WHAT IS RANDOM?

Edward Beltrami

Chance and Order in Mathematics and Life

2ND ED.

Springer

What Is Random?

Edward Beltrami

What Is Random?

Chance and Order in Mathematics and Life

Second Edition

Springer

Copernicus Books is a brand of Springer

Edward Beltrami
Department Applied Mathematics and Statistics
Stony Brook University
Stony Brook, NY, USA

ISBN 978-1-0716-0798-5 ISBN 978-1-0716-0799-2 (eBook)
https://doi.org/10.1007/978-1-0716-0799-2

Copernicus is part of Springer, an imprint published by Springer Nature
The registered company is Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

A Word About Notation

It is convenient to use a shorthand notation for certain mathematical expressions that appear often throughout the book. For any two numbers designated as a and b the product "a times b" is written as ab or equivalently $a \times b$, while "a divided by b" is a/b. The product of "a multiplied by itself b times" is a^b so that, for example, 2^{10} means 1024. The expression 2^{-n} is synonymous with $1/2^n$.

In a few places I use the standard notation $\sqrt{a}$ to mean "the square root of a," as in $\sqrt{25} = 5$.

All "numbers greater than a and less than b" are expressed as (a, b). If "greater than" is replaced by "greater than or equal to" then the notation is $[a, b)$.

A sequence of numbers, such as 53371…, is generally indicated by $a_1a_2a_3\ldots$ in which the subscripts 1, 2, 3, … indicate the "first, second, third, and so on" terms of the sequence, which can be finite in length or even infinite (such as the unending array of all even integers).

Preface

We all have memories of peering at a TV screen as myriad little balls churn about in an urn until a single candidate, a number inscribed on it, is ejected from the container. The hostess hesitantly picks it up and, pausing for effect, reads the lucky number. The winner thanks Lady Luck, modern descendant of the Roman goddess Fortuna, blind arbiter of good fortune. We, as spectators, recognize it as simply a random event and know that in other situations Fortuna's caprice could be equally malicious, as Shirley Jackson's dark tale "The Lottery" chillingly reminds us.

The Oxford Dictionary has it that a "random" outcome is one without perceivable cause or design, inherently unpredictable. But, you might protest, isn't the world around us governed by rules, by the laws of physics? If that is so, it should be possible to determine the positions and velocities of each ball in the urn at any future time, and the uncertainty of which one is chosen would simply be a failing of our mind to keep track of how the balls are jostled about. The sheer enormity of possible configurations assumed by the balls overwhelms our computational abilities. But this would be only a temporary limitation: a sufficiently powerful computer could conceivably do that brute task for us,

and randomness would thus be simply an illusion that can be dispelled. After thinking about this for a while you may begin to harbor a doubt. Although nature may have its rules, the future remains inherently unknowable because the positions and velocities of each ball can never really be ascertained with complete accuracy.

The first view of randomness is of clutter bred by complicated entanglements. Even though we know there are rules, the outcome is uncertain. Lotteries and card games are generally perceived to belong to this category. More troublesome is that nature's design itself is known imperfectly, and worse, the rules may be hidden from us, and therefore we cannot specify a cause or discern any pattern of order. When, for instance, an outcome takes place as the confluence of totally unrelated events, it may appear to be so surprising and bizarre that we say that it is due to blind chance. Jacques Monod, in his book "Chance and Necessity", illustrates this by the case of a man hurrying down a street in response to a sudden phone call at the same time that a roof worker accidentally drops a hammer that hits the unfortunate pedestrian's head. Here we have chance due to contingency, and it doesn't matter whether you regard this as an act of divine intervention operating according to a predestined plan or as an unintentional accident. In either case the cause, if there is one, remains indecipherable.

Randomness is the very stuff of life, looming large in our everyday experience. Why else do people talk so much about the weather, traffic, and the financial markets? Although uncertainty may contribute to a sense of anxiety about the future, it is also our only shield against boring repetitiveness. As I argue in a later chapter, chance provides the fortuitous accidents and capricious wit that gives life its pungency. It is important, therefore, to make sense of randomness beyond its anecdotal meanings. To do this we

employ a modest amount of mathematics in this book to remove much of the vagueness that encumbers the concept of random, permitting us to quantify what would otherwise remain elusive. The mathematics also provides a framework for unifying how chance is interpreted from the diverse perspectives of psychologists, physicists, statisticians, computer scientists, and communication theorists.

In the first chapter, I tell the story of how, beginning a few centuries ago, the idea of uncertainty was formalized into a theory of chance events, known today as probability theory. Mathematicians adopt the convention that selections are made from a set of possible outcomes in which each event is equally likely, though unpredictable. Chance is then asked to obey certain rules that epitomize the behavior of a perfect coin or an ideal die, and from these rules one can calculate the odds. *The Taming of Chance*, as this chapter is called, supplies the minimal amount of probability theory needed to understand the "law of large numbers" and the "normal law", which describe in different ways the uncanny regularity of large ensembles of chance events. With these concepts in hand, we arrive at our first tentative answer to the question "what is random?"

To crystalize our thinking, most of the book utilizes the simplest model of a succession of random events, namely a binary sequence (a string of zeros and ones). Although this may seem almost like a caricature of randomness, it has been the setting for some of the most illuminating examples of the workings of chance from the very beginnings of the subject three centuries ago to the present.

If some random mechanism generates a trail of ten zeros and ones, let us say, then there are 2 to the power 10, namely 1024, different binary strings that are possible. One of the first conundrums to emerge is that under the assumption that each string is equally likely, there is a (very) small

possibility that the string generated consists of ten zeros in succession, something that manifestly is not random by any intuitive notion of what random means. So it is necessary to distinguish between a device that operates in a haphazard manner to spew forth digits without rhyme or reason, a random process, and any particular realization of the output that such a process provides. One may argue that the vagaries of chance pertain to the ensemble of possibilities and not to an individual outcome. Nevertheless, a given binary string, such as 0101010101010101, may appear so orderly that we feel compelled to deny its randomness, or it may appear so incoherent, as in the case of 0011000010011011, that we yearn to call it random regardless of its provenance. The last situation conforms to what I mentioned earlier, namely that random is random even though the cause, if any, is unknown. In keeping with this idea the emphasis will shift away from the generating process in succeeding chapters and focus instead on the individual outcomes. This change in outlook parallels the actual shift that has taken place in the last several decades among a broad spectrum of thinkers on the subject, in contrast to the more traditional views that have been central over the last few centuries (and that are discussed in the first chapter). The chapter closes with a new topic for this edition, conditional probability, with Bayes' Theorem developed in the *Technical Notes*.

