)

The binary symmetric channel is one of the most basic forms of a noisy communication channel (that is, a channel that does not perfectly transmit its input without any chance of corruption). The subfield of *information theory* is devoted to analyzing topics like the (theoretical) efficiency of communication channels, including the binary symmetric channel. For much more, see a textbook on information theory.[5]

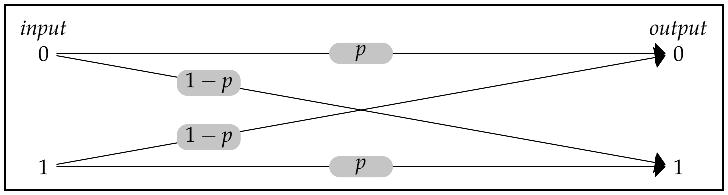


Figure 10.19: The binary symmetric channel.

Bayes' Rule

Bayes' Rule is a simple—but tremendously useful—rule for "flipping around" a conditional probability statement. It allows us to express the conditional probability of A given B in terms of the conditional probability of B given A:

Theorem 10.4 (Bayes' Rule)

For any two events A and B:

$$\texttt{Pr}\left[A \middle| B\right] = \frac{\texttt{Pr}\left[B \middle| A\right] \cdot \texttt{Pr}\left[A\right]}{\texttt{Pr}\left[B\right]}.$$

Bayes' Rule is named after Thomas Bayes, an 18th-century English mathematician.

Proof. Applying the chain rule to break apart $\texttt{Pr}\left[A \cap B\right]$ "in both orders," we have

$$\begin{aligned}
\texttt{Pr}\left[A \cap B\right] &= \texttt{Pr}\left[A \middle| B\right] \cdot \texttt{Pr}\left[B\right] \\
\texttt{Pr}\left[B \cap A\right] &= \texttt{Pr}\left[B \middle| A\right] \cdot \texttt{Pr}\left[A\right].
\end{aligned}$$

The left-hand sides of these equations are identical because $A \cap B = B \cap A$ (and therefore $\texttt{Pr}\left[A \cap B\right] = \texttt{Pr}\left[B \cap A\right]$), so their right-hand sides are equal, too:

$$\texttt{Pr}\left[A \middle| B\right] \cdot \texttt{Pr}\left[B\right] = \texttt{Pr}\left[B \middle| A\right] \cdot \texttt{Pr}\left[A\right].$$

Dividing both sides of this equality by $\texttt{Pr}\left[B\right]$ yields the desired equation:

$$\texttt{Pr}\left[A \middle| B\right] = \frac{\texttt{Pr}\left[B \middle| A\right] \cdot \texttt{Pr}\left[A\right]}{\texttt{Pr}\left[B\right]}. \qquad \square$$

[5] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 1991.

Here are a couple of simple examples of using Bayes' Rule:

Example 10.25 (Binary Symmetric Channel, again)
As in Example 10.24, assume a sender transmits a 0 with probability 0.3 and a 1 with probability 0.7 across a channel that corrupts every bit with probability 0.25. We showed in Example 10.24 that $\texttt{Pr}\,[\text{receive } 1] = 0.6$ and thus $\texttt{Pr}\,[\text{receive } 0] = 0.4$. Then the probability that the receiver receiving a "1" message was indeed sent a 1 is

$$\begin{aligned}\texttt{Pr}\left[\text{message sent was } 1 \middle| \text{receive } 1\right] &= \frac{\texttt{Pr}\left[\text{receive } 1 \middle| \text{send } 1\right] \cdot \texttt{Pr}\,[\text{send } 1]}{\texttt{Pr}\,[\text{receive } 1]} && \textit{by Bayes' Rule}\\ &= \frac{0.75 \cdot 0.7}{0.6} = 0.875.\end{aligned}$$

And the probability that the receiver receiving a "0" message was indeed sent a 0 is

$$\begin{aligned}\texttt{Pr}\left[\text{message sent was } 0 \middle| \text{receive } 0\right] &= \frac{\texttt{Pr}\left[\text{receive } 0 \middle| \text{send } 0\right] \cdot \texttt{Pr}\,[\text{send } 0]}{\texttt{Pr}\,[\text{receive } 0]} && \textit{by Bayes' Rule}\\ &= \frac{0.75 \cdot 0.3}{0.4} = 0.5625.\end{aligned}$$

(Qualitatively, these numbers tell us that most of received ones were actually sent as ones, but barely more than half of the received zeros were actually sent as zeros.)

Example 10.26 (9+heads, again)
We flip a fair coin 10 times. As in Example 10.21, let A denote the event that the first flip comes up heads and let H denote the event that there are 9 or more heads in the 10 flips. (There we showed $\texttt{Pr}\,[H] = 11/2^{10}$, $\texttt{Pr}\,[A] = \frac{1}{2}$, and $\texttt{Pr}\left[H \middle| A\right] = 10/2^9$.) Then

$$\texttt{Pr}\left[A \middle| H\right] = \frac{\texttt{Pr}\left[H \middle| A\right] \cdot \texttt{Pr}\,[A]}{\texttt{Pr}\,[H]} = \frac{(10/2^9) \cdot \frac{1}{2}}{11/2^{10}} = \frac{10}{11}.$$

Taking it further: A *speech recognition system* is supposed to "listen" to speech in a language like English, and recognize the words that are being spoken. Bayes' Rule allows us to think about two different types of evidence that such a system uses in deciding what words it "thinks" are being said; see p. 1036.

A particularly important application of Bayes' Rule is in "updating" one's beliefs about the world by observing new information. (Here "beliefs" take the form of a probability distribution.) One starts with a *prior distribution* which one then updates based on *evidence* to produce a *posterior distribution.* Here are two examples:

The prior (*pre* = before) is your best guess of the probability of the event prior to seeing the produced evidence; the posterior (*post* = after) is your best guess after seeing the evidence.

Example 10.27 (Alice the CS major)
We are interested in whether a student (let's call her Alice) is a computer science major. Our prior for Alice might be $\texttt{Pr}\left[\text{CS major}\right] = 0.05$ because 5% of students are CS majors. We learn that Alice took Ancient Philosophy. If we know that 10% of students as a whole take Ancient Philosophy, and 50% of CS majors do, then

$$\texttt{Pr}\left[\text{CS major}\middle|\text{phil}\right] = \frac{\texttt{Pr}\left[\text{phil}\middle|\text{CS major}\right] \cdot \texttt{Pr}\left[\text{CS major}\right]}{\texttt{Pr}\left[\text{phil}\right]} = \frac{0.5 \cdot 0.05}{0.10} = 0.25.$$

Our posterior distribution (that is, the updated guess) is that there is now a 25% chance that Alice is a CS major.

Example 10.28 (Flipping a coin to decide which coin to flip)

I have two coins in an opaque bag. The coins are visually indistinguishable, but one coin is fair ($\texttt{Pr}\left[\text{H}\right] = 0.5$); the other coin is 0.75-biased ($\texttt{Pr}\left[\text{H}\right] = 0.75$). I pull one of the two coins out at random.

- My *prior distribution* is that there is a 50% chance I'm holding the fair coin, and a 50% chance I'm holding the biased coin. (That is, $\texttt{Pr}\left[\text{biased}\right] = \texttt{Pr}\left[\text{fair}\right] = 0.5$.)

I flip the coin that I'm holding. It comes up heads.

- The *evidence* is the Heads flip.

Because the biased coin is more likely to produce Heads flips than the fair coin is (and we saw Heads), this evidence should make us view it as more likely that the coin that I'm holding is the biased coin. Let's compute my *posterior probability:*

- The posterior probability of an event is the probability of that event *conditioned on the observed evidence.* So we wish to compute $\texttt{Pr}\left[\text{biased}\middle|\text{H}\right]$:

$$\begin{aligned}
\texttt{Pr}\left[\text{biased}\middle|\text{H}\right] &= \frac{\texttt{Pr}\left[\text{H}\middle|\text{biased}\right] \cdot \texttt{Pr}\left[\text{biased}\right]}{\texttt{Pr}\left[\text{H}\right]} && \textit{Bayes' Rule} \\
&= \frac{\texttt{Pr}\left[\text{H}\middle|\text{biased}\right] \cdot \texttt{Pr}\left[\text{biased}\right]}{\texttt{Pr}\left[\text{H}\middle|\text{biased}\right] \cdot \texttt{Pr}\left[\text{biased}\right] + \texttt{Pr}\left[\text{H}\middle|\text{fair}\right] \cdot \texttt{Pr}\left[\text{fair}\right]} && \textit{Law of Total Probability} \\
&= \frac{0.75 \cdot \texttt{Pr}\left[\text{biased}\right]}{(0.75 \cdot \texttt{Pr}\left[\text{biased}\right]) + (0.5 \cdot \texttt{Pr}\left[\text{fair}\right])} && \textit{the given biases of the coins: } 0.75 \textit{ for biased, } 0.5 \textit{ for fair} \\
&= \frac{0.75 \cdot 0.5}{(0.75 \cdot 0.5) + (0.5 \cdot 0.5)} && \texttt{Pr}\left[\textit{biased}\right] = \texttt{Pr}\left[\textit{fair}\right] = 0.5\textit{, as defined by the prior} \\
&= \frac{0.375}{0.375 + 0.25} = 0.6.
\end{aligned}$$

So the posterior probability is $\texttt{Pr}\left[\text{biased}\middle|\text{H}\right] = 0.6$ and $\texttt{Pr}\left[\text{fair}\middle|\text{H}\right] = 0.4$.

Taking it further: The idea of Bayesian reasoning is used frequently in many applications of computer science—any time a computational system weighs various pieces of evidence in deciding what kind of action to take in a particular situation. One of the most noticeable examples of this type of reasoning occurs in *Bayesian spam filters*; see p. 1037 for more.

Computer Science Connections

Speech Recognition, Bayes' Rule, and Language Models

A software system for *speech recognition* must solve the following problem: given an audio stream $\mathcal{S}$ of spoken English as input, produce as output a transcript $\mathcal{W}$ of the words in $\mathcal{S}$. There will be many candidate transcripts of $\mathcal{S}$, and generally the task of the system is to produce the *most likely sequence of words given the audio stream*—that is, to find the $\mathcal{W}^*$ maximizing $\texttt{Pr}\left[\mathcal{W}^*|\mathcal{S}\right]$.

Using Bayes' Rule, we can rephrase $\texttt{Pr}\left[\mathcal{W}^*|\mathcal{S}\right]$ into an expression that's easier to understand:

$$\begin{aligned}
&\text{the } \mathcal{W}^* \text{ maximizing } \texttt{Pr}\left[\mathcal{W}^*|\mathcal{S}\right] \\
&= \text{the } \mathcal{W}^* \text{ maximizing } \frac{\texttt{Pr}\left[\mathcal{S}|\mathcal{W}^*\right] \cdot \texttt{Pr}\left[\mathcal{W}^*\right]}{\texttt{Pr}\left[\mathcal{S}\right]} && \textit{Bayes' Rule} \\
&= \text{the } \mathcal{W}^* \text{ maximizing } \texttt{Pr}\left[\mathcal{S}|\mathcal{W}^*\right] \cdot \texttt{Pr}\left[\mathcal{W}^*\right]. && \texttt{Pr}\left[\mathcal{S}\right] \textit{ is the same for each } \mathcal{W}^*
\end{aligned}$$

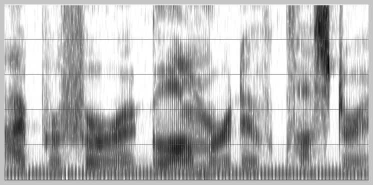


Figure 10.20: A *spectrogram* representation of an audio stream: the x-axis represents time, the y-axis represents frequency, and the darkness of the shading denotes the intensity of sound at that particular frequency at that particular time. (See p. 234 for more discussion.) The task is to turn this representation into its most probable sequence of words—in this case, the sentence "I prefer agglomerative clustering."

Thus there are two valuable sources of data for evaluating a candidate $\mathcal{W}$:

- $\texttt{Pr}\left[\mathcal{S}|\mathcal{W}\right]$, the *likelihood of the observation*: the probability that this sound stream would have been produced if $\mathcal{W}$ were the sequence of words; and
- $\texttt{Pr}\left[\mathcal{W}\right]$, the *probability of this output*: the probability of this sequence of words being uttered at all.

For example, *even if* the audio stream is a better acoustic match for the phrase *whirled Siri string*, you'd want your system to prefer the phrase *World Series ring*—because an English speaker is far more likely to say the latter phrase than the former. (That is, $\texttt{Pr}\left[\textit{World Series ring}\right]$ is much higher than $\texttt{Pr}\left[\textit{whirled Siri string}\right]$.) Of course, we still must take into account the audio stream $\mathcal{S}$—otherwise, *regardless of the audio,* we'd end up with a system that produced precisely the same output sentence (the most common sentence in English: *I'm sorry!*, or whatever it is) for any input sound stream.

Generally speaking, the quantity $\texttt{Pr}\left[\mathcal{S}|\mathcal{W}\right]$ would be estimated by an acoustic model of the vocal tract: if I'm trying to say *Camp Utah seance,* what is the probability that I produce a particular stream $\mathcal{S}$ of sounds?

The quantity $\texttt{Pr}\left[\mathcal{W}\right]$ is estimated by what's called a *language model.* We would acquire a large collection of English text, and then try to use that data to estimate how likely a particular sequence is. The simplest language model is the *unigram* model:

- from a giant data set with N total words, for each word w we count up the number of times $n(w)$ that w appears.
- if $\mathcal{W} = w_1, w_2, \ldots, w_k$, we estimate $\texttt{Pr}\left[\mathcal{W}\right]$ as $\frac{n(w_1)}{N} \cdot \frac{n(w_2)}{N} \cdot \;\ldots\; \cdot \frac{n(w_k)}{N}$.

A more complex language model might use *bigrams*—two-word sequences—instead; we count the number of occurrences of w_i, w_{i+1} consecutively in the giant data set, and estimate $\texttt{Pr}\left[\mathcal{W}\right]$ based on these counts. Other more complex language models are used in real systems.[6] There's also a great deal of complication with avoiding *overfitting* of the language model to the training data. (In addition to speech recognition, a variety of other natural language processing problems are generally solved with the same general approach.)

For much more, see

[6] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Pearson Prentice Hall, 2nd edition, 2008.

Computer Science Connections

Bayesian Modeling and Spam Filtering

There are, it's estimated, a few hundred billion email messages sent on earth per day. Of those, a significant fraction of those messages are unsolicited, unwanted bulk messages—that is, what's commonly known as *spam*. Somewhere between 50% and 95% of emails are currently spam. (It's hard to be precise; statistics and definitions of spam vary, and there's change over time as certain spammers are shut down, or not.)

See statistics on email and spam produced by the Radicati Group, for example: `www.radicati.com`.

The basic idea of a *spam filter* is to estimate the probability that a particular message m is spam. The email client, or possibly the individual user, can choose a threshold p; a message m for which $\texttt{Pr}\left[m \text{ is spam}\right] \geq p$ is placed into a spam folder. The choice of p depends on the user's relative concern about false positives (nonspam messages that end up being incorrectly treated as spam) versus false negatives (spam messages that end up being incorrectly left in the inbox). So, how might a spam filter actually make its decisions? Here's one approach, based fundamentally on Bayes' Rule. Consider a message consisting of words $w_1, w_2, \ldots, w_n$; we must compute $\texttt{Pr}\left[\text{spam}|w_1, w_2, \ldots w_n\right]$. Using Bayes' Rule, we turn around this probability:

$$\texttt{Pr}\left[\text{spam}|w_1, w_2, \ldots w_n\right] = \frac{\texttt{Pr}\left[w_1, w_2, \ldots w_n|\text{spam}\right] \cdot \texttt{Pr}\left[\text{spam}\right]}{\texttt{Pr}\left[w_1, w_2, \ldots w_n\right]}$$

It's a good test of your probabilistic intuition to ask: supposing that we have a spam filter that correctly classifies 90% of email messages as spam/nonspam, and 95% of email messages are spam, what fraction of email in your inbox is nonspam? The answer, by Bayes' Rule:

$$\begin{aligned} &\texttt{Pr}\left[\text{nonspam}|\text{inbox}\right] \\ &= \frac{\texttt{Pr}\left[\text{inbox}|\text{nonspam}\right] \texttt{Pr}\left[\text{nonspam}\right]}{\begin{pmatrix} \texttt{Pr}\left[\text{inbox}|\text{nonspam}\right] \texttt{Pr}\left[\text{nonspam}\right] \\ + \texttt{Pr}\left[\text{inbox}|\text{spam}\right] \texttt{Pr}\left[\text{spam}\right] \end{pmatrix}} \\ &= \frac{0.9 \cdot 0.05}{0.9 \cdot 0.05 + 0.1 \cdot 0.95} \\ &= \frac{0.045}{0.045 + .095} \\ &= 0.3214 \cdots. \end{aligned}$$

In other words, a full two thirds of messages in your inbox would be spam!

And, by the law of total probability (every message is either spam or not spam), we can further rewrite this probability as

$$\frac{\texttt{Pr}\left[w_1, w_2, \ldots w_n|\text{spam}\right] \cdot \texttt{Pr}\left[\text{spam}\right]}{\texttt{Pr}\left[w_1, w_2, \ldots w_n|\text{spam}\right] \texttt{Pr}\left[\text{spam}\right] + \texttt{Pr}\left[w_1, w_2, \ldots w_n|\text{not spam}\right] \texttt{Pr}\left[\text{not spam}\right]}.$$

That is, we want to know: what is the probability that the sequence of words $w_1, \ldots, w_n$ would have been generated in a spam message, relative to the probability that $w_1, \ldots, w_n$ would have been generated in a spam or nonspam message? (These "relative probabilities" are weighted by the background probability of spam-vs.-nonspam messages.)

A *naïve Bayes classifier* uses an additional assumption: that the appearance of every word in an email is an independent event. That is, we're going to estimate $\texttt{Pr}\left[w_1, w_2, \ldots w_n\right]$ as if the probability of each w_i appearing does not depend on any other word appearing. (Obviously that assumption isn't right: the probability of the word `MORTGAGE` appearing is *not* independent of the probability of the word `RATE` appearing, in either spam or nonspam.)

$$\texttt{Pr}\left[w_1, w_2, \ldots w_n|\text{spam}\right] \approx \texttt{Pr}\left[w_1|\text{spam}\right] \cdot \texttt{Pr}\left[w_2|\text{spam}\right] \cdot \;\cdots\; \cdot \texttt{Pr}\left[w_n|\text{spam}\right].$$

Thus a naïve Bayes classifier estimates the probability of a message being generated as spam by multiplying a measure of "how spammy" each word is. A spam filter would still need to have two numbers associated with each word w_i—namely $\texttt{Pr}\left[w_i|\text{spam}\right]$ and $\texttt{Pr}\left[w_i|\text{nonspam}\right]$. We can estimate these numbers from a *training set* of spam/nonspam emails, with some sort of "smoothing" mechanism to improve our estimate of the spamminess of a word that doesn't appear in any of the training emails.[7]

For more about the training of these estimates, and about *text classification*—the broader version of the problem that we're trying to solve in spam filtering—again see:

[7] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Pearson Prentice Hall, 2nd edition, 2008.

10.3.4 Exercises

Choose one of the 12 *months of the year uniformly at random. (That is, choose a number uniformly from the set* $\{1, 2, \ldots, 12\}$*.) Indicate whether the following pairs of events are independent or dependent. Justify your answers.*

10.53 "The month number is even" and "the month number is divisible by 3."

10.54 "The month number is even" and "the month number is divisible by 5."

10.55 "The month number is even" and "the month number is divisible by 6."

10.56 "The month number is even" and "the month number is divisible by 7."

We flip a fair coin 6 *times. Which of these events are independent or dependent? Justify your answers.*

10.57 "The number of heads is even" and "the number of heads is divisible by 3."

10.58 "The number of heads is even" and "the number of heads is divisible by 4."

10.59 "The number of heads is even" and "the number of heads is divisible by 5."

10.60 We flip three fair coins, called a, b, and c. Are the events "The number of heads in $\{a, b\}$ is odd" and "The number of heads in $\{b, c\}$ is odd" independent or dependent?

10.61 How (if at all) would your answer to the previous exercise change if the three coins are p-biased? (That is, assume $\text{Pr}[a = \text{H}]$, $\text{Pr}[b = \text{H}]$, and $\text{Pr}[c = \text{H}]$ are all equal to p.)

Consider the list of words and the events in Figure 10.21. Choose a word at random from this list. Which of these pairs of events are independent? For the pairs that are dependent, indicate whether the events are positively or negatively correlated. Justify your answers.

10.62 A and B

10.63 A and C

10.64 B and C

10.65 A and D

10.66 A and E

10.67 $A \cap B$ and E

10.68 $A \cap C$ and E

10.69 $A \cap D$ and E

```
ABIDES
BASES
CAJOLED
DATIVE
EXUDE
FEDORA
GASOLINES
HABANERO
```

(a) Some words.

A : "the first letter of the word is a consonant."

B : "the second letter of the word is a consonant."

C : "the second letter of the word is a vowel."

D : "the last letter of the word is a consonant."

E : "the word has even length."

(b) Some events.

Figure 10.21: A word list from which we choose a random word, and some events.

Let A and B be arbitrary events in a finite sample space.

10.70 Prove that if $\text{Pr}[B] = 0$, then A and B are independent.

10.71 Prove that A and B are independent if and only if A and $\overline{B}$ are independent.

A substitution cipher *(see Example 10.16) is a simple cryptographic scheme in which we choose a permutation π of the alphabet, and replace each letter i with π_i. (Decryption is the same process, but backward: replace π_i by i.) However, substitution ciphers are susceptible to* frequency analysis, *in which an eavesdropper who observes the encrypted message (the* ciphertext*) infers that the most common letter in the ciphertext probably corresponds to the most common letter in English text (the letter* `E`*), the second-most common to the second-most common (*`T`*), and so on.*

10.72 *(programming required)* Write a program that generates a random permutation π of the alphabet, and encrypts a given input text using π. (Leave all non-alphabetic characters unchanged.)

10.73 *(programming required)* Write a program that takes a text as input, converts it to upper case, and produces as output a vector $\langle f_{\texttt{A}}, f_{\texttt{B}}, \ldots, f_{\texttt{Z}} \rangle$, where $f_\bullet$ is the fraction of letters in the input text that are the letter $\bullet$. (So f will be a probability distribution over the alphabet.)

10.74 *(programming required)* Write a program that, given a reference text and a text encrypted with an unknown substitution cipher, attempts to decrypt by mapping the most common encrypted letters, in order, to the most common reference letters. You can find useful reference files—for example, the complete works of Shakespeare—from Project Gutenberg, `http://www.gutenberg.org/`.

A Caesar cipher *is a special kind of substitution cipher in which the permutation π is generated by choosing a numerical* shift *s and moving all letters s steps forward in the alphabet, wrapping back to the beginning of the alphabet as necessary. (For example, with a shift of* 5, `A` → `F` *and* `W` → `B`*.)*

10.75 *(programming required)* Write a Caesar cipher encryption program that encrypts a given input text file with a randomly chosen shift in $\{0, 1, \ldots, 25\}$.

10.76 *(programming required)* If you run your decryption program from Exercise 10.74 on Caesar-ciphered text, you'll find that your program generally doesn't work perfectly. Write a Caesar-cipher-decrypting program that takes advantage of the fact that every letter is shifted by the same amount. Find the most probable s—the s that minimizes the difference in the probabilities of each letter from the reference text and the deciphered text. That is, minimize $\sum_i |f'_i - f_{i+s}|$, where f comes from the ciphertext and f' comes from the reference text.

Flip n fair coins. For any two distinct indices i and j with $1 \leq i < j \leq n$, define the event $A_{i,j}$ as

$$A_{i,j} := (\textit{the ith coin flip came up heads})\ \textit{XOR}\ (\textit{the jth coin flip came up heads}).$$

For example, for $n = 4$ and the outcome $\langle \text{T}, \text{T}, \text{H}, \text{H} \rangle$, the events $A_{1,3}$, $A_{1,4}$, $A_{2,3}$, and $A_{2,4}$ all occur; $A_{1,2}$ and $A_{3,4}$ do not. Thus, from n independent coin flips, we've defined $\Omega(n^2)$ different events—$\binom{n}{2}$, to be specific. In the next few exercises, you'll show that these $\binom{n}{2}$ events are pairwise independent, *but not* fully independent.

10.77 Let i and $j > i$ be arbitrary. Show that $\texttt{Pr}\left[A_{i,j}\right] = \frac{1}{2}$.

10.78 Let i and $j > i$ be arbitrary, and let i' and $j' > i'$ be arbitrary. Show that any two distinct events $A_{i,j}$ and $A_{i',j'}$ are independent. That is, show that $\texttt{Pr}\left[A_{i,j} | A_{i',j'}\right] = \texttt{Pr}\left[A_{i,j} | \overline{A_{i',j'}}\right] = \frac{1}{2}$ if $\{i,j\} \neq \{i',j'\}$.

10.79 Show that there is a set of three distinct A events that are *not* mutually independent. That is, identify three events $A_{i,j}$, $A_{i',j'}$, and $A_{i'',j''}$ where the sets $\{i,j\}$, $\{i',j'\}$, and $\{i'',j''\}$ are all different (though not necessarily disjoint). Then show that if you know the value of $A_{i,j}$ and $A_{i',j'}$, the probability of $A_{i'',j''} \neq \frac{1}{2}$.

Suppose that you have the dominoes in Figure 10.22, and you shuffle them and draw one domino uniformly at random. (More specifically, you choose any particular domino with probability $\frac{1}{12}$. After you've chosen the domino, you choose an orientation, with a 50–50 *chance of either side pointing to the left.) What are the following conditional probabilities? ("Even total" means that the sum of the two halves of the domino is even. "Doubles" means that the two halves are the same.)*

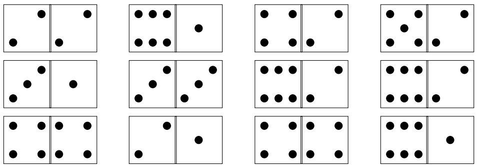


Figure 10.22: Some dominoes.

10.80 $\texttt{Pr}$ [even total|doubles]

10.81 $\texttt{Pr}$ [doubles|even total]

10.82 $\texttt{Pr}$ [doubles|at least one ⚃]

10.83 $\texttt{Pr}$ [at least one ⚃|doubles]

10.84 $\texttt{Pr}$ [total ≥ 7|doubles]

10.85 $\texttt{Pr}$ [doubles|total ≥ 7]

10.86 $\texttt{Pr}$ [even total|total ≥ 7]

10.87 $\texttt{Pr}$ [doubles|*left half* of drawn domino is ⚂]

10.88 Suppose A and B are mutually exclusive events—that is, $A \cap B = \varnothing$. Prove or disprove the following claim: A and B cannot be independent.

10.89 Let A and B be two events such that $\texttt{Pr}\left[A|B\right] = \texttt{Pr}\left[B|A\right]$. Which of the following is true? (a) A and B must be independent; (b) A and B must *not* be independent; or (c) A and B may or may not be independent (there's not enough information to tell). Justify your answer briefly.

Suppose, as we have done throughout the chapter, that $h : K \to \{1, \ldots, n\}$ is a random hash function.

10.90 Suppose that there are currently k cells in the array that are occupied. Consider a key $x \in K$ not currently stored in the hash table. What is the probability that the cell $h(x)$ into which x hashes is empty?

10.91 Suppose that you insert n distinct values $x_1, x_2, \ldots, x_n$ into an initially empty n-slot hash table. What is the probability that there are no collisions? (*Hint: if the first i elements have had no collisions, what is the probability that the $(i+1)$st hashed element does not cause a collision? Use Theorem 10.2 and Exercise 10.90.*)

There's a disease BCF ("base-case failure") that afflicts a small but very unfortunate fraction of the population. One in a thousand people in the population have BCF. Explain your answers to the following questions:

10.92 Doctor Genius has invented a BCF-detection test. Her test, though, isn't perfect:

- it has *false negatives*: if you do have BCF, then her test says that you're not sick with probability 0.01.
- it has *false positives*: if you don't have BCF, then her test says that you're sick with probability 0.03.

What is the probability p that Dr. Genius gives a random person x an erroneous diagnosis?

10.93 "Doctor" Quack has invented a BCF-detection test, too. He was a little confused by the statement "one in a thousand people in the population have BCF," so his test is this: no matter who the patient is, with probability $\frac{1}{1000}$ report "sick" and with probability $\frac{999}{1000}$ report "not sick." What is p now?

Alice wishes to send a 3-bit message 011 *to Bob, over a noisy channel that corrupts (flips) each transmitted bit independently with some probability. To combat the possibility of her transmitted message differing from the received message, she adds a parity bit to the end of her message (so that the transmitted message is* 0110*). [Bob checks that he receives a message with an even number of* 1*s, and if so interprets the first three received bits as the message.]*

10.94 Assume that each bit is flipped with probability 1%. Conditioned on receiving a message with an even number of 1s, what is the probability that the message Bob received is the message that Alice sent?

10.95 What if the probability of error is 10% per bit?

Suppose, as in Example 10.28, I have two coins—one fair and one p-biased. I pull one uniformly at random from an opaque bag, and flip it. What is $\texttt{Pr}$ [*I pulled the biased coin|the following observed flips*]*? Justify your answers.*

10.96 $p = \frac{2}{3}$, and I observe a single Heads flip.

10.97 $p = \frac{3}{4}$, and I observe the flip sequence HHHT.

10.98 $p = \frac{3}{4}$, and I observe the flip sequence HTTTHT.

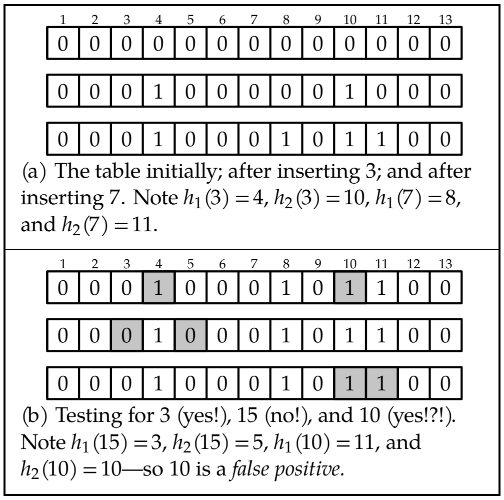
A Bloom filter *is a probabilistic data structure designed to store a set of elements from a universe U, allowing very quick query operations to determine whether a particular element has been stored.*[8] *Specifically, it supports the operations* **Insert**(x)*, which adds x to the stored set, and* **Lookup**(x)*, which reports whether x was previously stored. But, unlike most data structures for this problem, we will allow ourselves to (occasionally) make mistakes in lookups, in exchange for making these operations fast.*

Here's how a Bloom filter works. We will choose k different hash functions $h_1, \ldots, h_k : U \to \{1, \ldots, m\}$, and we will maintain an array of m bits, all initially set to zero. The operations are implemented as follows:

- *To insert x into the data structure, we set the k slots $h_1(x), h_2(x), \ldots, h_k(x)$ of the array to 1. (If any of these slots was already set to 1, we leave it as a 1.)*
- *To look up x in the data structure, we check that the k slots $h_1(x), h_2(x), \ldots, h_k(x)$ of the array are all set to 1. If they're all 1s, we report "yes"; if any one of them is a 0, we report "no."*

For an example, see Figure 10.98. Note that there can be a false positive *in a lookup: if all k slots corresponding to a query element x happen to have been set to 1 because of other insertions, then x will incorrectly be reported to be present.*

As usual, we treat each of the k hash functions as independently assigning each element of U to a uniformly chosen slot of the array. Suppose that we have an m-slot Bloom filter, with k independent hash functions, and we insert n elements into the data structure.



1	2	3	4	5	6	7	8	9	10	11	12	13
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	1	0	1	1	0	0

(a) The table initially; after inserting 3; and after inserting 7. Note $h_1(3) = 4$, $h_2(3) = 10$, $h_1(7) = 8$, and $h_2(7) = 11$.

1	2	3	4	5	6	7	8	9	10	11	12	13
0	0	0	1	0	0	0	1	0	1	1	0	0
0	0	0	1	0	0	0	1	0	1	1	0	0
0	0	0	1	0	0	0	1	0	1	1	0	0

(b) Testing for 3 (yes!), 15 (no!), and 10 (yes!?!). Note $h_1(15) = 3$, $h_2(15) = 5$, $h_1(10) = 11$, and $h_2(10) = 10$—so 10 is a *false positive.*

Figure 10.23: An example of a Bloom filter with $k = 2$ hash functions: $h_1(x) = x \bmod 13 + 1$ and $h_2(x) = x^2 \bmod 13 + 1$.

10.99 Suppose we have $k = 1$ hash functions, and we've inserted $n = 1$ element into the Bloom filter. Consider any particular slot of the m-slot table. What is the probability that this particular slot is still set to 0? (That is, what is the probability that this slot is *not* the slot set to 1 when the single element was inserted?)

10.100 Let the number k of hash functions be an arbitrary number $k \geq 1$, but continue to suppose that we've inserted only $n = 1$ element in the Bloom filter. What is the probability a particular slot is still set to 0 after this insertion?

10.101 Let the number k of hash functions be an arbitrary number $k \geq 1$, and suppose that we've inserted an arbitrary number $n \geq 1$ of elements into the Bloom filter. What is the probability a particular slot is still set to 0 after these insertions?