In *Uncertainty and Information*, the second chapter, I introduce the notion of information formulated by Claude Shannon nearly three quarters of a century ago, since this gives us an additional tool for discussing randomness. Here one talks of information bits and entropy, redundancy, and coding, and this leads me to pose a second test to resolve the question "what is random?" The idea is that if some shorter binary string can serve as a code to generate a longer binary

message, the longer message cannot be random since it has been compressed by removing some of the redundancies within the string. One application concerns the perception of randomness by people in general, a topic much studied by psychologists.

The third chapter, *Janus-Faced Randomness*, introduces a curious example of a binary sequence that is random in one direction but deterministic when viewed in reverse. Knowledge of the past and uncertainty about the future seem to be two faces of the same Janus-faced coin, but more careful scrutiny establishes that what appears as predictable is actually randomness in disguise. I establish that the two faces represent a tradeoff between ignorance now and disorder later. Though there are randomly generated strings that do not appear random, it now emerges that strings contrived by deterministic rules may behave randomly. To some extent, randomness is in the eye of the beholder. Just because you do not perceive a pattern doesn't mean there isn't one. The so-called random-number generators that crop up in many software packages are of this ilk, and they give support to the interpretation of chance as a complex process, like balls in an urn, which only appear random because of the clutter.

The chapter continues with a brief discussion of the early days of the study of thermodynamics in the nineteenth century, when troublesome questions were raised about the inexorable tendency of physical systems to move from order to disorder over time, in apparent contradiction to the laws of physics, which are completely reversible. I show that this dilemma is closely related to the question of binary strings that appear random in spite of being spawned by precise rules. We close with a new topic for this edition, a brief discussion of how uncertainty emerges as ineluctable randomness in quantum theory.

In the fourth chapter *Algorithms, Information, and Chance*, I follow the mathematicians Andrei Kolmogorov and Gregory Chaitin who say that a string is random if its shortest description is obtained by writing it out in its entirety. There is, in effect, no pattern within the string that would allow it to be compressed. More formally, one defines the complexity of a given string of digits to be the length, in binary digits, of the shortest string (i.e., the shortest computer program written in binary form) that generates the successive digits. Strings of maximum complexity are called random when they require programs of about the same length as the string itself. Although these ideas echo those of Chapter 2, they are now formulated in terms of algorithms implemented on computers and the question "what is random?" takes on a distinctly different cast.

Godel's celebrated incompleteness theorem in the version rendered by Alan Turing states that it may not be possible to determine whether a "universal" computer, namely one that can be programmed to carry out any computation whatever, will ever halt when it is fed a given input. Chaitin has reinterpreted this far-reaching result by showing that any attempt to decide the randomness of a sufficiently long binary string is inherently doomed to failure; his argument is reproduced in this chapter.

The penultimate chapter is more speculative. In *The Edge of Randomness*, I review recent work by a number of thinkers that suggests that naturally occurring processes seem to be balanced between tight organization, where redundancy is paramount, and volatility, in which little order is possible. One obtains a view of nature and the arts and the world of everyday affairs as evolving to the edge of these extremes, allowing for a fruitful interplay of chance and necessity, poised between surprise and inevitability. Fortuitous mutations and irregular natural disturbances, for example,

appear to intrude on the more orderly processes of species replication, providing evolution with an opportunity for innovation and diversity.

To illustrate this concept in a particular setting, I include a brief and self-contained account of naturally occurring events that display similar patterns at different scales; they satisfy what is known as *power laws* and describe processes in which there is a high frequency of small events interspersed with a few large magnitude occurrences. This is reminiscent of the order exhibited by chance events in the large as discussed in the first chapter. However, now the prediction of any individual occurrence remains inscrutable because it is contingent on the past history of the process. The emphasis on power laws is new to this edition.

The last chapter, *Fooled by Chance*, looks at some entertaining puzzles and surprising inferences that spring from chance in unexpected ways, spotlighting the often counterintuitive consequences of randomness in everyday life. Up until now this has not been the focus of the present work but it seemed appropriate for me to include this diversion. A powerful mathematical tool, the Poisson distribution, is also introduced in this chapter to provide a fresh insight into the workings of chance. The entire chapter is a further add-on for the second edition.

The book is intended to provoke, entertain, and inform by challenging the reader's ideas about randomness, providing first one and then another interpretation of what this elusive concept means. As the book progresses, I tease out the various threads and show how mathematics, communication engineering, computer science, philosophy, physics, and psychology all contribute to the discourse by illuminating different facets of the same idea.

The material in the book should be readily accessible to anyone with a smattering of college mathematics, *no*

calculus needed. I provide simple numerical examples throughout, coded in MATLAB, to illustrate the various iterative processes and binary sequences that crop up. Three appendices provide some of the background information regarding binary representations and logarithms that are needed here and there, but I keep this as elementary as possible. Although an effort is made to justify most statements of a mathematical nature, a few are presented without corroboration, since they entail close-knit arguments that would detract from the main ideas. You can safely bypass the details without any loss and, in any case, the fine points are available in the *Technical Notes* assembled at the end.

The current second edition is a revision of the earlier version. Certain material has been deleted as no longer compelling while introducing some fresh topics.

One final point is that I do not discuss certain twentieth-century approaches to the concept of randomness associated with the names of Per Martin-Löf, Bruno De Finetti, and Richard von Mises because they are technically more demanding. Moreover their contributions are, in one way or the other, indirectly touched upon in our presentation. To anyone wishing to pursue these ideas see the references in *Sources and Further Readings*.

Setauket, NY, USA Edward Beltrami

Acknowledgments

This book is an amalgam of many sources and ideas connected to the appearance of chance in everyday life. It began to take shape some years ago, when I first became aware of a paper by the statistician M. Bartlett that led to the conundrum of the Janus iterates discussed in Chapters 3 and 4.

The Second Edition of this book benefited enormously from the prompt, efficient, and unfailingly courteous handling of the manuscript by my editor at Springer Nature, Dr. Loretta Bartolini. She enthusiastically supported the project from the outset and provided the technical expertise I needed to overcome a number of difficulties that arose during the revision process. My many thanks to Loretta and her assistant editor Chris Eder.

I also wish to acknowledge Jonathan Cobb, formerly Senior Editor of Copernicus Books at Springer-Verlag, New York, and Dr. David Kramer for a detailed commentary of the First Edition. I am grateful to Alexis Beltrami, the prototypical educated layman, for his critical comments on the penultimate chapter, and my thanks to Professor Hondshik Ahn, of the State University at Stony Brook, for a critical review of the first chapter.

Finally, I am indebted to my wife Barbara for providing sound advice that led to improvements in a garbled early version of the book, and for lending a patient ear to my endless chatter about randomness during our frequent walks together.

Contents

1

The Taming of Chance

An oracle was questioned about the mysterious bond between two objects so dissimilar as the carpet and the city…for some time augurs had been sure that the carpet's harmonious design was of divine origin…but you could, similarly, come to the opposite conclusion: that the true map of the universe is the city, just as it is, a stain that spreads out shapelessly, with crooked streets, houses that crumble one upon the other amid clouds of dust

from Invisible Cities by Italo Calvino

However unlikely it might seem, no one had tried out before then a general theory of chance. Babylonians are not very speculative. They revere the judgments of fate, they deliver to them their lives, their hopes, their panic, but it does not occur to them to investigate fate's labyrinthine laws nor the gyratory spheres which reveal it. Nevertheless…the following conjecture was born: if the lottery is an intensification of chance, a periodical infusion of chaos in the cosmos, would it not be right for chance to intervene in all stages of the drawing and not in one alone? Is it not ridiculous for chance to dictate someone's death and not

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2_1

have the circumstances of that death-secrecy, publicity, the fixed time of an hour or a century-not subject to chance?
from "The Lottery in Babylon" by Jorge Luis Borges

From Unpredictable to Lawful

In the dim recesses of ancient history, the idea of chance was intertwined with that of fate. What was destined to be would be. Chance was personified, in the Roman Empire at least, by the Goddess Fortuna, who reigned as the sovereign of cynicism and fickleness. As Howard Patch puts it in his study of this Roman deity, "to men who felt that life shows no signs of fairness, and that what lies beyond is at best dubious, that the most you can do is take what comes your way, Fortuna represented a useful, if at times flippant, summary of the way things go."

To subvert the willfulness of Fortuna, one could perhaps divine her intentions by attempting to simulate her own mode of behavior. This could be accomplished by engaging in a game of chance in the hope that its results would reveal what choice Fortuna herself would make. There is evidence that pre-Christian people along the Mediterranean coast tossed animal heel bones, called tali, and that this eventually evolved into play with dice. When a chance outcome out of many possible outcomes had been revealed by the casting of lots, one could then try to interpret its omens and portents and decide what action to take next. Julius Caesar, for instance, resolved his agonizing decision to cross the Rubicon and advance upon Rome by allegedly hurling dice and exclaiming "iacta alea est," the die is cast. This is echoed in our own time when people draw lots to decide who's first or, for that matter, who is to be last. The very essence of fair play is to flip a coin so that chance can decide the next move.

The interpretation of omens was an attempt to decipher the babble of seemingly incoherent signs by encoding them into a compact and coherent prophesy or, perhaps, as a set of instructions to guide the supplicant. Seen this way, divination is a precursor to the use of coding in information theory and message compression in algorithmic complexity theory, topics that will figure prominently in later portions of this book.

Fortuna was not all foreboding. There is evidence that the elements of chance were employed not just for augury and divination but, on a more playful side, for diversion. Games involving chance and gambling were firmly established by the Renaissance, and a number of individuals had begun to notice that when regularly shaped and balanced dice were tossed repeatedly certain outcomes, such as five dots on one of the six faces, seemed to occur on the average about a sixth of the time. No one was more persuasive about this than the sixteenth-century figure of Girolamo Cardano, a celebrated Italian physician and mathematician, who wrote a small tract on gambling, *Liber de Ludo Aleae*, in which he demonstrated his awareness of how to calculate the winning odds in various games.

This newly found grasp on the workings of chance in simple games would soon evolve into a broader understanding of the patterns revealed by chance when many observations are made. In retrospect, it seems inevitable that the study of randomness would turn into a quantitative science, parallel to the manner in which the physical sciences were evolving during the late Renaissance. Over the next two centuries, a number of individuals, such as the mathematicians Blaise Pascal and Pierre de Fermat, had a hand in reining in the arbitrary spirit of Fortune, at least in games of dice and cards, but it wasn't until the early eighteenth century, after the calculus had been invented and mathematics in general had reached a level of maturity unthinkable a few centuries earlier, that probable and improbable events could

be adequately expressed in a mathematical form. It was at this time that the tools were forged that led to the modern theory of probability. At the same time, the study of statistics as we now know it began to emerge from the data gathering efforts directed at providing mortality tables and insurance annuities, both of which hinge on chance events.

The Swiss mathematician Jakob Bernoulli, in his *Ars Conjectandi* of 1713 and, shortly thereafter in 1718, Abraham de Moivre, in *The Doctrine of Chances*, stripped the element of randomness to its bare essentials by considering only two possible outcomes, such as black and white balls selected blindly from an urn, or tosses of an unbiased coin. Imagine that someone flips a balanced coin a total of n times, for some integer n. The proportion of heads in these n tosses, namely, the actual number of heads produced, divided by n, is a quantity usually referred to as the *sample average* since it depends on the particular sample of a sequence of coin flips that one gets. Different outcomes of n tosses will generally result in different sample averages.

What Bernoulli showed is that as the sample size n gets larger, it becomes increasingly likely that the proportion of heads in n flips of a balanced coin (the sample average) will not deviate from one half by more than some fixed margin of error. This assertion will be rephrased later in more precise terms as the "Law of Large Numbers." de Moivre put more flesh on Bernoulli's statement by establishing that if the average number of heads is computed many times, most of the sample averages have values that cluster about $\frac{1}{2}$, while the remainder spread themselves out more sparsely the further one gets from $\frac{1}{2}$. Moreover, de Moivre showed that as the sample size gets larger, the proportion of sample averages that are draped about $\frac{1}{2}$ at varying distances begins to look like a smooth bell-shaped curve, known either as the *normal* or the *Gaussian* curve (after the German mathematician Carl-Friedrich Gauss whose career straddled

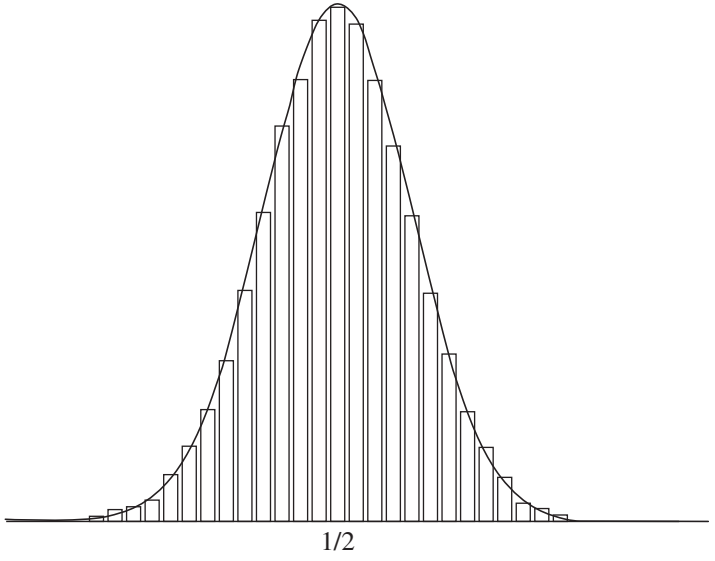

Fig. 1.1 The distribution of 10,000 sample averages at varying distances from ½. The smooth bell-shaped curve is a ubiquitous presence in statistics and is known as the normal, or sometimes the Gaussian, curve

the eighteenth and nineteenth centuries). This phenomenon is illustrated in Fig. 1.1, in which 10,000 sample averages are grouped into little segments along the horizontal axis. The height of the rectangular bins above the segments represents the number of sample averages that lie within the indicated interval. The rectangles decrease in size the further one gets from $\frac{1}{2}$ which shows that less and less sample averages are to be found at longer distances from the peak at $\frac{1}{2}$. The overall profile of the rectangles, namely, the distribution of sample averages, is bell-shaped, and, as n increases, this profile begins to approximate ever more closely the smooth (Gaussian) curve that you see superimposed on the rectangles in the figure. Though formal proofs of de Moivre's theorem and of Bernoulli's law are beyond the

scope of the present work, the results themselves will be put to good use later in this chapter.