Define the false-positive rate *of a Bloom filter (with m slots, k hash functions, and n inserted elements) to be the probability that we incorrectly report that y is in the table when we query for an uninserted element y.*

For many years (starting with Bloom's original paper about Bloom filters), people in computer science believed that the false positive rate was precisely p^k, where $p = (1 -$ [your answer to Exercise 10.101]*). The justification was the following. Let B_i denote the event "slot $h_i(y)$ is occupied." We have a false positive if and only if $B_1, B_2, \ldots, B_k$ are all true. Thus*

$$\textit{the false positive rate} = \texttt{Pr}\left[B_1 \textit{ and } B_2 \textit{ and } \cdots \textit{ and } B_k\right].$$

You showed in the previous exercise that $\texttt{Pr}[B_i] = p$. **Everything up until here is correct; the next step in the argument, however, was not!** *Therefore, because the B_i events are independent,*

$$\textit{the false positive rate} = \texttt{Pr}\left[B_1 \textit{ and } B_2 \textit{ and } \cdots \textit{ and } B_k\right] = \texttt{Pr}\left[B_1\right] \cdot \texttt{Pr}\left[B_2\right] \cdots \texttt{Pr}\left[B_k\right] = p^k.$$

But it turns out that B_i and B_j are not *independent!*[9] *(This error is a prime example of how hard it is to have perfect intuition about probability!)*

10.102 Let $m = 2$, $k = 2$, and $n = 1$. Compute *by hand* the false-positive rate. (*Hint: there are "only"* 16 *different outcomes, each of which is equally likely: the random hash functions assign values in $\{1, 2\}$ to $h_1(x)$, $h_2(x)$, $h_1(y)$, and $h_2(y)$. In each of these 16 cases, determine whether a false positive occurred.*)

10.103 Compute p^2—the answer you would have gotten by using

$$\text{false-positive rate} = (1 - [\text{your answer to Exercise 10.101}])^2.$$

Which is bigger—p^2 or [your answer to Exercise 10.102]? In approximately one paragraph, explain the difference, including an explanation of *why* the events B_1 and B_2 are not independent.

10.104 While the actual false-positive rate is not exactly p^k, it turns out that p^k is a very good approximation to the false-positive rate as long as m is sufficiently big and k is sufficiently small. Write a program that creates a Bloom filter with $m = 1{,}000{,}000$ slots and $k = 20$ hash functions. Insert $n = 100{,}000$ elements, and estimate the false positive probability by querying for n additional uninserted elements $y \notin X$. What is the false-positive rate *that you observe* in your experiment? How does it compare to p^k?

[8] Burton H. Bloom. Space/time tradeoffs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426, July 1970.

[9] Prosenjit Bose, Hua Guo, Evangelos Kranakis, Anil Maheshwari, Pat Morin, Jason Morrison, Michiel Smid, and Yihui Tang. On the false-positive rate of Bloom filters. *Information Processing Letters*, 108(4):210–213, 2008; and Ken Christensen, Allen Roginsky, and Miguel Jimeno. A new analysis of the false positive rate of a Bloom filter. *Information Processing Letters*, 110:944–949, 2010.

10.4 *Random Variables and Expectation*

> Acts of sacrifice, charity and penance are not to be given up but should be performed. . . . All these activities should be performed without any expectation of result.
>
> *Bhagavad Gita* 18:5–6

Thus far, we have been considering *whether or not* something occurs—that is, using the language of probability, we have been interested in *events.* But often we will also be interested in *how many?* questions and not just *did it or did it not?* questions. How many heads came up in 1000 coin flips? How many times do we have to flip a coin before it comes up heads for the 1000th time? For a randomly ordered array $A[1 \ldots n]$ of the integers $\{1, \ldots, n\}$, for how many indices i is $A[i] < A[i+1]$? To address these types of questions, we will introduce the concept of a *random variable*, which measures some numerical quantity that varies from outcome to outcome. We will also consider the *expectation* of a random variable, which is the value of that variable averaged over all of the outcomes in the sample space.

Warning! A "random variable" is one of the worst-named concepts in this entire book. A random variable is not a variable—rather, it's a *function* that maps each outcome to a numerical value. But everyone calls it a random variable, so that's what we'll call it, too.

10.4.1 *Random Variables*

We begin with the definition of a random variable itself:

Definition 10.11 (Random variable)
A random variable X *assigns a numerical value to every outcome in the sample space* S*. (In other words, a random variable is a function* $X : S \to \mathbb{R}$*.)*

Here are a few simple examples:

Example 10.29 (Counting heads in 3 flips)
Suppose that we flip a fair coin independently, three times. (Then the sample space is $S = \{\mathrm{H}, \mathrm{T}\}^3$, and $\Pr[x] = \frac{1}{8}$ for any $x \in S$.) Define the random variables

$$\begin{aligned} X &= \text{the number of heads} \\ Y &= \text{the number of initial consecutive tails.} \end{aligned}$$

These random variables take on the following values:

$$\begin{aligned}
X(\mathrm{HHH}) &= 3 & \qquad Y(\mathrm{HHH}) &= 0 \\
X(\mathrm{HHT}) &= 2 & Y(\mathrm{HHT}) &= 0 \\
X(\mathrm{HTH}) &= 2 & Y(\mathrm{HTH}) &= 0 \\
X(\mathrm{HTT}) &= 1 & Y(\mathrm{HTT}) &= 0 \\
X(\mathrm{THH}) &= 2 & Y(\mathrm{THH}) &= 1 \\
X(\mathrm{THT}) &= 1 & Y(\mathrm{THT}) &= 1 \\
X(\mathrm{TTH}) &= 1 & Y(\mathrm{TTH}) &= 2 \\
X(\mathrm{TTT}) &= 0 & Y(\mathrm{TTT}) &= 3.
\end{aligned}$$

Example 10.30 (Word length, and number of vowels)
Select a word from the sample space $\{\texttt{Now}, \texttt{is}, \texttt{the}, \texttt{winter}, \texttt{of}, \texttt{our}, \texttt{discontent}\}$ by choosing word w with probability proportional to the number of letters in w, as in Example 10.5. Define a random variable L to denote the number of letters in the word chosen. Thus $L(\texttt{discontent}) = 10$ and $L(\texttt{winter}) = 6$, for example. We can also define a random variable V to denote the number of *vowels* in the word chosen. Thus $V(\texttt{discontent}) = 3$ and $V(\texttt{winter}) = 2$, for example. Here are the values for these two random variables for each outcome in the sample space:

w	$\texttt{Pr}[w]$	$L(w)$	$V(w)$
`Now`	3/29	3	1
`is`	2/29	2	1
`the`	3/29	3	1
`winter`	6/29	6	2
`of`	2/29	2	1
`our`	3/29	3	2
`discontent`	10/29	10	3

Although it's an abuse of notation, often we just write X to denote the value of a random variable X *for a realization chosen according to the probability distribution* `Pr`. (So we might write "$X = 3$ with probability $\frac{1}{8}$" or "there are L letters in the chosen word.")

We can state probability questions about events based on random variables, as the following example illustrates:

Example 10.31 (More word length and vowel counts)
Choose a word as in Example 10.30. Define L as the number of letters in the word, and define V as the number of vowels in the word. Then $\texttt{Pr}[L = 3]$ denotes the probability that we choose an outcome w for which $L(w) = 3$. (In other words, $L = 3$ denotes the event $\{w : L(w) = 3\}$.) Thus (see the table in Example 10.30)

$$\begin{aligned} \texttt{Pr}[L = 3] &= \texttt{Pr}\left[\{\texttt{Now}, \texttt{the}, \texttt{our}\}\right] &= \tfrac{9}{29} \\ \texttt{Pr}[V = 3] &= \texttt{Pr}\left[\{\texttt{discontent}\}\right] &= \tfrac{10}{29}. \end{aligned}$$

We will also abuse notation by performing arithmetic on random variables (remember, these are functions!): for two random variables X and Y, we write $X + Y$ as a new random variable that, for any outcome x, denotes the sum of $X(x)$ and $Y(x)$. We will interpret similarly any other arithmetic expression that involves random variables. (The notational analogue here is writing "sin + cos" to denote the function $f(x) = \sin(x) + \cos(x)$.) Here's a small example:

Example 10.32 (Number of consonants)
We can express the number of consonants in the randomly chosen word from our running example (see Example 10.30) as $L - V$. For example, $L - V = 1$ when the chosen word is `our`, and $L - V = 4$ when the chosen word is `winter`.

Indicator random variables

One special type of random variable that will come up frequently is an *indicator random variable,* which only takes on the values 0 and 1. (Such a random variable "indicates" whether a particular event has occurred.) Here's a simple example:

Example 10.33 (Indicator random variables in coin flips)
Suppose that we flip three fair coins independently. Let X_1 be an indicator random variable that reports whether the first flip came up heads. Similarly, let X_2 and X_3 be indicator random variables for the second and third flips. Then:

outcome	X_1	X_2	X_3
HHH	1	1	1
HHT	1	1	0
HTH	1	0	1
HTT	1	0	0
THH	0	1	1
THT	0	1	0
TTH	0	0	1
TTT	0	0	0

Note that the *total number of heads* is given by the random variable $X_1 + X_2 + X_3$.

Independence of random variables

Just as with independence for events, we will often be concerned with whether knowing the value of one random variable tells us something about the value of another. Two random variables X and Y are *independent* if every two events of the form "$X = x$" and "$Y = y$" are independent: for every value x and y, it must be the case that $\Pr\left[X = x \text{ and } Y = y\right] = \Pr\left[X = x\right] \cdot \Pr\left[Y = y\right]$. For example:

Example 10.34 (Some independent/dependent random variables)
The random variables X_2 and X_3 from Example 10.33—we flip 3 fair coins independently; X_2 and X_3 indicate whether the 2nd and 3rd flips are heads—are independent. You can check all four possibilities; for example,

$$\Pr\left[X_2 = 1 \text{ and } X_3 = 1\right] = \tfrac{1}{4} = \tfrac{1}{2} \cdot \tfrac{1}{2} = \Pr\left[X_2 = 1\right] \cdot \Pr\left[X_3 = 1\right] \text{ and}$$
$$\Pr\left[X_2 = 1 \text{ and } X_3 = 0\right] = \tfrac{1}{4} = \tfrac{1}{2} \cdot \tfrac{1}{2} = \Pr\left[X_2 = 1\right] \cdot \Pr\left[X_3 = 0\right].$$

On the other hand, the random variables X and Y from Example 10.29—we flip 3 fair coins independently; X is the number of heads and Y is the number of consecutive initial tails—are not independent; for example,

$$\Pr\left[X = 3\right] \cdot \Pr\left[Y = 3\right] = \tfrac{1}{8} \cdot \tfrac{1}{8} \qquad \text{but} \qquad \Pr\left[X = 3 \text{ and } Y = 3\right] = 0.$$

10.4.2 *Expectation*

A random variable X measures a numerical quantity that varies from realization to realization. We will often be interested in the "average" value of X, which is otherwise known as the random variable's *expectation:*

Definition 10.12 (Expectation)
The expectation *of a random variable X, denoted $\mathtt{E}[X]$, is the average value of X, defined as*

$$\mathtt{E}[X] = \sum_{x \in S} X(x) \cdot \mathtt{Pr}[x].$$

The expectation of X is also sometimes called the mean *of X.*

We can equivalently write $\mathtt{E}[X] = \sum_y \left(y \cdot \mathtt{Pr}\left[X = y\right]\right)$ by summing over each possible value y that X can take on, rather than by summing over outcomes.

The alternate version of the summation for expectation in Definition 10.12 follows by collecting together each outcome x that has the same value of the random variable $X(x)$:

$$\begin{aligned}
&\sum_{x \in S} X(x) \cdot \mathtt{Pr}[x] \\
&= \sum_{y \in \mathbb{R}} \sum_{\substack{x \in S: \\ X(x) = y}} y \cdot \mathtt{Pr}[x] \\
&= \sum_{y \in \mathbb{R}} y \cdot \sum_{\substack{x \in S: \\ X(x) = y}} \mathtt{Pr}[x] \\
&= \sum_{y \in \mathbb{R}} y \cdot \mathtt{Pr}\left[X = y\right].
\end{aligned}$$

In other words, $\mathtt{E}[X]$ is the average value of X over all outcomes (where the average is weighted, with weights defined by the probability function). For example:

Example 10.35 (Expectation of a Bernoulli random variable)
Let X be an indicator random variable for a Bernoulli trial with parameter p—that is, $X = 1$ with probability p and $X = 0$ with probability $1 - p$. Then $\mathtt{E}[X]$ is precisely

$$\begin{aligned}
\mathtt{E}[X] &= 1 \cdot \mathtt{Pr}[X = 1] + 0 \cdot \mathtt{Pr}[X = 0] && \textit{definition of expectation (alternative version)} \\
&= 1 \cdot p\ + 0 \cdot (1 - p) && \textit{definition of a Bernoulli trial with parameter } p \\
&= p.
\end{aligned}$$

Example 10.36 (Counting heads in 3 flips, again)
Problem: Recall Example 10.29, where the random variable X denotes the number of heads in three independent flips of a fair coin. (The sample space was $S = \{\mathrm{H}, \mathrm{T}\}^3$, and $\mathtt{Pr}[x] = \frac{1}{8}$ for any $x \in S$.) What is $\mathtt{E}[X]$?

Solution: The expectation of X is

$$\begin{aligned}
\mathtt{E}[X] &= \sum_{x \in \{\mathrm{H},\mathrm{T}\}^3} \mathtt{Pr}[x] \cdot X(x) \\
&= \tfrac{1}{8} X(\mathrm{HHH}) + \tfrac{1}{8} X(\mathrm{HHT}) + \tfrac{1}{8} X(\mathrm{HTH}) + \tfrac{1}{8} X(\mathrm{HTT}) \\
&\qquad + \tfrac{1}{8} X(\mathrm{THH}) + \tfrac{1}{8} X(\mathrm{THT}) + \tfrac{1}{8} X(\mathrm{TTH}) + \tfrac{1}{8} X(\mathrm{TTT}) \\
&= \tfrac{1}{8} \cdot \left[3 + 2 + 2 + 1 + 2 + 1 + 1 + 0\right] \\
&= \tfrac{12}{8} = 1.5.
\end{aligned}$$

In other words, in three flips of a fair coin, we expect 1.5 flips to come up Heads.

Warning! Just because $\mathtt{E}[X] = 1.5$ doesn't mean that $\mathtt{Pr}[X = 1.5]$ is big! (If you ever flip three fair coins and see exactly 1.5 heads, it might be a sign that the world is ending.) Remember that "average" and "typical" aren't the same thing!

Example 10.37 (Counting letters and vowels, again)

Recall the probabilistic process of choosing a word from the sentence `Now is the winter of our discontent` in proportion to word length. Recall also the random variables from Example 10.30: L denotes the chosen word's length, and V the number of vowels in the chosen word. (See Figure 10.24 for a reminder.) Then we have

$$\begin{aligned} \mathrm{E}\left[L\right] &= 3\cdot\tfrac{3}{29} + 2\cdot\tfrac{2}{29} + 3\cdot\tfrac{3}{29} + 6\cdot\tfrac{6}{29} + 2\cdot\tfrac{2}{29} + 3\cdot\tfrac{3}{29} + 10\cdot\tfrac{10}{29} \\ &= \tfrac{171}{29} \\ &\approx 5.8966. \end{aligned}$$

$$\begin{aligned} \mathrm{E}\left[V\right] &= 1\cdot\tfrac{3}{29} + 1\cdot\tfrac{2}{29} + 1\cdot\tfrac{3}{29} + 2\cdot\tfrac{6}{29} + 1\cdot\tfrac{2}{29} + 2\cdot\tfrac{3}{29} + 3\cdot\tfrac{10}{29} \\ &= \tfrac{57}{29} \\ &\approx 1.9656. \end{aligned}$$

outcome	Pr	L	V
`Now`	$\frac{3}{29}$	3	1
`is`	$\frac{2}{29}$	2	1
`the`	$\frac{3}{29}$	3	1
`winter`	$\frac{6}{29}$	6	2
`of`	$\frac{2}{29}$	2	1
`our`	$\frac{3}{29}$	3	2
`discontent`	$\frac{10}{29}$	10	3

Figure 10.24: A reminder of the sample space, probabilities, and random variables for Example 10.37.

Taking it further: If we think about it without a great deal of care, there's something apparently curious about the result from Example 10.37. We've plopped down our thumb on a random letter in the sentence `Now is the winter of our discontent`, and we've computed that the word that our thumb lands on has an average length of about 5.9 letters. That seems a little puzzling, because there are 7 words in the sentence, with an average word length of $\frac{29}{7} = 4.1428$ letters. But there's a good reason for this discrepancy: *longer words are more likely to be chosen* because they have more letters, and therefore the average word that's chosen has more letters than average. An analogous phenomenon occurs in many other settings, too. When you're driving, you spend most of your time on longer-than-average trips. Most people in Canada live in a larger-than-average-sized Canadian city. Most 3rd-grade students in California are in a larger-than-average-size 3rd-grade class. (In fact, this broader phenomenon is sometimes called the *class-size paradox.*) Perhaps even more jarringly, a random person x knows fewer people than the average number of people known by someone x knows—that is, on average, your friends are more popular than you are.[10] (Why? A very popular person—call her Oprah—is, by definition, the friend of many people, and therefore Oprah's astronomical popularity is averaged into the popularity of many people x. In computing the popularity of a randomly chosen person x, Oprah only contributes her popularity once for $x = $ Oprah—but she contributes it many times to the popularity of x's friends.)