Taken together the assertions of Bernoulli and de Moivre describe a kind of latent order emerging from a mass of disordered data, a regularity that manifests itself amid the chaos of a large sample of numbers. By the early years of the nineteenth century, the use of the new methods to handle large masses of data was enthusiastically employed to harness uncertainty in every sphere of life, beyond the fairly benign examples of games, and never more so than by zealous government bureaucracies bent on grappling with the torrents of data being collected about the populations within their borders. Insanity, crime, and other forms of aberrant behavior had been abundantly cataloged in the eighteenth and early nineteenth centuries, and a way for extracting the social implications of these numbers had at last become available. One could now define "normal" behavior as being within a certain fraction of the average of a large sample of people, with deviants lying outside this range, in the same way as one spoke of the average of many coin tosses.

What was once puzzling and unpredictable now appeared to fall into patterns dictated by the normal curve. This curve was empirically regarded by sciences like astronomy as a "law of errors," but some scientists, the French astronomer Adolphe Quetelet in particular, turned it into a bastion of social theory. By 1835 Quetelet went so far as to frame the concept of "the average man" that is still with us today. The period of sorting and interpreting data, the very stuff of statistics, had begun in earnest, and it is no exaggeration to say that the taming of chance had come of age.

Throughout the nineteenth and into the beginnings of the twentieth century, the applications of mathematical ideas to the study of uncertainty became more widespread,

and, in particular, they played an important role in physics with the study of statistical mechanics (a topic that we touch on in Chapter 3), where large ensembles of molecules collide with each other and scatter at random.

The mathematical methods for the study of random phenomena became more sophisticated throughout the first part of the twentieth century, but it remained for the Russian mathematician Andrei Kolmogorov to formalize the then current thinking about probability in a short but influential monograph that was published in 1933. He established a set of hypotheses about random events that, when properly used, could explain how chance behaves when one is confronted with a large number of similar observations of some phenomenon. Probability has since provided the theoretical underpinnings of statistics in which inferences are drawn from numerical data by quantifying the uncertainties inherent in using a finite sample to draw conclusions about a very large or even unlimited set of possibilities. You will see a particular instance of statistical inference a little later in this chapter, when we attempt to decide whether a particular string of digits is random.

The fascination of randomness is that it is pervasive, providing the surprising coincidences, bizarre luck, and unexpected twists that color our perception of everyday events. Although chance may contribute to our sense of unease about the future it is also, as I argue in the penultimate chapter, a bulwark against stupefying sameness, giving life its sense of ineffable mystery. That is why we urgently ask "what is random"?

The modest amount of mathematics employed in this book allows us to move beyond the merely anecdotal meanings of randomness, quantifying what would otherwise remain elusive and brings us closer to knowing what random is.

Probability

The subject of probability begins by assuming that some mechanism of uncertainty is at work giving rise to what is called randomness, but it is not necessary to distinguish between chance that occurs because of some hidden order that may exist and chance that is the result of blind lawlessness. This mechanism, figuratively speaking, churns out a succession of events, each individually unpredictable, or it conspires to produce an unforeseeable outcome each time a large ensemble of possibilities is sampled.

A requirement of the theory of probability is that we be able to describe the outcomes by numbers. The totality of deaths from falling hammers, the winning number in a lottery, the price of a stock, and even yes and no situations, as in it does or does not rain today (which is quantifiable by a zero for "no" or a one for "yes"), are all examples of this.

The collection of events whose outcomes are dictated by chance is called a *sample space* and may include something as simple as two elements, head or tail in coin tossing, or it may be something more involved, as the eight triplets of heads and tails in three tosses, or the price of a hundred different stocks at the end of each month. The word "outcome" is interchangeable with observation, occurrence, experiment, or trial, and in all cases it is assumed that the experiment, observation, or whatever can be repeated under essentially identical conditions as many times as desired even though the outcome is each time unpredictable. With games, for example, one assumes that cards are thoroughly shuffled or that balls scrambled in an urn are selected by the equivalent of a blindfolded hostess. In other situations where the selection mechanism is not under our control, it is assumed that nature arbitrarily picks one of the possible outcomes. It matters not whether we think of the

succession of outcomes as a single experiment repeated many times or a large sample of many experiments carried out simultaneously. A single die tossed three times is the same as three dice tossed once each.

Each possible outcome in a finite sample space is called an *elementary event* and is assigned a number in the range from zero to one. This number, the event's *probability*, designates the likelihood that the event takes place. Zero means that it cannot happen, and the number one is reserved for something that is certain to occur. The interesting cases lie in-between.

Imagine that a large, possibly unlimited, number of observations are made and that some particular event takes place unexpectedly from time to time. The probability of this event is a number between zero and one that expresses the ratio between the actual number of occurrences of the event and the total number of observations. In tossing a coin, for example, head and tail are each assigned a probability of $\frac{1}{2}$ whenever the coin seems to be balanced. This is because one expects that the event of a head or tail is equally likely in each flip and so the average number of heads (or tails) in a large number of tosses should be close to $\frac{1}{2}$.

More general events in a sample space are obtained by considering the union of several elementary events. An event E (sometimes other letters are used, such as A or B) has a probability assigned to it just as was done with the elementary events. For example, the number of dots in a single toss of a balanced die leads to six elementary events, namely, the number of dots, from one to six, on the upturned face. Event E might then be "the upturned face shows a number greater than four," which is the union of the elementary events "5 dots" and "6 dots," and, in this case, the probability of "5 or 6 dots" is $\frac{2}{6}$ or $\frac{1}{3}$.

Two events that represent disjoint subsets of a sample space, subsets having no point in common, are said to be *mutually exclusive*. If A and B represent mutually exclusive events, the event "either A or B takes place," usually denoted by the shorthand $A \cup B$, has *a probability equal to the sum of the individual probabilities of A and B*. The individual probabilities of the union of any number of mutually exclusive subsets of the sample space (i.e., a bunch of events that represent sets of possibilities having nothing in common) must add up to unity since one of them is certain to occur.

If the occurrence or nonoccurrence of a particular event is hit or miss, totally unaffected by whether it happened before, we say that the outcomes are *statistically independent* or, simply, *independent*.

For a biased coin in which a head has only probability $\frac{1}{3}$ of happening, the long-term frequency of heads in a sequence of many tosses of this coin carried out in nearly identical circumstances should appear to settle down to the value $\frac{1}{3}$. Although this probability reflects the uncertainty associated with obtaining a head or tail for a biased coin, I attach particular significance in this book to unbiased coins or, more generally, to the situation in which all the elementary events are equally-likely. When there is no greater propensity for one occurrence to take place over another, the outcomes are said to be *uniformly distributed*, and, in this case, the probabilities of the N separate elementary events that constitute some sample space are each equal to the same value $\frac{1}{N}$. For instance, the six faces of a balanced die are all equally-likely to occur in a toss, and so the probability of each face is $\frac{1}{6}$.

The quintessential *random process* will be thought of, for now at least, as a succession of *independent and uniformly distributed outcomes*.

To avoid burdening the reader at this point with what might be regarded as boring details, I append additional information and examples about probability in an optional section in the *Technical Notes* for this chapter.

In thinking about random processes in this book, it often simplifies matters if we regard successive outcomes as having only two values, zero or one. This could represent any binary process such as coin tossing or, in more general circumstances, occurrence or nonoccurrence of some event (success or failure, yes or no, on or off). Ordinary whole numbers can always be expressed in *binary form*, namely, a string of zeros and ones. It suffices to write each number as a sum of powers of 2. For example,

$$9 = 1\times 2^3 + 0\times 2^2 + 0\times 2^1 + 1\times 2^0 = 8 + 0 + 0 + 1;$$

the number 9 is then identified with the binary digits multiplying the individual powers of 2, namely, 1001. Similarly,

$$\begin{aligned} 30 &= 1\times 2^4 + 1\times 2^3 + 1\times 2^2 + 1\times 2^1 + 0\times 2^0 \\ &= 16 + 8 + 4 + 2 + 0 \end{aligned}$$

and so 30 is identified with 11110 (additional details about binary representations are provided in Appendix B).

The examples in this chapter and, indeed, throughout most of this book are framed in terms of binary digits. Any outcome, such as the price of a volatile stock or the number of individuals infected with a contagious disease, may therefore be coded as a string of zeros and ones called *binary strings* or, as they are sometimes referred to, *binary sequences*.

Computers, incidentally, express numbers in binary form since the circuits imprinted on chips respond to low/high voltages, and the familiar Morse code, long the staple of telegraphic communication, operates with a binary system of dots and dashes.

Order in the Large

It has already been mentioned that one expects the proportion of heads in n flips of a balanced coin to be close to $\frac{1}{2}$ leading one to infer that the probability of a head (or tail) is precisely $\frac{1}{2}$. What Jakob Bernoulli did was to turn this assertion into a formal mathematical theorem by hypothesizing a sequence of independent observations or, as they are often referred to, independent trials, of a random process with two possible outcomes that are labeled zero or one, each mutually exclusive of the other. Instead of being of being restricted to just balanced coins, he more generally assumed a probability, designated by p, for the event "one will occur" and a corresponding probability $1 - p$ for the event "zero will occur" (i.e., the digit one does not appear). Since one of the two events must take place at each trial, their probabilities p and $1 - p$ sum to unity. For $p = \frac{1}{3}$, let us say, it would mean that the coin is unbalanced biased toward coming up tails.

These assumptions idealize what would actually be observed in viewing a binary process in which there are only two outcomes. The sequence of outcomes, potentially unlimited in number, will be referred to as *Bernoulli p-trials* or, equivalently, as a *Bernoulli p-process*.

In place of one and zero, the binary outcomes could also be thought of as success and failure, true and false, yes and no, or some similar dichotomy. In medical statistics, for example, one might ask whether or not a particular drug treatment is effective when tested on a group of volunteers selected without bias from the population at large.

For a sequence of *Bernoulli p-trials*, you intuitively expect that the proportion of ones in n independent trials should approach p as n increases. In order to turn this intuition into a more precise statement, let S_n denote the number of ones in n trials. Then S_n divided by n, namely, S_n/n,

represents the fraction of ones in n trials; it is customary to call S_n/n the *sample average or sample mean*. What Jakob Bernoulli established is that as n increases, it becomes increasingly certain that the absolute difference between S_n/n and p is less than any pre-assigned measure of discrepancy. Stated more formally, this says that the probability of the event "S_n/n is within some fixed distance from p" will tend to one as n is allowed to increase without bound. Bernoulli's theorem is known as the *Law of Large Numbers*.

The Bernoulli $\frac{1}{2}$-trials are the quintessential example of a random process as we defined it earlier: the succession of zeros and ones is independent and uniformly distributed with each digit having an equal chance of occurring. In Fig. 1.2 there is a plot of S_n/n versus n generated by a computer simulated random coin toss with $p = \frac{1}{2}$, and in this

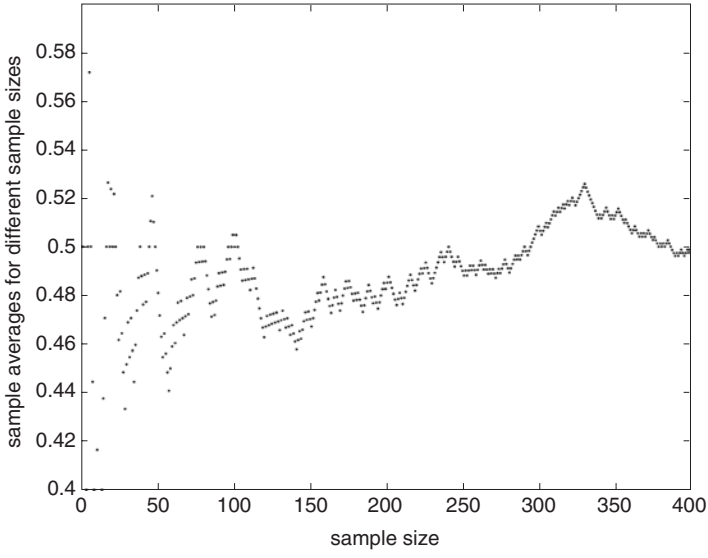

Fig. 1.2 Fluctuations in the value of the sample average S_n/n for n up to 400. Note that the fluctuations appear to settle down to the value ½ as n gets larger, much as the Law of Large Numbers leads us to expect

instance, we see that the sample mean does meander toward $\frac{1}{2}$ in its own idiosyncratic manner.