This phenomenon may illustrate an example of a *sampling bias,* in which we try to draw a uniform sample from a population but we end up with some kind of bias that overweights some members of the population at the expense of others. Sampling biases are a widespread concern in any statistical approach to understanding a population. For example, consider a telephone-based political poll that collects voters' preferences for candidates one evening by randomly dialing phone numbers until somebody answers, and records the answerer's preference. This poll will overweight those people who are sitting around at home during the evening—which correlates with the voter's age, which correlates with the voter's political affiliation.

Example 10.38 (Number of aces in a bridge hand)

Problem: Suppose that we are dealt a 13-card hand from a standard 52-card deck. What is the expected number of aces in our hand?

Solution: Later we'll solve this problem more easily (see Example 10.41), but here we'll do it the hard way. We'll compute the probability of getting $0, 1, \ldots, 4$ aces:

- There are $\binom{52}{13}$ different hands.
- There are $\binom{4}{k} \cdot \binom{48}{13-k}$ hands with exactly k aces. (We have to pick k ace cards from the 4 aces in the deck, and $13 - k$ non-ace cards from the 48 non-aces.)

[10] Scott L. Feld. Why your friends have more friends than you do. *American Journal of Sociology*, 96(6):1464–1477, May 1991.

Because each hand is equally likely to be chosen, therefore

$$\texttt{Pr}\left[\text{drawing exactly } k \text{ aces}\right] = \frac{\binom{4}{k} \cdot \binom{48}{13-k}}{\binom{52}{13}}.$$

And thus, letting A be a random variable denoting the number of aces, we have

$$\begin{aligned}
\texttt{E}\left[A\right] &= \sum_{h} \texttt{Pr}\left[\text{being dealt hand } h\right] \cdot (\text{number of aces in } h) \\
&= \sum_{i=0}^{4} i \cdot \texttt{Pr}\left[A = i\right] \qquad \textit{(reordering sum by collecting all hands with the same number of aces)} \\
&= \frac{\overbrace{0 \cdot \binom{4}{0} \cdot \binom{48}{13}}^{0\cdot\texttt{Pr}[A=0]} + \overbrace{1 \cdot \binom{4}{1} \cdot \binom{48}{12}}^{1\cdot\texttt{Pr}[A=1]} + \overbrace{2 \cdot \binom{4}{2} \cdot \binom{48}{11}}^{2\cdot\texttt{Pr}[A=2]} + \overbrace{3 \cdot \binom{4}{3} \cdot \binom{48}{10}}^{3\cdot\texttt{Pr}[A=3]} + \overbrace{4 \cdot \binom{4}{4} \cdot \binom{48}{9}}^{4\cdot\texttt{Pr}[A=4]}}{\binom{52}{13}} \\
&= \frac{0 \cdot 1 \cdot \binom{48}{13} + 1 \cdot 4 \cdot \binom{48}{12} + 2 \cdot 6 \cdot \binom{48}{11} + 3 \cdot 4 \cdot \binom{48}{10} + 4 \cdot 1 \cdot \binom{48}{9}}{\binom{52}{13}} \\
&= \frac{0 + 278{,}674{,}137{,}872 + 271{,}142{,}404{,}416 + 78{,}488{,}590{,}752 + 6{,}708{,}426{,}560}{635{,}013{,}559{,}600} \\
&= \frac{635{,}013{,}559{,}600}{635{,}013{,}559{,}600} \\
&= 1.
\end{aligned}$$

That is, the expected number of aces in a 13-card hand is precisely 1.

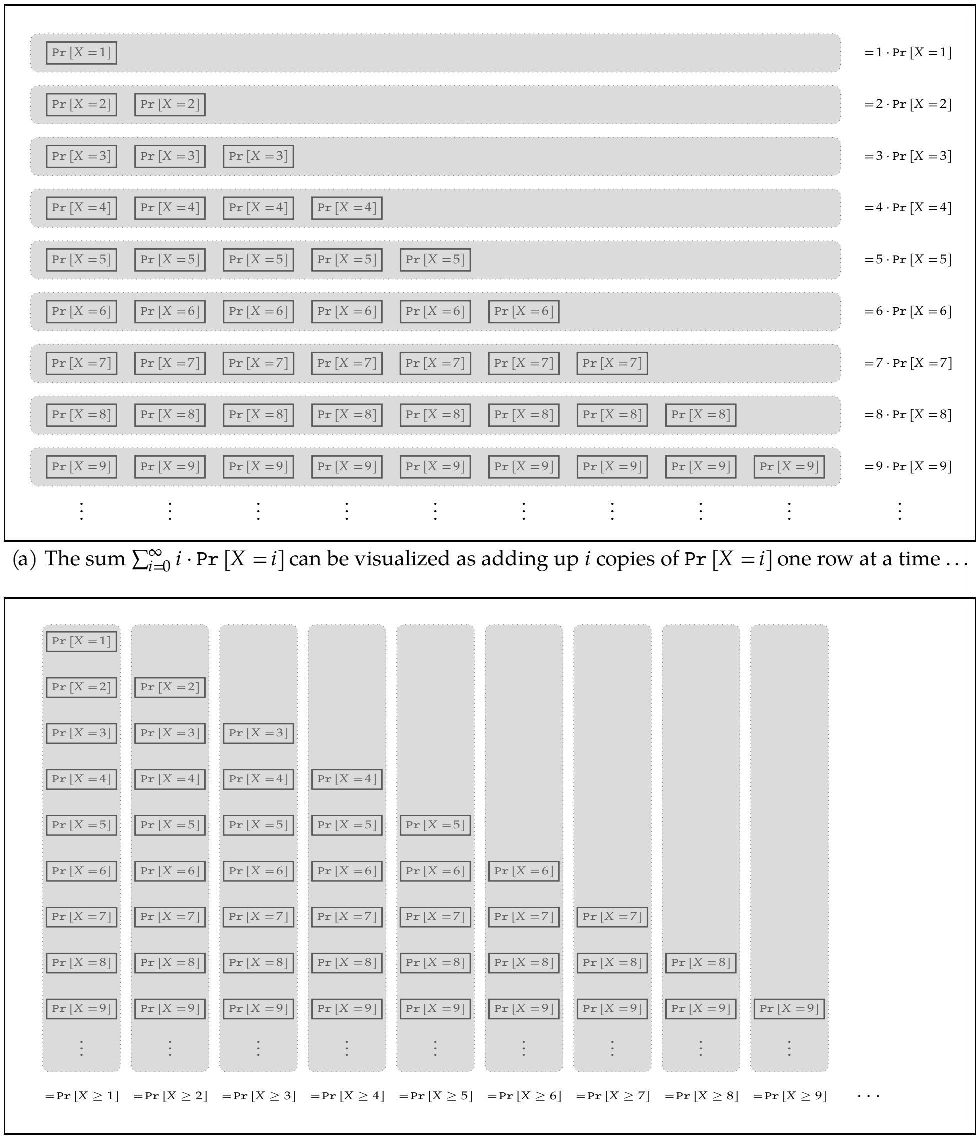
A useful property of expectation

We've now seen several examples of computing the expectation of random variables by directly following the definition of expectation. Here we'll introduce a transformation that can often make expectation calculations easier, at least for positive integer–valued random variables:

Theorem 10.5 (A new formula for expectation, for nonnegative integers)
Let $X : S \to \mathbb{Z}^{\geq 0}$ be a random variable. Then $\texttt{E}\left[X\right] = \sum_{i=1}^{\infty} \texttt{Pr}\left[X \geq i\right]$.

(Note that by definition $\texttt{E}\left[X\right] = \sum_{i=0}^{\infty} i \cdot \texttt{Pr}\left[X = i\right]$, so we're trading the multiplication of i for the replacement of $=$ by $\geq$.)

The proof will follow by changing the order of summation in the expectation formula. We'll give an algebraic proof in a moment, but it may be easier to follow the idea by looking at a visualization first. See Figure 10.25.



(a) The sum $\sum_{i=0}^{\infty} i \cdot \texttt{Pr}\left[X = i\right]$ can be visualized as adding up i copies of $\texttt{Pr}\left[X = i\right]$ one row at a time . . .

(b) . . . or by visualizing adding up one *column* at a time. The jth column contains a copy of $\texttt{Pr}\left[X = i\right]$ for every i greater than or equal to j, so the value of the sum of the jth column is $\texttt{Pr}\left[X \geq j\right]$.

Figure 10.25: A change of summation. View $E[X] = \sum_{i=0}^{\infty} i \cdot \texttt{Pr}\left[X = i\right]$ as the sum of the entries of an infinite table, where the ith row of the table contains i copies of $\texttt{Pr}\left[X = i\right]$. By computing column sums instead of row sums, we see $\sum_{i=0}^{\infty} i \cdot \texttt{Pr}\left[X = i\right] = \sum_{j=1}^{\infty} \texttt{Pr}\left[X \geq j\right]$.

Proof of Theorem 10.5. We proceed using the manipulation from Figure 10.25:

$$
\begin{aligned}
\texttt{E}\left[X\right] &= \sum_{i=0}^{\infty} i \cdot \texttt{Pr}\left[X = i\right] && \textit{definition of expectation} \\
&= \sum_{i=0}^{\infty} \sum_{j=1}^{i} \texttt{Pr}\left[X = i\right] && i = \textstyle\sum_{j=1}^{i} 1 \\
&= \sum_{j=1}^{\infty} \sum_{i=j}^{\infty} \texttt{Pr}\left[X = i\right] && \textit{changing the order of summation (see Figure 10.25)} \\
&= \sum_{j=1}^{\infty} \texttt{Pr}\left[X \geq j\right]. \qquad \square && \texttt{Pr}\left[X \geq j\right] = \textstyle\sum_{i=j}^{\infty} \texttt{Pr}\left[X = i\right]
\end{aligned}
$$

We can use this theorem to find the expected value of a geometric random variable:

Example 10.39 (Expectation of a geometric random variable)
Let X be a geometric random variable with parameter p. (That is, X measures the number of flips of a p-biased coin before we get Heads for the first time.) Then $\mathtt{E}\left[X\right]$ is precisely $\frac{1}{p}$:

$$\begin{aligned}
\mathtt{E}\left[X\right] &= \sum_{i=1}^{\infty} \mathtt{Pr}\left[X \geq i\right] && \textit{Theorem 10.5 } (\mathtt{E}\left[X\right] = \textstyle\sum_{i=1}^{\infty} \mathtt{Pr}\left[X \geq i\right]) \\
&= \sum_{i=1}^{\infty} \mathtt{Pr}\left[\text{fail to get heads in } i-1 \text{ flips}\right] && \textit{definition of geometric random variable} \\
&= \sum_{i=1}^{\infty} (1-p)^{i-1} && \textit{need } i-1 \textit{ consecutive tails flips} \\
&= \sum_{i=0}^{\infty} (1-p)^{i} && \textit{changing index of summation} \\
&= \frac{1}{1-(1-p)} = \frac{1}{p}. && \textit{formula for geometric summations}
\end{aligned}$$

For example, we expect to flip a fair coin (with $p = \frac{1}{2}$) *twice* before we get heads.

10.4.3 *Linearity of Expectation*

Here's a very useful general property of expectation, called *linearity of expectation*: the expectation of a sum is the sum of the expectations. (A *linear function* is a function f that satisfies $f(a+b) = f(a) + f(b)$—for example, $f(x) = 3x$ or $f(x) = 0$.) The usefulness of Linearity of Expectation will come from the way in which it lets us "break down" a complicated random variable into the sum of a collection of simple random variables. (We can then compute $\mathtt{E}\left[\text{Complicated}\right] = \mathtt{E}\left[\sum_i \text{Simple}_i\right] = \sum_i \mathtt{E}\left[\text{Simple}_i\right]$.)

We'll see several useful examples soon, but let's start with the proof:

Theorem 10.6 (Linearity of Expectation)
Consider a sample space S, and let $X : S \to \mathbb{R}$ and $Y : S \to \mathbb{R}$ be any two *random variables. Then $\mathtt{E}\left[X+Y\right] = \mathtt{E}\left[X\right] + \mathtt{E}\left[Y\right]$.*

Proof. We'll be able to prove this theorem by just invoking the definition of expectation and following our algebraic noses:

$$\begin{aligned}
\mathtt{E}\left[X+Y\right] &= \sum_{s \in S} (X+Y)(s) \cdot \mathtt{Pr}\left[s\right] && \textit{definition of expectation} \\
&= \sum_{s \in S} \left[X(s) + Y(s)\right] \cdot \mathtt{Pr}\left[s\right] && \textit{definition of the random variable } X+Y \\
&= \left[\sum_{s \in S} X(s) \cdot \mathtt{Pr}\left[s\right]\right] + \left[\sum_{s \in S} Y(s) \cdot \mathtt{Pr}\left[s\right]\right] && \textit{distributing the multiplication; rearranging} \\
&= E[X] + E[Y]. && \textit{definition of expectation}
\end{aligned}$$

Therefore $\mathtt{E}\left[X+Y\right] = \mathtt{E}\left[X\right] + \mathtt{E}\left[Y\right]$, as desired. □

Notice that Theorem 10.6 does *not* impose any requirement of independence on the random variables X and Y: even if X and Y are highly correlated (positively or negatively), we *still* can use linearity of expectation to conclude that $\mathtt{E}[X+Y] = \mathtt{E}[X] + \mathtt{E}[Y]$. There are many apparently complicated problems in which using linearity of expectation makes a solution totally straightforward. Here are a few examples:

Example 10.40 (Expectation of a binomial random variable)

Problem: We have a p-biased coin (that is, $\mathtt{Pr}[\text{heads}] = p$) that we flip 1000 times. What is the expected number of heads that come up in these 1000 flips?

Solution: The intuition is fairly straightforward: a p-fraction of flips are heads, so we should expect $1000p$ heads in 1000 flips. But doing the math requires a bit of work.

An abandoned first attempt: Let's compute the probability that there are exactly k heads in a sequence of 1000 flips, and then apply the definition of expectation directly. There are $\binom{1000}{k}$ sequences of 1000 flips that have exactly k heads, and the probability of any one of these sequences is $p^k(1-p)^{1000-k}$, so

$$\begin{aligned}
&\mathtt{E}[\text{number of heads}] \\
&= \sum_{k=0}^{1000} k \cdot \mathtt{Pr}[\text{number of heads} = k] && \textit{definition of expectation} \\
&= \sum_{k=0}^{1000} k \cdot \binom{1000}{k} \cdot p^k \cdot (1-p)^{1000-k}. && \textit{above analysis of } \mathtt{Pr}[\textit{number of heads} = k]
\end{aligned}$$

We could try to simplify this expression (but it turns out to be pretty hard!). Instead, let's start over with a different approach.

A second try: Here's a strategy that ends up being much easier. Define 1000 random variables $X_1, X_2, \ldots, X_{1000}$, where X_i is the indicator random variable

$$X_i = \begin{cases} 1 & \text{if the } i\text{th flip of the coin comes up Heads} \\ 0 & \text{if the } i\text{th flip of the coin comes up Tails.} \end{cases}$$

The total number of heads in the 1000 coin flips is given by the random variable

$$X = X_1 + X_2 + \cdots + X_{1000}.$$

We can use this definition of X and linearity of expectation to compute the expected number of heads much more easily:

$$\begin{aligned}
\mathtt{E}[\text{number of heads}] = \mathtt{E}[X] = \mathtt{E}\left[\sum_{i=1}^{1000} X_i\right] & && \textit{definition of } X \\
= \sum_{i=1}^{1000} \mathtt{E}[X_i] & && \textit{linearity of expectation} \\
= \sum_{i=1}^{1000} p & && \textit{Example 10.35 (expectation of a Bernoulli variable)} \\
= 1000p. &
\end{aligned}$$

Problem-solving tip: Often, the easiest way to compute an expectation is by finding a way to express the quantity of interest in terms of a sum of indicator random variables.

Example 10.41 (Number of aces in a bridge hand, better)
Recall Example 10.38, where we showed that the number A of aces in a randomly chosen 13-card hand from a standard 52-card deck has $\mathtt{E}[A] = 1$. Here is a *much* easier way of solving that problem:

Number your cards from 1 to 13. Let A_i be an indicator random variable that reports whether the ith card in your hand is an ace. Then $A = A_1 + A_2 + \ldots + A_{13}$. Note that $\mathtt{Pr}[A_i = 1] = \frac{1}{13}$ (there are $\frac{4}{52} = \frac{1}{13}$ aces in the deck), so

$$\begin{aligned}
\mathtt{E}[A] &= \mathtt{E}[A_1 + A_2 + \cdots + A_{13}] \\
&= \mathtt{E}[A_1] + \mathtt{E}[A_2] + \cdots + \mathtt{E}[A_{13}] && \textit{linearity of expectation} \\
&= 13 \cdot \tfrac{1}{13} && \mathtt{Pr}[A_i = 1] = \tfrac{1}{13} \textit{ as above, and so } \mathtt{E}[A_i] = \tfrac{1}{13} \textit{ (Example 10.35)} \\
&= 1.
\end{aligned}$$

(The random variables A_i and A_j are correlated—but, again, linearity of expectation doesn't care! We can still use it to conclude that $\mathtt{E}\left[A_i + A_j\right] = \mathtt{E}[A_i] + \mathtt{E}\left[A_j\right]$.)