This is a good a place to comment on the oft-quoted "*law of averages*," which is epitomized by the belief that after a long streak of bad luck, as seen by a repeated block of zeros (e.g., tails, you lose), fortune will eventually turn things around by favoring the odds of a one (heads, or a win). The perception is that a long string of zeros is so unlikely that after such a bizarre succession of losses the chance of a head is virtually inevitable, and a gambler has a palpable sense of being on the cusp of a win. However, the independence of tosses exposes this as a delusion: the probability of another zero remains the same $\frac{1}{2}$ as for every other toss. What is true is that in the ensemble of all possible binary strings of a given length the likelihood that any particular string has a freakishly long block of zeros is quite small. This dichotomy between what is true of a particular realization of a random process and the ensemble of all possible outcomes is a persistent thorn in the flank of probability theory.

If, instead of the sample mean, one looks simply at the sum S_n something unexpected happens that is worth revealing since it underscores the often counterintuitive nature of randomness. In the place of zero and one, suppose the two outcomes are plus and minus one and consider a coin tossing game between Peter and Paul in which Paul gains a dollar, denoted by +1, if a fair coin comes up heads and loses a dollar, unsurprisingly -1 in this case, if the outcome is tails. Just the opposite is true for his opponent Paul. Now S_n will indicate Peter's total winnings, positive or negative, after n tosses. A gain for Peter is of course a loss for Paul. S_n executes a random walk among the positive and negative integers fluctuating wildly in either direction returning on occasion to zero (the players are even) before wandering off with positive and negative excursions, some large and others small. Our intuition tells us that one of the players will

be in the lead roughly half the time. But that is emphatically not true! In fact it can be shown (not a computation we want to do here) that one player will be in the lead during most of the game's duration. In Fig. 1.3 accumulated winnings of 5000 tosses are plotted, and it illustrates dramatically that *a lead or loss is maintained for most of the game*, even though it tends to fluctuate in value. What is true is that if one considers a large number of similar games of length n, Peter will indeed be in the lead in about half of these games and Paul in the other half. This ties in with the Law of Large Numbers if we now return to zero and one variables with one indicating that Paul is in the lead at the nth toss. Allowing N games to be played we have N Bernoulli trials, one for each game, with probability $p = \frac{1}{2}$.

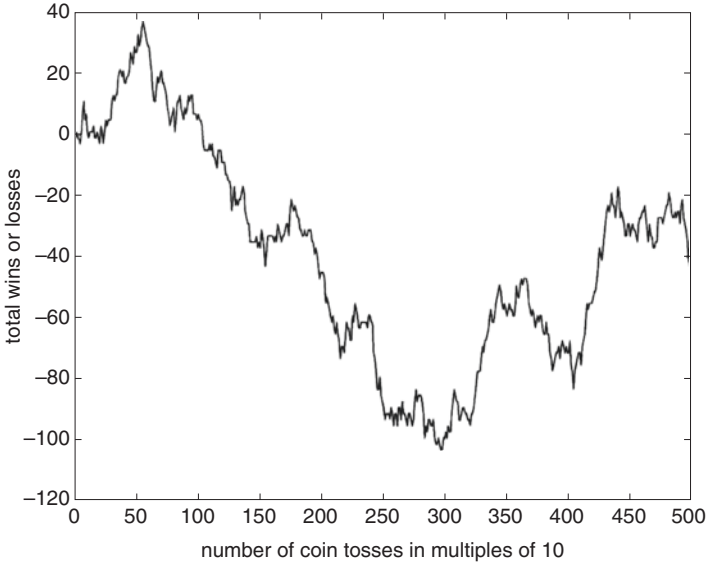

Fig. 1.3 Fluctuations of total wins and losses in a game of heads and tails with a fair coin over a span of 5000 tosses, plotted every tenth value. A loss is clearly maintained in over more than 4000 flips of the coin

Then the sample mean S_N/N will be close to $\frac{1}{2}$ with mounting confidence as N increases.

The Normal Law

As we mentioned earlier, de Moivre gave a new twist to the Law of Large Numbers for Bernoulli p-trials. The Law of Large Numbers says that for a large enough sample size, the sample average is likely to be near p, but de Moivre's theorem, a special case of what is known today as the Central Limit Theorem, allows one to estimate the probability that a sample average is within a specified distance from p. One first must choose a measure of how far S_n/n is from p. This measure is conventionally defined to be $\sigma / \sqrt{n}$, where the symbol σ stands for $\sqrt{p(1-p)}$. With p equal to .25, for example, σ is equal to the square root of $\frac{3}{16}$.

de Moivre's theorem says that the proportion (percentage) of sample averages S_n/n that fall in the interval of values between $p - c\sigma / \sqrt{n}$ and $p + c\sigma / \sqrt{n}$ for a given value of c is approximately equal to the fraction of total area that lies under the normal curve between $-c$ and c. The *normal curve*, often called the *Gaussian curve*, or even the *law of errors*, is the ubiquitous bell-shaped curve that is familiar to all students of statistics and that is displayed in Fig. 1.1. The essence of de Moivre's theorem is that the error made in using the normal curve to approximate the probability that S_n/n is within the given interval decreases as n gets larger. It is this fact that will shortly allow us to devise a test, the first of several, for deciding whether a given binary string is random or not.

The area under the normal curve has been computed for all c, and these numbers are available as a table in virtually every statistics text and statistics software package. One

value of c that is particularly distinguished by common usage is 1.96 which corresponds, according to the tables, to a probability of .95. A probability of .95 means that S_n/n is expected to lie within the interval in 19 out of 20 cases or, put another way, the odds in favor of S_n/n falling within the interval is 19 to 1.

In preparation for using the above results to test randomness, I will attempt to put de Moivre's theorem in focus for $p = \frac{1}{2}$ (think of a balanced coin that is not biased in favor of falling on one side or the other). Before we carry out this computation, I hope you will agree that it is less cumbersome to express the c interval by the mathematical shorthand

$$\left(p - \frac{c\sigma}{\sqrt{n}},\ p + \frac{c\sigma}{\sqrt{n}} \right).$$

Since the quantity σ is the square root of $\frac{1}{4}$ in the present case, namely, $\frac{1}{2}$, and since a probability of .95 corresponds to c = 1.96, as you just saw, the expression above becomes

$$\left(.5 - \frac{1.96}{2\sqrt{n}},\ .5 + \frac{1.96}{2\sqrt{n}} \right).$$

and by rounding 1.96 up to 2.0, this takes on the pleasingly simple form

$$\left(.5 - \frac{1}{\sqrt{n}},\ .5 + \frac{1}{\sqrt{n}} \right)$$

which means that roughly 95% of the sample averages are expected to take on values between .5 plus or minus $1/\sqrt{n}$.

For example, in 10,000 tosses of a fair coin, there is a .95 probability that the sample averages lie within (.49, .51) since the square root of 10,000 is 100. If there are actually 5300 heads in 10,000 tosses, then S_n/n is .53, and we are disinclined to believe that the coin is balanced in favor of thinking that it is unbalanced (p not equal to $\frac{1}{2}$). Computations like this are part of the obligatory lore of most courses in statistics.