Some examples about hashing

Here are two more problems about expectation, both involving hashing:

Example 10.42 (Hashing)
Problem: Suppose that we hash 1000 elements into a 1000-slot hash table, using a completely random hash function, resolving collisions by chaining. (See Section 10.1.1.) How many empty slots do we expect?

Solution: Let's compute the probability that some particular slot is empty:

$$\begin{aligned}
&\mathtt{Pr}\left[\text{slot } i \text{ is empty}\right] \\
&= \mathtt{Pr}\left[\text{none of the 1000 elements hash to slot } i\right] \\
&= \mathtt{Pr}\left[\text{every element } j \in \{1, 2, \ldots, 1000\} \text{ hashes to a slot other than } i\right] \\
&= \prod_{j=1}^{1000} \mathtt{Pr}\left[\text{element } j \text{ hashes to a slot other than } i\right] && \textit{elements are hashed independently} \\
&= \prod_{j=1}^{1000} \frac{999}{1000} && \textit{elements are hashed uniformly, and there are 999 other slots} \\
&= \left(\frac{999}{1000}\right)^{1000} = 0.3677\cdots.
\end{aligned}$$

We'll finish with the by-now-familiar calculation that also concluded the last two examples: we define a collection of indicator random variables and use linearity of expectation. Let X_i be an indicator random variable that's 1 if slot i is empty and 0 if slot i is full. Then the expected number of empty slots is

$$\mathrm{E}\left[\sum_{i=1}^{1000} X_i\right] = \sum_{i=1}^{1000} \mathrm{E}\left[X_i\right] = 1000 \cdot \left(\tfrac{999}{1000}\right)^{1000} \approx 367.7.$$

Taking it further: If we stated the question from Example 10.42 in full generality, we would ask: *if we hash n elements into n slots, how many empty slots are there in expectation?* Using the same approach as in Example 10.42, we'd find that the fraction of empty slots is, in expectation, $(1 - 1/n)^n$. Using calculus, it's possible to show that $(1 - 1/n)^n$ approaches $1/e \approx 0.367879$ as $n \to \infty$. So, for large n, we'd expect to have $\frac{n}{e}$ empty slots when we hash n elements into n slots.

We can also turn this hashing problem on its head: we've been asking "if we hash n elements into n slots, how many slots do we expect to find empty?" Instead we can ask "how many elements do we expect have to hash into n slots before all n slots are full?" This problem is called the *coupon-collector problem*; see Exercises 10.136–10.137 for more.

Let's also consider a second example about hashing—this time counting the (expected) number of collisions, rather than the (expected) number of empty slots:

Example 10.43 (Expected collisions in a hash table)

Problem: Hash n elements $A = \{x_1, \ldots, x_n\}$ into an m-slot hash table. Recall that a *collision* between two elements x_i and x_j (for $i \neq j$) occurs when $h(x_i) = h(x_j)$.

1. Consider two elements $x_i \neq x_j$. What's $\Pr$ [there's a collision between x_i and x_j]?
2. What is the expected number of collisions among the elements of A?

Solution: 1. A collision between x_i and x_j occurs precisely when, for some index k, we have $h(x_i) = k$ *and* $h(x_j) = k$. Thus:

$$\begin{aligned}
&\Pr\left[\text{collision between } x_i \text{ and } x_j\right] \\
&= \Pr\left[\left[h(x_i) = h(x_j) = 1\right] \text{ or } \left[h(x_i) = h(x_j) = 2\right] \text{ or } \cdots \text{ or } \left[h(x_i) = h(x_j) = m\right]\right] \\
&= \sum_{k=1}^{m} \Pr\left[h(x_i) = k \text{ and } h(x_j) = k\right] && \textit{by the sum rule; these events are disjoint} \\
&= \sum_{k=1}^{m} \Pr\left[h(x_i) = k\right] \cdot \Pr\left[h(x_j) = k\right] && \textit{hashing assumption: hash values are independent} \\
&= \sum_{k=1}^{m} \frac{1}{m} \cdot \frac{1}{m} && \textit{hashing assumption: hash values are uniform} \\
&= \frac{m}{m^2} = \frac{1}{m}.
\end{aligned}$$

So the probability that a particular pair of elements collides is precisely $\frac{1}{m}$.

2. Given (1), we can again compute the expected number of collisions using indicator random variables and linearity of expectation. The number of collisions between elements of A is precisely the number of unordered pairs $\{x_i, x_j\}$ that collide. For indices i and $j > i$, then, define $X_{i,j}$ as the indicator random variable

$$X_{i,j} = \begin{cases} 1 & \text{if } x_i \text{ and } x_j \text{ collide} \\ 0 & \text{if they do not.} \end{cases}$$

Thus the expected number of collisions among the elements of A is given by

$$\begin{aligned}
&\mathrm{E}\left[\sum_{1 \le i < j \le n} X_{i,j} \right] && \textit{summing over all unordered pairs of elements} \\
&= \sum_{1 \le i < j \le n} \mathrm{E}\left[X_{i,j}\right] && \textit{linearity of expectation} \\
&= \sum_{1 \le i < j \le n} \frac{1}{m} && \textit{part 1 of this example: we showed } \mathrm{E}\left[X_{i,j}\right] = \Pr\left[X_{i,j} = 1\right] = \tfrac{1}{m} \\
&= \frac{\binom{n}{2}}{m} = \frac{n(n-1)}{2m}. && \textit{there are } \tbinom{n}{2} = \tfrac{n(n-1)}{2} \textit{ unordered pairs of elements}
\end{aligned}$$

One consequence of this analysis is that we'd expect the first collision in an m-slot hash table to occur when the number n of hashed elements reaches approximately $\sqrt{2m}$: for $n = \sqrt{2m} + 1$, the expected number of collisions would be

$$\frac{n(n-1)}{2m} = \frac{(\sqrt{2m}+1) \cdot \sqrt{2m}}{2m} \approx \frac{2m}{2m} = 1.$$

Taking it further: Example 10.43 also explains the so-called *birthday paradox*. Assume that a person's birthday is uniformly and independently chosen from the $m = 365$ days in the year. (Close, but not quite true; certain parts of the year are nine months before days whose probabilities are notably more than $\frac{1}{365}$.) Under this assumption, you can think of "birthday" as a random hash function mapping people to $\{1, 2, \ldots, 365\}$. By Example 10.43, if you're in a room with more than $\sqrt{2 \cdot 365} = 27.018$ people, you'd expect to find a pair that shares a birthday. (It's called a "paradox" because most people's intuition is that you'd need way more than 28 people in a room before you'd find a shared birthday.)

Two more examples of expectation: breaking PINs and Insertion Sort

Here's another example of expectation, in a simple security context:

Example 10.44 (Brute-force breaking of PINs)

Problem: I steal a debit card from a (former) friend. The card has a 4-digit PIN, between 0000 and 9999, that I need to know to get all my friend's money. Here are two strategies:

1. every day, I try a random PIN.
2. every day, I try a random PIN *that I haven't tried before.*

How many days would I expect to wait before I get into my friend's account?

Solution: 1. Observe that the probability of getting the correct PIN on a particular day is $\frac{1}{10000}$. Thus we have a geometric random variable with parameter $\frac{1}{10000}$, so by Example 10.39 we expect to need 10000 days to break the PIN.

2. As usual, there are multiple ways to solve this problem—and, for illustrative purposes, we'll describe two of them, using fairly different approaches.

Solution A: $\texttt{Pr}\left[\textbf{winning on day \#}i\right]$**.** The key will be to find the probability of breaking the code for the first time on day i. (For the purposes of this analysis, imagine that we keep guessing new PINs even after we find the correct answer.)

Because we make $i-1$ guesses on the $i-1$ days before day i, we know

$$\texttt{Pr}\left[\text{getting the PIN } \textit{before} \text{ day } i\right] = \tfrac{i-1}{10000} \tag{1}$$

$$\texttt{Pr}\left[\underline{\text{not}} \text{ getting the PIN } \textit{before} \text{ day } i\right] = 1 - \tfrac{i-1}{10000} = \tfrac{10001-i}{10000}. \tag{2}$$

Furthermore, on day i there are $10000-(i-1)$ untried guesses, and so

$$\texttt{Pr}\left[\text{getting the PIN } \textit{on} \text{ day } i \mid \underline{\text{not}} \text{ getting it } \textit{before} \text{ day } i\right] = \tfrac{1}{10000-(i-1)} = \tfrac{1}{10001-i}. \tag{3}$$

Thus the expected number of days that we have to keep guessing is:

$$\begin{aligned}
&\sum_{i=1}^{10000} i \cdot \texttt{Pr}\left[\text{we first break the code on day \#i}\right] && \textit{definition of expectation}\\
&= \sum_{i=1}^{10000} i \cdot \texttt{Pr}\left[\text{wrong on days } 1,\ldots,i-1\right] \\
&\qquad\qquad \cdot \texttt{Pr}\left[\text{right on day } i | \text{wrong on days } 1,\ldots,i-1\right] && \textit{Chain Rule}\\
&= \sum_{i=1}^{10000} i \cdot \tfrac{10001-i}{10000} \cdot \tfrac{1}{10001-i} && \textit{(2) and (3), as argued above}\\
&= \tfrac{1}{10000} \cdot \sum_{i=1}^{10000} i && \textit{algebra}\\
&= \tfrac{1}{10000} \cdot \tfrac{10000 \cdot 10001}{2} = 5000.5. && \textit{arithmetic summation (Example 5.4)}
\end{aligned}$$

(Another way to view this solution: our PIN-guessing strategy corresponds to choosing a permutation of $\{0000,\ldots,9999\}$ uniformly at random, and guessing in the chosen order. The correct PIN is equally likely to be at any position in the permutation so, for any i, we require exactly i days with probability precisely $\frac{1}{10000}$.)

Solution B: $\texttt{Pr}\left[\textbf{have to guess on day \#}i\right]$**.** Define an indicator random variable X_i, where $X_i = 1$ if we have to make a guess on day $\#i$, and $X_i = 0$ if we do not. Thus the number of days that we have to guess is precisely $X := \sum_{i=1}^{10000} X_i$. Observe that

$$\texttt{E}\left[X_i\right] = \texttt{Pr}\left[X_i\right] = \texttt{Pr}\left[\text{lose on days } 1,\ldots,i-1\right] = \tfrac{10001-i}{10000}$$

by the same reasoning as in Solution A. Thus

$$\begin{aligned}
\mathtt{E}\left[X\right] &= \sum_{i=1}^{10000} \mathtt{E}\left[X_i\right] && \textit{linearity of expectation} \\
&= \sum_{i=1}^{10000} \frac{10001-i}{10000} && \textit{the above argument} \\
&= \sum_{j=1}^{10000} \frac{j}{10000} && \textit{change of variables } j = 10001 - i \\
&= 5000.5. && \textit{just as in Solution A}
\end{aligned}$$

So avoiding duplication saves, in expectation, just less than half of the days: we expect to use 10000 days if we allow duplication, and 5000.5 days if we avoid it.

(Incidentally, the argument in Solution B is just another way of viewing the transformation from Theorem 10.5: instead of calculating the value of $\sum_i i \cdot \mathtt{Pr}\left[\text{exactly } i \text{ days}\right]$, we calculated $\sum_i \mathtt{Pr}\left[\text{at least } i \text{ days}\right]$.)

Let's conclude with one last example of another type: analyzing the expected performance of an algorithm on a randomly chosen input. In Example 6.13, we gave a brief intuition for the average-case (expected) performance of Insertion Sort. (See Figure 10.26 for a reminder of the algorithm.) Here is a somewhat different version of the analysis, which comes out with the same result:

```
insertionSort($A[1 \ldots n]$):
  for $i := 2$ to $n$:
      $j := i$
      while $j > 1$ and $A[j] < A[j-1]$:
          swap $A[j]$ and $A[j-1]$
          $j := j - 1$
```

Figure 10.26: A reminder of Insertion Sort.

Example 10.45 (Expected performance of Insertion Sort)

Problem: Let the array A be a permutation of $\{1, \ldots, n\}$ chosen uniformly at random. What is the expected number of swaps performed by **insertionSort**($A[1 \ldots n]$)?

Solution: Define an indicator random variable $X_{j,i}$ for indices $j < i$:

$$X_{j,i} = \begin{cases} 1 & \text{if the (original) elements } A[j] \text{ and } A[i] \text{ are swapped by } \textbf{insertionSort} \\ 0 & \text{if not.} \end{cases}$$

Note that $\mathtt{E}\left[X_{j,i}\right] = \mathtt{Pr}\left[X_{j,i} = 1\right] = \frac{1}{2}$: precisely half of permutations have their ith element larger than their jth element. (There's a bijection between the set of permutations with their ith element larger than their jth element and the set of permutations with their ith element smaller than their jth element. Because these sets have the same size, the probability of choosing one of the former is $\frac{1}{2}$.)

Because **insertionSort** correctly sorts its input and only swaps out-of-order pairs once per pair, the total number of swaps done is precisely

$$X = \sum_{i=2}^{n} \sum_{j=1}^{i-1} X_{i,j}.$$

Note that the number of indicator random variables in this sum is

$$\sum_{i=2}^{n}\sum_{j=1}^{i-1} 1 = \sum_{i=2}^{n}(i-1) = \sum_{i=1}^{n-1} i = \tfrac{(n-1)\cdot n}{2} = \tbinom{n}{2}.$$

Thus by linearity of expectation we have

$$\mathtt{E}\left[X\right] = \tbinom{n}{2} \cdot \mathtt{E}\left[X_{i,j}\right] = \tbinom{n}{2} \cdot \tfrac{1}{2}.$$

10.4.4 Conditional Expectation

Just as we did with conditional probability in Section 10.3, we can define a notion of *conditional expectation:* that is, the average value of a random variable X *when a particular event occurs.*

Definition 10.13 (Conditional expectation)
The conditional expectation *of a random variable X given an event E, denoted $\mathtt{E}\left[X|E\right]$, is the average value of X over all outcomes where E occurs:*

$$\mathtt{E}\left[X|E\right] = \sum_{x \in E} X(x) \cdot \mathtt{Pr}\left[x|E\right].$$

In the original definition of expectation, we summed over all x in the whole sample space; here we sum only over the outcomes in the event E. Furthermore, here we weight the value of X by $\mathtt{Pr}\left[x|E\right]$ rather than by $\mathtt{Pr}\left[x\right]$. We'll omit the details, but conditional expectation has analogous properties to those of the original (nonconditional) version of expectation, including linearity of expectation.

Here's a brief example of computing some conditional expectations:

Example 10.46 (Hearts in Poker)

Problem: In Texas Hold 'Em, a particular variant of poker, after a standard deck of cards is randomly shuffled, you are dealt two "personal" cards, and then five "community" cards are dealt. Let P denote the number of your personal cards that are hearts, and let C denote the number of community cards that are hearts. What are the following?

1. $\mathtt{E}\left[P\right]$
2. $\mathtt{E}\left[C\right]$
3. $\mathtt{E}\left[C|P=0\right]$
4. $\mathtt{E}\left[C|P=2\right]$

Solution: 1 & 2. Each card that's dealt has a $\frac{13}{52} = \frac{1}{4}$ chance of being a heart. By linearity of expectation, then, $\mathtt{E}\left[P\right] = \frac{2}{4} = 0.5$ and $\mathtt{E}\left[C\right] = \frac{5}{4} = 1.25$. (Implicitly, we're defining indicator random variables for "the ith card is a heart," so $P = P_1 + P_2$ and $C = C_1 + \cdots + C_5$.)

3. Given that 2 of the 39 non-heart cards were dealt as your personal cards, there are still 13 undealt hearts among the remaining 50 undealt cards. Thus there is a $\frac{13}{50} = 0.26$ chance that any particular undealt card is a heart. Thus, again by linearity of expectation, we have that $\mathtt{E}\left[C|P=0\right] = 5 \cdot \frac{13}{50} = 1.30$.
4. Similarly, there are 11 undealt hearts among the remaining 50 undealt cards. Thus there is an $\frac{11}{50} = 0.22$ chance that any particular undealt card is a heart, and $\mathtt{E}\left[C|P=2\right] = 5 \cdot \frac{11}{50} = 1.10$.

We'll omit the proof, but it's worth noting a useful property that connects expectation to conditional expectation, an analogy to the law of total probability:

Theorem 10.7 (Law of Total Expectation)
For any random variable X and any event E:

$$\mathtt{E}\left[X\right] = \mathtt{E}\left[X|E\right] \cdot \mathtt{Pr}\left[E\right] + \mathtt{E}\left[X|\overline{E}\right] \cdot (1 - \mathtt{Pr}\left[E\right]).$$

That is, the expectation of X is the (weighted) average of the expectation of X when E occurs and when E does not occur.

> **Taking it further:** One tremendously valuable use of probability is in *randomized algorithms,* which flip some coins as part of solving some problem. There is a massive variety in the ways that randomization is used in these algorithms, but one example—the computation of the *median* element of an unsorted array of numbers—is discussed on p. 1060. (We'll make use of Theorem 10.7.) Median finding is a nice example of problem for which there is a very simple, efficient algorithm that makes random choices in its solution. (There *are* deterministic algorithms that solve this problem just as efficiently, but they are *much* more complicated than this randomized algorithm.)

10.4.5 Deviation from Expectation

Let X be a random variable. By definition, the value of $\mathtt{E}\left[X\right]$ is the average value that X takes on, where we're averaging over many different realizations. But how far away from $\mathtt{E}\left[X\right]$ is X, on average? That is, what is the average difference between (a) X, and (b) the average value of X? We might care about this quantity in applications like political polling or scientific experimentation, for example. Suppose X is a random variable defined as follows:

$$X = \begin{cases} -1 & \text{the voter will vote for the Democratic candidate} \\ 0 & \text{the voter will vote for neither the Democratic nor Republican candidates} \\ +1 & \text{the voter will vote for the Republican candidate} \end{cases}$$

for a voter chosen uniformly at random from the population. If $\mathtt{E}\left[X\right] < 0$, then the Democrat will beat the Republican in the election; if $\mathtt{E}\left[X\right] > 0$, then the Republican will beat the Democrat. We might estimate $\mathtt{E}\left[X\right]$ by calling, say, 500 uniformly chosen voters from the population and averaging their responses. We'd like to know whether our estimate is accurate (that is, if our estimate is close to $\mathtt{E}\left[X\right]$). This kind of question is the core of statistical reasoning. We'll only begin to touch on these questions, but here are a few of the most important concepts.

Definition 10.14 (Variance)
Let X be a random variable. The variance *of X is*

$$\text{var}(X) = \mathtt{E}\left[(X - \mathtt{E}[X])^2\right].$$

The standard deviation *is* $\text{std}(X) = \sqrt{\text{var}(X)}$.

(Exercise: why didn't we just define $\text{std}(X) = \mathtt{E}[X - \mathtt{E}[X]]$?)

Here's a simple example:

Example 10.47 (Variance/standard deviation of a Bernoulli random variable)
Let X be the outcome of a flipping a p-biased coin. (That is, X is a Bernoulli random variable.) We previously showed that $\mathtt{E}[X] = p$, so the variance of X is

$$\begin{aligned}
\text{var}(X) &= \mathtt{E}[\ (X - \mathtt{E}[X])^2\ ] && \textit{definition of expectation}\\
&= \mathtt{E}[\ (X-p)^2\ ] && \textit{expectation of a Bernoulli random variable (Example 10.35)}\\
&= \mathtt{Pr}[X=0]\cdot(0-p)^2 + \mathtt{Pr}[X=1]\cdot(1-p)^2 && \textit{definition of expectation}\\
&= (1-p)\cdot(0-p)^2 + p\cdot(1-p)^2 && \textit{definition of Bernoulli random variable}\\
&= (1-p)p^2 + p(1-p)^2\\
&= (1-p)p\cdot(p+1-p)\\
&= (1-p)p.
\end{aligned}$$

Thus the standard deviation is $\text{std}(X) = \sqrt{\text{var}(X)} = \sqrt{(1-p)p}$.

(For example, for a fair coin, the standard deviation is $\sqrt{(1-0.5)0.5} = \sqrt{0.25} = 0.5$: an average coin flip is 0.5 units away from the mean 0.5. In fact, every coin flip is that far away from the mean!)

Here's another simple example, illustrating the fact that two random variables can have the same mean but wildly different variances:

Example 10.48 (Roulette bets)
Here are two bets available to a player in roulette (see Figure 10.27 for a reminder):

- Bet \$1 on "red": If the spin lands on one of the 18 red numbers, you get \$2 back; otherwise you get nothing.
- Bet \$1 on "17": If the spin lands on the number 17, you get \$36 back; otherwise you get nothing.

Let X denote the payoff from playing the first bet, so $X = 0$ with probability $\frac{20}{38}$ and $X = 2$ with probability $\frac{18}{38}$. Let Y denote the payoff from playing the second bet, so $Y = 0$ with probability $\frac{37}{38}$ and $X = 36$ with probability $\frac{1}{38}$. The expectations match:

$$\begin{aligned}
\mathtt{E}[X] &= \tfrac{20}{38}\cdot 0 + \tfrac{18}{38}\cdot 2 = \tfrac{36}{38}\\
\mathtt{E}[Y] &= \tfrac{37}{38}\cdot 0 + \tfrac{1}{38}\cdot 36 = \tfrac{36}{38}.
\end{aligned}$$

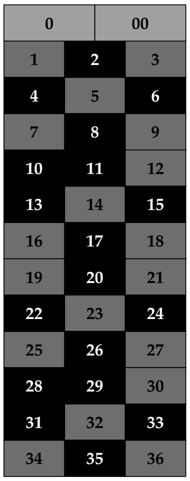


Figure 10.27: A reminder of the roulette outcomes. A number in the set $\{0, 00, 1, 2, \ldots, 36\}$ is chosen uniformly at random by a spinning wheel; there are 18 *red* numbers and 18 *black* numbers; 0 and 00 are neither red nor black.

But the variances are very different:

$$\begin{aligned}\text{var}(X) &= \tfrac{20}{38}\cdot(0-\tfrac{36}{38})^2+\tfrac{18}{38}\cdot(2-\tfrac{36}{38})^2 &= 0.9972\cdots\\ \text{var}(Y) &= \tfrac{37}{38}\cdot(0-\tfrac{36}{38})^2+\tfrac{1}{38}\cdot(36-\tfrac{36}{38})^2 &= 33.2077\cdots.\end{aligned}$$

Generally speaking, the expectation of a random variable measures "how good it is" (on average), while the variance measures "how risky it is."

Variance, the squared expectation, and the expectation of the square

Here's a useful property of variance, which sometimes helps us avoid tedium in calculations. We can write $\text{var}(X)$ as $\text{var}(X) = \mathtt{E}[X^2] - (\mathtt{E}[X])^2$, that is, the difference between the *expectation of the square of* X and the *square of the expectation of* X:

Theorem 10.8 (Variance = expectation of the square minus the expectation^2)
For any random variable X, we have

$$\text{var}(X) = \mathtt{E}\left[X^2\right] - (\mathtt{E}[X])^2.$$

Proof. Writing $\mu := \mathtt{E}[X]$, we have

$$\begin{aligned}&\text{var}(X)\\ &= \mathtt{E}\left[(X-\mu)^2\right] && \textit{definition of expectation}\\ &= \mathtt{E}\left[X^2 - 2X\mu + \mu^2\right] && \textit{multiplying out}\\ &= \mathtt{E}\left[X^2\right] + \mathtt{E}[-2X\mu] + \mathtt{E}\left[\mu^2\right] && \textit{linearity of expectation}\\ &= \mathtt{E}\left[X^2\right] - 2\mu\cdot\mathtt{E}[X] + \mu^2 && \textit{Exercise 10.151}\\ &= \mathtt{E}\left[X^2\right] - 2\mu\cdot\mu + \mu^2 && \textit{definition of } \mu = \mathtt{E}[X]\\ &= \mathtt{E}\left[X^2\right] - \mu^2\\ &= \mathtt{E}\left[X^2\right] - (\mathtt{E}[X])^2. && \square\end{aligned}$$

Here is a simple example in which Theorem 10.8 eases the computation:

Example 10.49 (Variance/standard deviation of a uniform random variable)
Problem: Let X be the result of a roll of a fair die. What is $\text{var}(X)$?

Solution: Because $\mathtt{Pr}[X = k] = \frac{1}{6}$ for all $k \in \{1,\ldots,6\}$, we have that

$$\begin{aligned}\mathtt{E}[X] &= \tfrac{1}{6}\cdot(1+2+3+4+5+6)\\ &= \tfrac{1}{6}\cdot 21\\ &= 3.5.\end{aligned}$$

Similarly, we can compute $\mathtt{E}\left[X^2\right]$ as follows:

$$\begin{aligned}\mathtt{E}\left[X^2\right] &= \tfrac{1}{6} \cdot (1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2) \\ &= \tfrac{1}{6} \cdot 91 \\ &\approx 15.1666 \cdots .\end{aligned}$$

Therefore, by Theorem 10.8,

$$\text{var}\,(X) \;= \mathtt{E}\left[X^2\right] - \left(\mathtt{E}\left[X\right]\right)^2 \;= \tfrac{91}{6} - \tfrac{49}{4} \;= \tfrac{35}{12} \;\approx\; 2.9116 \cdots ,$$

and std $(X) = \sqrt{35/12} \approx 1.7078 \cdots$.

(In Exercise 10.150, you'll show that the standard deviation of the average result of two independent dice rolls is much smaller.)

Taking it further: Suppose that we need to estimate the fraction of [very complicated objects] that have [easy-to-verify property]: would I win a higher fraction of chess games with Opening Move A or B? Roughly how many different truth assignments satisfy Boolean formula φ? Roughly how many integers in $\{2, 3, \ldots, n-1\}$ evenly divide n? Is the array A "mostly" sorted?

One nice way to approximate the answer to these questions is the *Monte Carlo method,* one of the simplest ways to use randomization in computation. The basic idea is to compute many *random* candidate elements—chess games, truth assignments, possible divisors, etc.—and test each one; we can then estimate the answer to the question of interest by calculating the fraction of those random candidates that have the property in question. See p. 1062 for more discussion.

Computer Science Connections

A Randomized Algorithm for Finding Medians

The *median* element of an array $A[1 \ldots n]$ is the item that would appear in the $\lceil n/2 \rceil$th slot of the sorted order if we sorted A. For example, the median of $[1, 3, 5, 7, 9]$ is 5, and the median of $[4, 3, 2, 1]$ is 2. (We arbitrarily chose to find the $\lceil n/2 \rceil$th element instead of the $\lfloor n/2 \rfloor$th.) This description already suggests a solution to the median problem: sort A, and then return $A[\lceil n/2 \rceil]$. But we can do better than the sorting-based approach: we'll give a faster algorithm for finding the median element of an unsorted array. Our algorithm will be randomized, and the *expected* running time of the algorithm will be linear. It will turn out to be easier to solve a generalization of the median problem, called Select. See Figure 10.28.

Select:

Given: an array $A[1 \ldots n]$ and an index $k \in \{1, \ldots, n\}$.

Output: the element x in A such that, if you were to sort A, x would appear in the kth slot of the sorted array.

Figure 10.28: The Select problem.

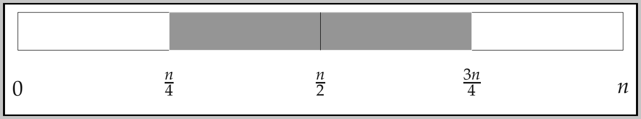
A recursive solution to Select is given in Figure 10.29; we can solve the median problem by calling **randSelect**$(A[1 \ldots n], \lceil n/2 \rceil)$. A proof of correctness of the algorithm—that is, a proof that **randSelect** actually solves the Select problem—is reasonably straightforward by induction. (In fact, correctness is guaranteed *regardless of how we choose x in Line 3* of the algorithm.) But we still have to analyze the running time.

```
randSelect($A[1 \ldots n]$, $i$):
// Find the ith-largest element of $A$.
// If $i \notin \{1, 2, \ldots, n\}$, then error.
1:  if $n = 1$ then
2:      return $A[1]$. (If $i \neq 1$, then error.)
3:  choose $x \in \{1, \ldots, n\}$ randomly
4:  $Losers[1 \ldots \ell] := \{y \in A : y < A[x]\}$
5:  $Winners[1 \ldots w] := \{y \in A : y > A[x]\}$.
6:  if $i < \ell + 1$ then
7:      return randSelect($Losers$, $i$)
8:  else if $i = \ell + 1$ then
9:      return $A[x]$
10: else if $i > \ell + 1$ then
11:     return randSelect($Winners$, $i - \ell - 1$)
```

Figure 10.29: Randomized Median finding. (We build *Losers* and *Winners* by going through A element-by-element.)

Running Time: The Big Picture

Think about an invocation of **randSelect**(A), and imagine the array A in sorted order and divided into quartiles:



Here are two crucial observations:

1. Suppose that the element $A[x]$ chosen in step 3—call $A[x]$ the *pivot*—falls within the shaded region of the quartile picture above. Then we know that *$|Losers| \leq \frac{3n}{4}$ and $|Winners| \leq \frac{3n}{4}$*.
2. The shaded region contains half of the elements of A.

(Why? To put it briefly: because half of the elements of A are in the middle half of the array A.) So what? Let's think intuitively for a moment, and defer the formal analysis. Whenever we choose an element from the middle half of the sorted order, the next recursive call is on an array of size at most $\frac{3}{4}$ the size of the original input. Also observe that the running time of any particular call (aside from the recursive call) is linear in the input size. Thus, if we got lucky every time and picked an element from the middle half of the array, we'd have a recurrence like the following:

$$T(1) = 1 \qquad\qquad T(n) \leq n + T(3n/4)$$

That's a classic Master Method recurrence with a solution of $T(n) = \Theta(n)$. (Actually the master method only says that $T(n) = O(n)$, because we have an inequality in the recurrence. But it's trivial that the running time is $\Omega(n)$ as well, because just building *Losers* and *Winners* at the root takes $\Omega(n)$ time.)

Computer Science Connections

A Randomized Algorithm for Finding Medians, Continued

Running Time: Making it Formal

We engaged in wishful thinking in the last paragraph: it's obviously not true that we get a pivot in the middle half of the array every time. In fact, it's only half the time! But this isn't so bad: *even if we imagine that picking a pivot outside the middle half yields zero progress at all toward the base case,* we'd only double the estimate of the running time! Let's make this formal. Define

$C_n :=$ the number of comparisons performed by **randSelect** on an input of size n.

Notice that C_n is a random variable: the number of comparisons that are performed depends on which pivots are chosen! But we can analyze $\mathtt{E}[C_n]$.

Before we start, let's make one quick observation: the expected running time of this algorithm is monotonic in its input size. That is, $\mathtt{E}[C_n] \leq \mathtt{E}[C_{n'}]$ if $n \leq n'$. (This fact is tedious to prove rigorously, but is still pretty obvious.)

Theorem: $\mathtt{E}[C_n] \leq 8n$.

Proof (by strong induction on n). Base case ($n = 1$): In fact, when $n = 1$, the algorithm performs zero comparisons, and indeed $0 \leq 8$.

Inductive case ($n \geq 2$): We assume the inductive hypothesis, namely that for any $n' < n$, we have that $\mathtt{E}\left[C'_n\right] \leq 8n'$. We must prove that $\mathtt{E}[C_n] \leq 8n$.

Let's consider the comparisons that are made on an input array of size n. First, there are n comparisons performed in Lines 4–5, to compute *Losers* and *Winners*. Then there are whatever comparisons are made in the recursive call. Because we're trying to compute a worst-case bound, we'll make do with the following observation: $C_n \leq n + \mathcal{C}_{\max(|Losers|,|Winners|)}$.

Let $\mathcal{M}$ denote the event that our pivot is in the middle half of A (that is, falls in the shaded region of the diagram on the previous page). Thus:

$$\begin{aligned}
\mathtt{E}[C_n] &\leq \mathtt{E}\left[n + \mathcal{C}_{\max(|Losers|,|Winners|)}\right] && \textit{the above accounting of the comparisons}\\
&= n + \mathtt{E}\left[\mathcal{C}_{\max(|Losers|,|Winners|)}\right] && \textit{linearity of expectation}\\
&= n + \mathtt{E}\left[\mathcal{C}_{\max(|Losers|,|Winners|)} | \mathcal{M}\right] \cdot \mathtt{Pr}[\mathcal{M}] + \mathtt{E}\left[\mathcal{C}_{\max(|Losers|,|Winners|)} | \overline{\mathcal{M}}\right] \cdot \mathtt{Pr}\left[\overline{\mathcal{M}}\right]\\
& && \textit{Law of Total Expectation (Theorem 10.7)}\\
&= n + \tfrac{1}{2} \cdot \left[\mathtt{E}\left[\mathcal{C}_{\max(|Losers|,|Winners|)} | \mathcal{M}\right] + \mathtt{E}\left[\mathcal{C}_{\max(|Losers|,|Winners|)} | \overline{\mathcal{M}}\right]\right] && \textit{Crucial observation \#2: } \mathtt{Pr}[\mathcal{M}] = \mathtt{Pr}\left[\overline{\mathcal{M}}\right] = \tfrac{1}{2}\\
&\leq n + \tfrac{1}{2} \cdot \left[\mathtt{E}\left[C_{3n/4}\right] + \mathtt{E}[C_n]\right]. && \textit{Crucial observation \#1: if } \mathcal{M} \textit{ occurs, we recurse on} \leq \tfrac{3n}{4} \textit{ elements; else it's certainly on} \leq n \textit{ elements.}
\end{aligned}$$

Thus we have argued that

$$\begin{aligned}
\mathtt{E}[C_n] &\leq n + \tfrac{1}{2} \cdot \mathtt{E}\left[C_{3n/4}\right] + \tfrac{1}{2} \cdot \mathtt{E}[C_n] \text{ and therefore}\\
\mathtt{E}[C_n] &\leq 2n + \mathtt{E}\left[C_{3n/4}\right]. && \textit{starting with the previous inequality and subtracting } \tfrac{1}{2} \cdot \mathtt{E}[C_n] \textit{ from both sides, and then multiplying both sides by 2}
\end{aligned}$$

The inductive hypothesis says that $\mathtt{E}\left[C_{3n/4}\right] \leq 8 \cdot \frac{3n}{4} = 6n$, so we therefore have

$$\mathtt{E}[C_n] \leq 2n + 6n = 8n. \qquad \square$$

Computer Science Connections

The Monte Carlo Method

If we need to compute some (potentially very complicated) quantity, one way to do so is the *Monte Carlo method*. Let's take a computation of area of a potentially complicated shape as an example. If we identify a bounding box (a rectangle surrounding the shape) and then generate a sequence of random points in the bounding box, we can count how many of those points fall into the shape in question.