The distribution of values of 10,000 different samples of S_n/n for $n = 15$ is plotted in Fig. 1.4 (upper half) for the case

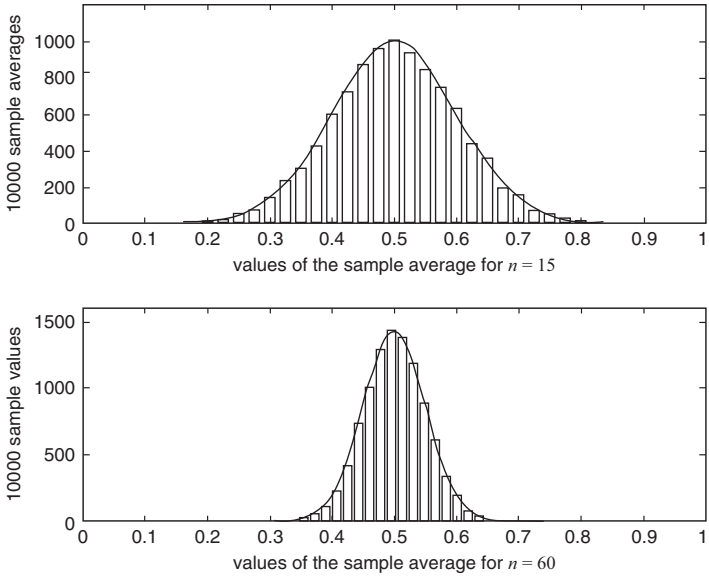

Fig. 1.4 The distribution of 10,000 sample averages S_n/n for $n = 15$ in the top half. The height of each rectangle indicates the fraction of all sample averages that lie in the indicated interval of the horizontal axis. The distribution of 10,000 sample averages for $n = 60$ is on the bottom half. The spread about ½ in the bottom figure is less than in the upper, and it peaks higher. Thus, the 95% confidence interval is narrower. This is because the dispersion about ½ is inversely proportional to the square root of n. Observe that the distribution of values appears to follow a normal curve

$p = 1/2$, and we see that they cluster about $1/2$ with a dispersion that appears roughly bell-shaped. The height of each rectangle in the figure indicates the number of all sample averages that lie in the indicated interval along the horizontal axis. For purposes of comparison, the actual normal curve is superimposed on the rectangles, and it represents the theoretical limit approached by the distribution of sample averages as n increases without bound.

The lower half of Fig. 1.4 compares the difference between using 10,000 sample averages of size $n = 60$. Since the dispersion about $1/2$ decreases inversely with the square root of n, the bottom distribution is narrower and its peak is higher.

It appears, then, that from the disarray of individual numbers a certain lawfulness emerges when large ensembles of numbers are viewed from afar. The wanton and shapeless behavior of these quantities when seen up close gels, in the large, into the orderly bell-shaped form of a "law of errors," the *normal law*.

Extensions of the normal law to other than Bernoulli p-trials are today known today as the "Central Limit Theorem." It must be remembered, however, that the normal law is a mathematical statement based on a set of assumptions about the perception of randomness that may or may not always fully square with actuality.

Probability theory permits statements like the Law of Large Numbers or the Central Limit Theorem to be given a formal proof, capturing what seems to be the essence of statistical regularity as it unfolds from large arrays of messy data, but it does not, of itself, mean that nature will accommodate us by guaranteeing that the sample average will tend to p as n increases without bound or that the values of the sample average will distribute themselves according to a normal curve. No one ever conducts an infinite sequence of

trials. What we have instead is an empirical observation based on limited experience, and not a law of nature. This reminds me of an oft-quoted observation of the mathematician Henri Poincare to the effect that practitioners believe that the normal law is a theorem of mathematics, while the theoreticians are convinced that it is a law of nature. It is amusing to compare this to the ecstatic remarks of the nineteenth-century social scientist Francis Galton, who wrote "I scarcely know of anything so apt to impress the imagination as the wonderful form of cosmic order expressed by the 'Law of Frequency of Error' (*the normal law*) it reigns with serenity and in complete self-effacement, amid the wildest confusion. The larger the mob and the greater the apparent anarchy, the more perfect is its sway. It is the supreme law of unreason."

Is It Random?

If there was a signature example in statistics, it would be to test whether a particular binary string is due to chance or whether it can be dismissed as an aberration. Put another way, if you do not know the origins of this string, it is legitimate to ask whether it is likely to be the output of a random process or if it is a contrived and manipulated succession of digits. Let us see how a statistician can negate the hypothesis that it came from a random mechanism, with a prescribed degree of confidence, by applying de Moivre's theorem. In the computations below n is chosen to be 25 which, as it turns out, is large enough for the normal law to be applied without undue error and small enough to avoid the numbers from becoming unwieldy.

Suppose you are given the following string of 25 zeros and ones, which seems to have preponderance of ones, and you ask whether it is likely to have been generated from a

random process with $p = \frac{1}{2}$; statisticians would call this a "*null hypothesis*":

11101100011011111111011110

I now carry out a computation that is similar to the one done a bit earlier in this chapter. With S_n denoting the number of ones, let us adopt a probability .95 that the sample average S_n/n differs from $\frac{1}{2}$ in magnitude by less than $1/\sqrt{n}$. Since $n = 25$, $1/\sqrt{n}$ equals 1/5. The sample average is therefore expected to lie within the interval of $\frac{1}{2}$ plus or minus $\frac{1}{5}$, namely, (.4, .6), in 19 out of 20 cases. There are actually 18 ones in the given string, and so S_n/n equals 18/25 = .72. This number lies outside the bounds of the interval, and therefore the null hypothesis that p equals $\frac{1}{2}$ is rejected in favor of the alternative hypothesis that p does not equal $\frac{1}{2}$. Note the word *reject*. If the sample average did in fact lie within the interval, we would not be accepting the hypothesis of $p = \frac{1}{2}$ but simply not rejecting it. This is because there remains some doubt as to whether the string was randomly generated at all. It could have been deliberately set up with the intent to deceive, or, perhaps, it represents the outcome of Bernoulli p-trials in which p is not equal to $\frac{1}{2}$. Consider, in fact, the string below generated with $p = .4$:

01011100100101111111001000

In this case $S_n = 13/25 = .52$, well within the designated interval, even though the hypothetical coin that generated the string is biased. Similarly, though the sequence

01010101010101010101010101

has a regularly repeating pattern, we find that S_n/n is 12/25 = .48, and therefore the null hypothesis of randomness cannot be rejected in spite of the suspicious nature of the string!