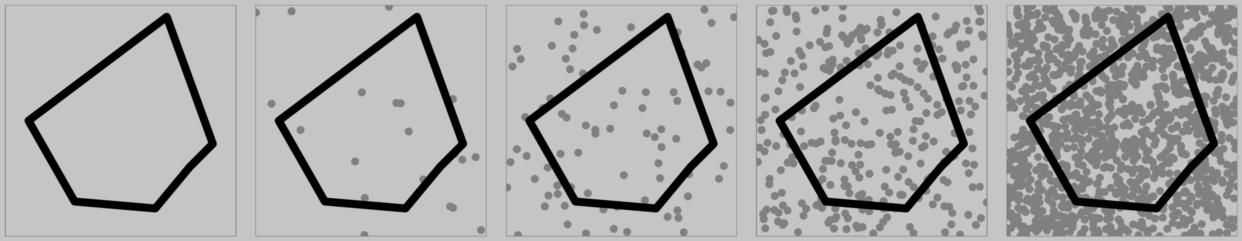


Figure 10.30: A shape, and an estimate of its area with random points: we simply estimate the area using the fraction of the chosen points that fall within the shape. The more points, the more accurate the estimate.

For example, to find the area of the shape in Figure 10.30, we can throw a random point into the bounding box. The probability that the randomly chosen point is inside the polygon is precisely the ratio of the area of the polygon to the area of the bounding box—and thus the expected fraction of points that land inside the shape precisely yields the area of the shape. Of course, the more points we throw at the bounding box, the more accurate our estimate of the area will be: the fraction of heads in n flips of a p-biased coin has a much lower variance (but the same expectation) as n gets bigger and bigger. (See Exercise 10.155.)

There are a few issues complicating this approach. First, we must find a bounding box for which the shape in question covers a "large" fraction of the bounding box. (If the probability p of a random point falling into the shape is tiny, then a little bad luck in sampling—2 points land inside instead of 3?—causes huge relative [multiplicative] error in our area estimate.) Second, we've described this process as choosing a uniform point from the bounding box—which requires infinitesimal probabilities associated with each of the infinitely many points inside the bounding box. The handle this, typically we would define a "mesh" of points: we specify a "resolution" ε and choose a coordinate of a random point as k/ε for a random $k \in \{0, 1, \ldots, 1/\varepsilon\}$.

The example in Figure 10.30 is a nice way of being lazy—we *could* have calculated the area of the polygon with some tedious algebra—but there are some other examples in which this technique is even more useful. Some of the simplest methods for estimating the value of π in the last century were based on Monte Carlo methods. One option is to throw a point $\langle x, y \rangle$ into the unit square $[0,1] \times [0,1]$ and test what fraction have $x^2 + y^2 \leq 1$. Another is an algorithm called *Buffon's needle*—named after an 18th-century French mathematician—in which we throw unit-length "needles" onto a surface with parallel lines one unit apart; one can show that $\texttt{Pr}\left[\text{a needle crosses a line}\right] = \frac{2}{\pi}$. See Figure 10.31.

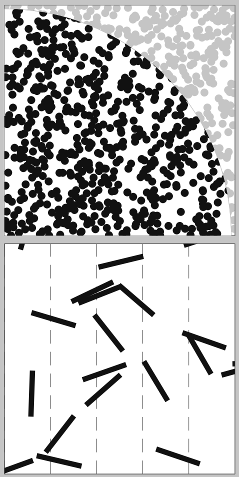


Figure 10.31: Estimating π with a point in the unit square, or with Buffon's needle.

10.4.6 Exercises

Choose a word in $S = \{\texttt{Computers}, \texttt{are}, \texttt{useless}, \texttt{They}, \texttt{can}, \texttt{only}, \texttt{give}, \texttt{you}, \texttt{answers}\}$ *(a quote attributed to Pablo Picasso) by choosing a word* w *with probability proportional to the number of letters in* w*. Let* L *be a random variable denoting the number of letters in the chosen word, and let* V *be a random variable denoting the number of vowels.*

10.105 Give a table of outcomes and their probabilities, together with the values of L and V.

10.106 What is $\texttt{Pr}[L = 4]$? What is $\texttt{E}\left[V | L = 4\right]$?

10.107 Are L and V independent?

10.108 What are $\texttt{E}[L]$ and $\texttt{E}[V]$?

10.109 What is $\text{var}(L)$?

10.110 What is $\text{var}(V)$?

Flip a fair coin 16 *times. Define the following two random variables:*

- *let* H *be an indicator random variable that's* 1 *if at least one of the* 16 *flips comes up heads, and* 0 *otherwise.*
- *let* R *be a random variable equal to the length of the longest "run" in the flips. (A* run *of length* k *is a sequence of* k *consecutive flips that all come up Heads, or* k *consecutive flips that all come up Tails.)*

10.111 What's $\texttt{E}[H]$?

10.112 What's $\texttt{E}[R]$? (*Hint: write a program*—not *by simulating many sequences of* 16 *coin flips, but rather by listing exhaustively all outcomes.*)

10.113 Are H and R independent?

In 1975, a physicist named Michael Winkelmann invented a dice-based game with the following three (fair) dice:

Blue die: *sides* 1, 2, 5, 6, 7, 9 **Red die:** *sides* 1, 3, 4, 5, 8, 9 **Black die:** *sides* 2, 3, 4, 6, 7, 8

There are some weird properties of these dice, as you'll see.

10.114 Choose one of the three dice at random, roll it, and call the result X. Show that $\texttt{Pr}[X = k] = \frac{1}{9}$ for any $k \in \{1, \ldots, 9\}$.

10.115 Choose one of the three dice at random, roll it, and call the result X. Put that die back in the pile and again (independently) choose one of the three dice at random, roll it, and call the result Y. Show that $\texttt{Pr}[9X - Y = k] = \frac{1}{81}$ for any $k \in \{0, \ldots, 80\}$.

10.116 Roll each die. Call the results B (blue), R (red), and K (black). Compute $\texttt{E}[B]$, $\texttt{E}[R]$, and $\texttt{E}[K]$.

10.117 Define B, R, and K as in the last exercise. Compute $\texttt{Pr}\left[B > R | B \neq R\right]$, $\texttt{Pr}\left[R > K | R \neq K\right]$, and $\texttt{Pr}\left[K > B | K \neq B\right]$—in particular, show that all three of these probabilities (strictly) exceed $\frac{1}{2}$.

The last exercise demonstrates that the red, blue, and black dice are nontransitive, *using the language of relations (Chapter 8): you'd bet on Blue beating Red and you'd bet on Red beating Black, but (surprisingly) you'd want to bet on Black beating Blue. Here's another, even weirder, example of nontransitive dice. (And if you're clever and mildly unscrupulous, you can win some serious money in bets with your friends using these dice.)*

Kelly die: *sides* 3, 3, 3, 3, 3, 6 **Lime die:** *sides* 2, 2, 2, 5, 5, 5 **Mint die:** *sides* 1, 4, 4, 4, 4, 4

These dice are fair; each side comes up with probability $\frac{1}{6}$*. Roll each die, and call the resulting values* K*,* L*, and* M*.*

10.118 Show that the expectation of each of these three random variables is identical.

10.119 Show that $\texttt{Pr}[K > L]$, $\texttt{Pr}[L > M]$, and $\texttt{Pr}[M > K]$ are all strictly greater than $\frac{1}{2}$.

You can think of the last exercise as showing that, if you had to bet on which of K *or* L *would roll a higher number, you should bet on* K*. (And likewise for* L *over* M*, and for* M *over* K*.) Now let's think about rolling each die* twice *and adding the two rolled values together. Roll each die* twice*, and call the resulting values* K_1*,* K_2*,* L_1*,* L_2*,* M_1*, and* M_2*, respectively.*

10.120 Show that the expectation of the three values $K_1 + K_2$, $L_1 + L_2$, and $M_1 + M_2$ are identical.

10.121 (*programming required*) Show that the following probabilities are all strictly *less* than $\frac{1}{2}$:

$$\texttt{Pr}[K_1 + K_2 > L_1 + L_2], \texttt{Pr}[L_1 + L_2 > M_1 + M_2], \text{ and } \texttt{Pr}[M_1 + M_2 > K_1 + K_2].$$

(Notice that which die won switched directions—and all we did was go from rolling the dice once to rolling them twice!) To show this result, write a program to check how many of the 6^4 outcomes cause $K_1 + K_2 > L_1 + L_2$, etc.

Suppose that you are dealt a 5*-card hand from a standard deck. For the purposes of the next two questions, a* pair *consists of any two cards with the same rank—so* ♣A♡A♢A23 *contains* three *pairs* (♡A♢A *and* ♣A♢A *and* ♣A♡A)*. Let* P *denote the number of pairs in your hand.*

10.122 Compute $\texttt{E}[P]$ "the hard way," by computing $\texttt{Pr}[P = 0]$, $\texttt{Pr}[P = 1]$, $\texttt{Pr}[P = 2]$, and so forth. (There can be as many as 6 pairs in your hand, if you have four-of-a-kind.)

10.123 Compute $\texttt{E}[P]$ "the easy way," by defining an indicator random variable $R_{i,j}$ that's 1 if and only if cards #i and #j are a pair, computing $\texttt{E}\left[R_{i,j}\right]$, and using linearity of expectation.

In bridge, you are dealt a 13*-card hand from a standard deck. A hand's* high-card points *are awarded for face cards:* 4 *for an ace,* 3 *for a king,* 2 *for a queen, and* 1 *for a jack. A hand's* distribution points *are awarded for having a* small *number of cards in a particular suit:* 1 *point for a "doubleton" (only two cards in a suit),* 2 *points for a "singleton" (only one card in a suit), and* 3 *points for a "void" (no cards in a suit).*

10.124 What is the expected number of high-card points in a bridge hand? (*Hint: define some simple random variables, and use linearity of expectation.*)

10.125 What is the expected number of distribution points *for hearts* in a bridge hand? (*Hint: calculate the probability of having exactly* 2 *hearts, exactly* 1 *heart, or no hearts in a hand.*)

10.126 Using the results of the last two exercises and linearity of expectation, find the expected number of points (including both high-card and distribution points) in a bridge hand.

We've shown linearity of expectation—the expectation of a sum equals the sum of the expectations—even when the random variables in question aren't independent. It turns out that the expectation of a product equals the product of the expectations when the random variables are independent, but not in general when they're dependent.

10.127 Let X and Y be independent random variables. Prove that $\mathrm{E}[X \cdot Y] = \mathrm{E}[X] \cdot \mathrm{E}[Y]$.

On the other hand, suppose that X and Y are dependent *random variables. Prove that . . .*

10.128 . . . $\mathrm{E}[X \cdot Y]$ is not necessarily equal to $\mathrm{E}[X] \cdot \mathrm{E}[Y]$.

10.129 . . . $\mathrm{E}[X \cdot Y]$ is also not necessarily *unequal* to $\mathrm{E}[X] \cdot \mathrm{E}[Y]$.

We showed in Example 10.39 that the expected number of flips of a p-biased coin before we get Heads is precisely $\frac{1}{p}$.

10.130 How many flips would you expect to have to make before you see 1000 heads *in total* (not necessarily consecutive)? (*Hint: define a random variable* X_i *denoting the number of coin flips* after the $(i-1)$st Heads *before you get another Heads. Then use linearity of expectation.*)

10.131 How many flips would you expect to make before you see two *consecutive* heads?

In Insertion Sort, we showed in Example 10.45 that the expected number of swaps is $\binom{n}{2}/2$ *for a randomly sorted input. With respect to* comparisons, *it's fairly easy to see that each element participates in one more comparison than it does swap—with one exception: those elements that are swapped all the way back to the beginning of the array. Here you'll precisely analyze the expected number of comparisons.*

10.132 What is the probability that the ith element of the array is swapped all the way back to the beginning of the array?

10.133 What's the expected number of comparisons done by Insertion Sort on a randomly sorted n-element input?

```
insertionSort(A[1 . . . n]):
for i := 2 to n:
    j := i
    while j > 1 and A[j] < A[j − 1]:
        swap A[j] and A[j − 1]
        j := j − 1
```

Figure 10.32: A reminder of Insertion Sort.

Suppose we hash n elements into an 100,000*-slot hash table, resolving collisions by chaining.*

10.134 Use Example 10.43 to identify the smallest n for which the expected number of collisions first reaches 1. What the smallest n for which the expected number of collisions exceeds 100,000?

10.135 (*programming required*) Write a program to empirically test your answers from the last exercise, by doing $k = 1000$ trials of loading [*your first answer from Exercise 10.134*] elements into a 100,000-slot hash table. Also do $k = 100$ trials of loading [*your second answer from Exercise 10.134*] elements. On average, how many collisions did you see?

Consider an m-slot hash table that resolves collisions by chaining. In the next few problems, we'll figure out the expected number of elements that must be hashed into this table before every *slot is "hit"—that is, until every cell of the hash table is full.*

10.136 Suppose that the hash table currently has $i - 1$ filled slots, for some number $i \in \{1, \ldots, m\}$. What is the probability that the next element that's hashed falls into an *unoccupied* slot? Let the random variable X_i denote the number of elements that are hashed *until one more cell is filled.* What is $\mathrm{E}[X_i]$?

10.137 Argue that the total number X of elements hashed before the entire hash table is full is given by $X = \sum_{i=1}^{m} X_i$. Using Exercise 10.136 and linearity of expectation, prove that $\mathrm{E}[X] = m \cdot H_m$.

(*Recall that* H_m *denotes the mth* harmonic number, *where* $H_m := \sum_{i=1}^{m} \frac{1}{i}$*. See Definition 5.4.*)

The problem you've addressed in the last two exercises is called the coupon collector problem *among computer scientists: imagine, say, a cereal company that puts one of n coupons into each box of cereal that it sells, choosing which coupon type goes into each box randomly. How many boxes of cereal must a serial cereal eater buy before he collects a complete set of the n coupons?*

True story: some nostalgic friends and I were trying to remember all of the possible responses on a Magic 8 Ball, a pseudopsychic toy that reveals one of 20 *answers uniformly at random when it's shaken—things like*

$$\{\text{ask again later, signs point to yes, don't count on it, } \ldots\}.$$

We found a toy shop with a Magic 8 Ball in stock and started asking it questions. We hoped to have learned all 20 *different answers before we got kicked out of the store.*

10.138 What is the probability that we'd get 20 different answers in our first 20 trials?

10.139 In expectation, how many trials would we need before we found all 20 answers? (Use the result on coupon collecting from Exercise 10.137.)

In Exercise 10.139, you determined the number of trials that, on average, *are necessary to get all* 20 *answers. But how* likely *are we to succeed with a certain number of trials?*

10.140 Suppose we perform 200 trials. What is the probability that a *particular* answer (for example, "ask again later") was never revealed in any of those 200 trials?

10.141 Use the Union Bound (Exercise 10.37) and the previous exercise to argue that the probability that we need more than 200 trials to see all 20 answers is less than 0.1%.

10.142 Suppose that one random bit in a 32-bit number is corrupted (that is, flipped from 0 to 1 or from 1 to 0). What is the expected size of the error (thinking of the change of the value in binary)? What about for a random bit in an n-bit number?

10.143 Suppose that the numbers $\{1, \ldots, n\}$ are randomly ordered—that is, we choose a random permutation π of $\{1, \ldots, n\}$. For a particular index i, what is the probability that $\pi_i = i$—that is, the ith biggest element is in the ith position?

10.144 Let X be a random variable denoting the number of indices i for which $\pi_i = i$. What is $\mathtt{E}[X]$? (*Hint: define indicator random variables and use linearity of expectation.*)

Markov's inequality *states that, for a random variable X that is always nonnegative (that is, for any x in the sample space, we have $X(x) \geq 0$), the following statement is true, for any $\alpha \geq 1$:*

$$\mathtt{Pr}[X \geq \alpha] \leq \frac{\mathtt{E}[X]}{\alpha}.$$

Markov's inequality is named after Andrey Markov, a 19th-to-20th-century Russian mathematician. A number of other important ideas in probability are also named after him, like Markov processes, Hidden Markov models, and more.

10.145 Prove Markov's inequality. (*Hint: use conditional expectation.*)

10.146 The *median* of a random variable X is a value x such that

$$\mathtt{Pr}[X \leq x] \geq \tfrac{1}{2} \qquad \text{and} \qquad \mathtt{Pr}[X \geq x] \geq \tfrac{1}{2}.$$

Using Markov's inequality, prove that the median of a nonnegative random variable X is at most $2 \cdot \mathtt{E}[X]$.

Take a fair coin, and repeatedly flip it until it comes up heads. Let K be a random variable indicating the number of flips performed. (We've already shown that $\mathtt{E}[K] = 2$, in Example 10.39.) You are offered a chance to play a gambling game, for the low low price of y dollars to enter. A fair coin will be flipped until it comes up heads, and you will be paid $(3/2)^K$ dollars if K flips were required. (So there's a $\frac{1}{2}$ chance that you'll be paid \$1.50 because the first flip comes up heads; a $\frac{1}{4}$ chance that you'll be paid \$2.25 $= (1.50)^2$ because the first flip comes up tails and the second comes up heads, and so forth.)

10.147 Assuming that you care *only* about expected value—that is, you're willing to play if and only if $\mathtt{E}[(3/2)^K] \geq y$—then what value of y is the break-even point? (In other words, what is $\mathtt{E}[(3/2)^K]$?)

10.148 Let's sweeten the deal slightly: you'll be paid 2^K dollars if K flips are required. Assuming that you still care *only* about expected value, then what value of y is the break-even point? (*Be careful!*)

10.149 Let X be the number of heads flipped in 4 independent flips of a fair coin. What is $\text{var}(X)$?

10.150 Let Y be the average of two independent rolls of a fair die. What is $\text{var}(Y)$?

10.151 Let $a \in \mathbb{R}$, and let X be a random variable. Prove that $\mathtt{E}[a \cdot X] = a \cdot \mathtt{E}[X]$.

10.152 Let $a \in \mathbb{R}$, and let X be a random variable. Prove that $\text{var}(a \cdot X) = a^2 \cdot \text{var}(X)$.

10.153 Prove that $\text{var}(X + Y) = \text{var}(X) + \text{var}(Y)$ for two independent random variables X and Y. (*Hint: use Exercise 10.127.*)

10.154 Let X be a random variable following a binomial distribution with parameters n and p. (That is, X is the number of heads found in n flips of a p-biased coin.) Using Exercise 10.153 and the logic as in Example 10.40, show that $\mathtt{E}[X] = np$ and $\text{var}(X) = np(1-p)$.

10.155 Flip a p-biased coin n times, and let Y be a random variable denoting the *fraction* of those n flips that came up heads. What are $\mathtt{E}[Y]$ and $\text{var}(Y)$?

In the next few exercises, you'll find the variance of a geometric random variable. This derivation will require a little more work than the result from Exercise 10.154 (about the variance of a binomial random variable); in particular, we'll need a preliminary result about summations first:

10.156 (*Calculus required.*) Prove the following two formulas, for any real number r with $0 \le r < 1$:

$$\sum_{i=0}^{\infty} ir^i = \frac{r}{(1-r)^2} \qquad\qquad \sum_{i=0}^{\infty} i^2 r^i = \frac{r(1+r)}{(1-r)^3}.$$

(*Hint: use the geometric series formula* $\sum_{i=0}^{n} r^i = \frac{r^{n+1}-1}{r-1}$ *from Theorem 5.2, differentiate, and take the limit as n grows. Repeat for the second derivative.*)

10.157 Let X be a geometric random variable with parameter p. (That is, X denotes the number of flips of a p-biased coin we need before we see heads for the first time.) What is $\text{var}(X)$? (*Hint: compute both* $\mathtt{E}[X]^2$ *and* $\mathtt{E}[X^2]$. *The previous exercise will help with at least one of those computations.*)

Recall from Chapter 3 that a proposition is in 3-conjunctive normal form (3CNF) *if it is the conjunction of* clauses, *where each clause is the disjunction of three different variables/negated variables. For example,*

$$(\neg p \vee q \vee r) \wedge (\neg q \vee \neg r \vee x)$$

is in 3CNF. Recall further that a proposition φ is satisfiable *if it's possible to give a truth assignment for the variables of φ to true/false so that φ itself turns out to be true. We've previously discussed that it is believed to be computationally very difficult to determine whether a proposition φ is satisfiable (see p. 326)—and it's believed to be very hard to determine whether φ is satisfiable* even if φ is in 3CNF. *But you'll show here an easy way to satisfy "most" clauses of a proposition φ in 3CNF, using randomization.*

10.158 Let φ be a proposition in 3CNF. Consider a *random truth assignment* for φ—that is, each variable is set independently to True with probability $\frac{1}{2}$. Prove that a particular clause of φ is true under this truth assignment with probability $\geq \frac{7}{8}$.

10.159 Suppose that φ has m clauses and n variables. Prove that the *expected* number of satisfied clauses under a random truth assignment is at least $\frac{7m}{8}$.

10.160 Prove the following general statement about any random variable: $\mathtt{Pr}[X \geq \mathtt{E}[X]] > 0$. (*Hint: use conditional expectation.*) Then, using this general fact and Exercise 10.159, argue that, for any 3CNF proposition φ, *there exists a truth assignment that satisfies at least $\frac{7}{8}$ of φ's clauses.*

> **Taking it further:** One can also show that there's a very good chance—at least $8/m$—that a random truth assignment satisfies at least $7m/8$ clauses, and therefore we expect to find such a truth assignment within $m/8$ random trials. This algorithm is called *Johnson's algorithm*, named after the researcher David Johnson; for details of this and other randomized algorithms for satisfiability, see a good book on randomized algorithms.[11]

[11] Michael Mitzenmacher and Eli Upfal. *Probability and computing: randomized algorithms and probabilistic analysis*. Cambridge University Press, 2005; Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995; and Jon Kleinberg and Éva Tardos. *Algorithm Design*. Addison–Wesley, 2006.

10.5 Chapter at a Glance

Probability, Outcomes, and Events

Imagine a process by which some quantities of interest are determined in some random way. An *outcome*, or *realization,* of this probabilistic process is the sequence of results for all randomly determined quantities. The *sample space* S is the set of all possible outcomes. A *probability function* $\texttt{Pr} : S \to \mathbb{R}$ describes, for each outcome $s \in S$, the fraction of the time that s occurs. The probability function Pr must satisfy two conditions: (i) $\sum_{s \in S} \texttt{Pr}[s] = 1$, and (ii) $\texttt{Pr}[s] \geq 0$ for every $s \in S$.

An *event* is a subset of S, and the *probability of an event* E, written $\texttt{Pr}[E]$, is the sum of the probabilities of the outcomes contained in E. We have that $\texttt{Pr}[S] = 1$ and $\texttt{Pr}[\varnothing] = 0$. For events A and B, writing $\overline{A}$ ("not A") to denote the event $\overline{A} = S - A$, we have that $\texttt{Pr}[\,\overline{A}\,] = 1 - \texttt{Pr}[A]$, and $\texttt{Pr}[A \cup B] = \texttt{Pr}[A] + \texttt{Pr}[B] - \texttt{Pr}[A \cap B]$.

We can use a *tree diagram* to represent a sequence of random choices, where internal nodes of the tree correspond to random decisions made by the probabilistic process; leaves correspond to the outcomes in the sample space. Every edge leaving an internal node is labeled with the probability of the corresponding random decision; the probability of a particular outcome is precisely equal to the product of the labels on the edges leading from the root to its corresponding leaf.

The *uniform distribution* is the probability distribution in which all outcomes in the sample space S are equally likely—that is, when $\texttt{Pr}[s] = \frac{1}{|S|}$ for each $s \in S$. (*Nonuniform probability* is when this equality does not hold.)

The *Bernoulli distribution with parameter* p is the probability distribution that results from flipping one coin, where the sample space is $\{\text{H}, \text{T}\}$ and $\texttt{Pr}[\text{H}] = p$ (and thus $\texttt{Pr}[\text{T}] = 1 - p$). Such a coin is called *p-biased.* Each coin flip is called a *trial*; the flip is called *fair* if $p = \frac{1}{2}$.

The *binomial distribution with parameters n and p* is a distribution over the sample space $\{0, 1, \ldots, n\}$ determined by flipping a p-biased coin n times and counting the number of times the coin comes up heads. Here $\texttt{Pr}[k] = \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k}$ denotes the probability that there are precisely k heads in the n flips.

The *geometric distribution with parameter* p is a distribution over the positive integers, where the output is determined by the number of flips of a p-biased coin required before we first see a heads; thus $\texttt{Pr}[k] = (1-p)^{k-1} \cdot p$ for any integer $k \geq 1$.

Independence and Conditional Probability

When there are multiple events of interest, then one useful way understanding the relationship between two events is to understand whether one event's occurrence changes the likelihood of the other event also occurring. When there's no change, the events are called *independent*; when there is a change in the probability, the events are called *dependent.* More formally, two events A and B are *independent* (or *uncorrelated*) if and only if $\texttt{Pr}[A \cap B] = \texttt{Pr}[A] \cdot \texttt{Pr}[B]$. Otherwise the events A and B are called *dependent* (or *correlated*). Intuitively, A and B are dependent if A's occurrence/nonoccurrence tells us something about whether B occurs. When knowing that A occurred makes B

more likely to occur, we say that A and B are *positively correlated*; when A makes B less likely to occur, we say that A and B are *negatively correlated.*

The *conditional probability of A given B* is

$$\texttt{Pr}\left[A|B\right] = \frac{\texttt{Pr}\left[A \cap B\right]}{\texttt{Pr}\left[B\right]}.$$

(Treat $\texttt{Pr}\left[A|B\right]$ as undefined when $\texttt{Pr}\left[B\right] = 0$.) Intuitively, we can think of $\texttt{Pr}\left[A|B\right]$ as "zooming" the universe down to the set B. Two events A and B for which $\texttt{Pr}\left[B\right] \neq 0$ are independent if and only if $\texttt{Pr}\left[A|B\right] = \texttt{Pr}\left[A\right]$.

There are a few useful equivalences based on conditional probability. For any events A and B, the *chain rule* says that $\texttt{Pr}\left[A \cap B\right] = \texttt{Pr}\left[B\right] \cdot \texttt{Pr}\left[A|B\right]$; more generally,

$$\begin{aligned} &\texttt{Pr}\left[A_1 \cap A_2 \cap A_3 \cap \cdots \cap A_k\right] \\ &= \texttt{Pr}\left[A_1\right] \cdot \texttt{Pr}\left[A_2|A_1\right] \cdot \texttt{Pr}\left[A_3|A_1 \cap A_2\right] \cdot \;\cdots\; \cdot \texttt{Pr}\left[A_k|A_1 \cap \cdots \cap A_{k-1}\right]. \end{aligned}$$

The *law of total probability* says that $\texttt{Pr}\left[A\right] = \texttt{Pr}\left[A|B\right] \cdot \texttt{Pr}\left[B\right] + \texttt{Pr}\left[A|\,\overline{B}\,\right] \cdot \texttt{Pr}\left[\,\overline{B}\,\right]$.

Bayes' Rule is a particularly useful rule that allows us to "flip around" a conditional probability statement: for any two events A and B, we have

$$\texttt{Pr}\left[A|B\right] = \frac{\texttt{Pr}\left[B|A\right] \cdot \texttt{Pr}\left[A\right]}{\texttt{Pr}\left[B\right]}.$$

Random Variables and Expectation

The probabilistic statements that we've considered so far are about events ("whether or not" questions); we can also consider probabilistic questions about "how much" or "how often." A *random variable* X assigns a numerical value to every outcome in the sample space S—that is, a random variable is a function $X : S \to \mathbb{R}$. (Often we write X to denote the value of a random variable X for a realization chosen according to $\texttt{Pr}$, or perform arithmetic on random variables.) An *indicator random variable* is a $\{0, 1\}$-valued random variable. Two random variables X and Y are *independent* if every two events of the form "$X = x$" and "$Y = y$" are independent.

The *expectation* of a random variable X, denoted $\texttt{E}\left[X\right]$, is the average value of X, defined as $\texttt{E}\left[X\right] = \sum_{x \in S} X(x) \cdot \texttt{Pr}\left[x\right]$. A Bernoulli random variable with parameter p has expectation p. A binomial random variable with parameters p and n has expectation pn. A geometric random variable with parameter p has expectation $\frac{1}{p}$.

Linearity of expectation is the very useful fact that the expectation of a sum is the sum of the expectations. That is, for random variables $X : S \to \mathbb{R}$ and $Y : S \to \mathbb{R}$, we have $\texttt{E}\left[X + Y\right] = \texttt{E}\left[X\right] + \texttt{E}\left[Y\right]$. (Note that there is no requirement of independence on X and Y!) Another useful fact is that, for a positive integer–valued random variable $X : S \to \mathbb{Z}^{\geq 0}$, we have $\texttt{E}\left[X\right] = \sum_{i=1}^{\infty} \texttt{Pr}\left[X \geq i\right]$.

The *conditional expectation* of a random variable X given an event E is the average value of X over outcomes where E occurs, defined as $\texttt{E}\left[X|E\right] = \sum_{x \in E} X(x) \cdot \texttt{Pr}\left[x|E\right]$.

The *variance* of a random variable X is

$$\text{var}\left(X\right) = \texttt{E}\left[\left(X - \texttt{E}\left[X\right]\right)^2\right] = \texttt{E}\left[X^2\right] - \left(\texttt{E}\left[X\right]\right)^2.$$

The *standard deviation* is $\text{std}\left(X\right) = \sqrt{\text{var}\left(X\right)}$.

Key Terms and Results

Key Terms

Probability, Outcomes, and Events

- outcome/ realization
- sample space
- probability function/ distribution
- event
- tree diagram
- uniform vs. nonuniform probability
- fair vs. biased coin flips
- uniform distribution
- Bernoulli distribution
- binomial distribution
- geometric distribution

Independence and Conditional Probability

- independent/ uncorrelated events
- dependent/ correlated events
- positive/ negative correlation
- conditional probability
- chain rule
- law of total probability
- Bayes' Rule

Random Variables and Expectation

- random variable
- indicator random variable
- independent random variables
- expectation
- linearity of expectation
- conditional expectation
- variance
- standard deviation

Key Results

Probability, Outcomes, and Events

1. For a sample space S and events A and B, writing $\overline{A}$ ("not A") to denote the event $S - A$, we have that $\mathtt{Pr}\left[S\right] = 1$, $\mathtt{Pr}\left[\varnothing\right] = 0$, $\mathtt{Pr}\left[\,\overline{A}\,\right] = 1 - \mathtt{Pr}\left[A\right]$, and $\mathtt{Pr}\left[A \cup B\right] = \mathtt{Pr}\left[A\right] + \mathtt{Pr}\left[B\right] - \mathtt{Pr}\left[A \cap B\right]$.
2. Under the uniform distribution, $\mathtt{Pr}\left[s\right] = \frac{1}{|S|}$ for every $s \in S$. Consider parameters p and n. Under a Bernoulli distribution, $\mathtt{Pr}\left[\text{H}\right] = p$ and $\mathtt{Pr}\left[\text{T}\right] = 1 - p$. Under a binomial distribution, $\mathtt{Pr}\left[k\right] = \binom{n}{k} p^k (1-p)^{n-k}$. Under a geometric distribution, $\mathtt{Pr}\left[k\right] = (1-p)^{k-1} p$.

Independence and Conditional Probability

1. Events A and B are independent if and only if $\mathtt{Pr}\left[A \cap B\right] = \mathtt{Pr}\left[A\right] \cdot \mathtt{Pr}\left[B\right]$, or, equivalently, if $\mathtt{Pr}\left[A|B\right] = \mathtt{Pr}\left[A\right]$.
2. The chain rule: $\mathtt{Pr}\left[A \cap B\right] = \mathtt{Pr}\left[B\right] \cdot \mathtt{Pr}\left[A|B\right]$.
3. The law of total probability: $\mathtt{Pr}\left[A\right] = \mathtt{Pr}\left[A|B\right] \cdot \mathtt{Pr}\left[B\right] + \mathtt{Pr}\left[A|\,\overline{B}\,\right] \cdot \mathtt{Pr}\left[\,\overline{B}\,\right]$.
4. Bayes' Rule: $\mathtt{Pr}\left[A|B\right] = \frac{\mathtt{Pr}\left[B|A\right] \cdot \mathtt{Pr}\left[A\right]}{\mathtt{Pr}\left[B\right]}$.

Random Variables and Expectation

1. The *expectation* of a random variable X is the average value of X, defined as $\mathtt{E}\left[X\right] = \sum_{x \in S} X(x) \cdot \mathtt{Pr}\left[x\right]$.
2. A Bernoulli random variable with parameter p has expectation p. A binomial random variable with parameters p and n has expectation pn. A geometric random variable with parameter p has expectation $\frac{1}{p}$.
3. Linearity of expectation: for any two random variables X and Y, we have $\mathtt{E}\left[X + Y\right] = \mathtt{E}\left[X\right] + \mathtt{E}\left[Y\right]$. (Note that there is no requirement of independence on X and Y!)
4. For a random variable $X : S \to \mathbb{Z}^{\geq 0}$, we have that $\mathtt{E}\left[X\right] = \sum_{i=1}^{\infty} \mathtt{Pr}\left[X \geq i\right]$.
5. For a random variable X, we have $\text{var}\left(X\right) = \mathtt{E}\left[(X - \mathtt{E}\left[X\right])^2\right] = \mathtt{E}\left[X^2\right] - \left(\mathtt{E}\left[X\right]\right)^2$.