The decision not to reject is a cautionary tactic in light of the evidence, with the agreement that a .95 probability is the demarcation line for disbelief. In the opening pages of Dorothy Sayer's 1930 book Strong Poison, there is an analogous situation with the one adopted in a court of law in which "every accused person is held to be innocent unless and until he is proven otherwise. It is not necessary for him, or her, to prove innocence; it is, in the modern slang, 'up to' the Crown to prove guilt, and unless you are quite satisfied that the Crown has done this beyond any reasonable doubt, it is your duty to return a verdict of 'not guilty.' This does not mean that the prisoner has established her innocence by proof; it simply means that the Crown has failed to produce in your minds an undoubted conviction of guilt."

The flip side of the coin, however, is that one can err by rejecting a particular sequence, such as a string of mostly zeros, even though it might have, in fact, been generated by a Bernoulli $\frac{1}{2}$ process of uniformly distributed outcomes. Not only can a truly guilty person be set free, but an innocent soul might be judged guilty!

The test invoked above is evidently not a powerful discriminator of randomness. After all, it only checks on the relative balance between zeros and ones as a measure of chance. I included it solely to illustrate the normal law and to provide the first, historically the oldest, and arguably the weakest of tools in the arsenal needed to field the question "is it random?"

There are a host of other statistical tests, decidedly more robust and revealing, to decide that a binary sequence is random. One could, for example, check to see whether the

total number of successive runs of zeros or ones, or the lengths of these runs, is consistent with randomness at some appropriate confidence level. A *run* of zeros means an unbroken succession of zeros in the string flanked by the digit one or by no digit at all, and a run of ones is defined in a similar manner. Too few runs, or runs of excessive length, are indicators of a lack of randomness, as are too many runs in which zero and one alternate frequently. This can be arbitrated by a statistical procedure called the *runs test* that is too lengthy to be described here, but let us look at two examples to see what is involved. The sequence 000000000000011100000000 has 2 runs of zeros and a single run of ones, whereas the sequence 010101010101010101010101 has 12 runs of 1 and 13 of zero. Neither of these sequences would pass muster as random using a runs test at some appropriate confidence level, even though the use of de Moivre's theorem does not reject the randomness of the second string, as you saw.

Examining runs and their lengths raises the interesting question of how people perceive randomness. The tendency is for individuals to reject patterns such as a long run as not typical of randomness and to compensate for this by judging frequent alternations between zeros and ones to be more typical of chance. Experiments by psychologists who ask subjects to either produce or evaluate a succession of digits reveal a bias in favor of more alternations than an acceptably random string can be expected to have; people tend to regard a clumping of digits as a signature pattern of order when in fact the string is randomly generated. Confronted with HHHTTT and HTTHTH, which do you think is more random? Both, of course, are equally-likely outcomes of tossing a fair coin.

Incidentally, people also tend to discern patterns whenever an unequal density of points occurs spatially, even if

the points are actually distributed by chance. The emergence of clumps in some portions of space may lead some observers to erroneously conclude that there is some causal mechanism at play. A high incidence of cancer in certain communities, for example, is sometimes viewed as the result of some local environmental condition when, in fact, it may be consistent with a random process.

The psychologists Maya Bar-Hillel and Willem Wagenaar comment that an individual's assessment of randomness in tosses of a fair coin seems to be based on the "equal probability of the two outcomes together with some irregularity in the order of their appearance; these are expected to be manifest not only in the long run, but even in relatively short segments-as short as six or seven. The flaws in people's judgments of randomness in the large is the price of their insistence on its manifestation in the small." The authors provide an amusing example of this when they quote Linus in the comic strip "Peanuts." Linus is taking a true-false test and decides to foil the examiners by contriving a "random" order of TFFTFT; he then triumphantly exclaims "if you're smart enough you can pass a true or false test without being smart." Evidently Linus understands that in order for a short sequence of six or seven T and F to be perceived as random, it would be wise not to generate it from Bernoulli $\frac{1}{2}$-trials, since this could easily result in a nonrandom looking string.

The illusion of randomness can also be foisted on an unsuspecting observer in other ways. A deck of cards retains a vestigial "memory" of former hands unless it has been shuffled many times. One mix permutes the ordering of the cards, but the previous arrangement is not totally erased, and you can be deluded into thinking that the pack is now randomly sorted.

We saw that whole numbers can be represented by finite binary strings. It turns out that any number, in particular

all numbers between zero and one, can be similarly expressed in terms of a binary string that is usually infinite in length. We accept this fact at present and leave the details to Appendix B.

A number between zero and one was defined to be *normal* by the mathematician Emile Borel (not to be confused with the term "normal" used earlier) if every digit 0 or 1 in the binary string that represents the number appears with equal frequency as the number of digits grows to infinity and, additionally, if the proportions of all possible runs of binary digits of a given length are also equal. In other words, in order for x to be normal, not only must 0 and 1 appear with equal frequency in the binary representation of x, but so must 00, 01, 10, and 11 and, similarly, all 8 triplets 000, 001, …, 111. The same, moreover, must be true for each of the 2^k k-tuples of any length k. Since successive digits are independent, the probability of any block of length k is the same. In a precise technical sense that we need not get into here, most numbers are normal, and in the next chapter, you will see that any reasonable candidate for a random sequence must, at the very least, define a *normal number* although providing an explicit example of one has proven to be an elusive task. The only widely quoted example, due to D. Champernowne, is the number whose binary representation is found by taking each digit singly and then in pairs and then in triplets, and so on:

$$0100011011000001010100\supset$$

A Bayesian Perspective

By the middle of the eighteenth century, a subtle new slant on the idea of chance emerged based on the work of an obscure clergyman Rev. Thomas Bayes who, in 1763,

posthumously published a pamphlet titled *An Essay Toward Solving a Problem in the Doctrine of Chances*. His work was extended and clarified by Pierre-Simon Laplace a little later in 1774. Although Bernoulli had established the likelihood that the sample means S_n/n lie within some fixed interval about the known probability p of some event, the perspective shifted to obtaining the likelihood that an unknown probability of an event lies within some given interval of sample means. We can say that if Bernoulli reasoned from cause to effect, then Bayes and Laplace worked in the opposite direction from an effect to a cause. At first blush these sound like equivalent goals since for large enough n the sample means get close to the given probability with increasing confidence so one might be excused for taking the value of p to equal the limit of the samples. This is called the frequentist definition of probability, namely, that p equals the limiting frequency (or, equivalently, the proportion) of successes in n trials as n becomes increasingly large. More generally, if an event E occurs r times in n independent trials, the probability of E is very nearly equal to the relative frequency r/n for sufficiently large n. Everything we've done so far in this book has been consonant with this viewpoint.

A whimsical way to think about the difference between the two approaches is that if Bernoulli tells us that p is the probability of getting a head when tossing a coin, he is making a proclamation about the behavior of the coin, whereas Bayes doesn't know what the coin will do and so what he asserts *is not about the coin itself but about his belief regarding what the coin will do*.

To explain the Bayesian approach more generally requires a brief detour to introduce conditional probabilities. Suppose that for any two events, call them A and B, we construct a new event "A and B" to mean that if one has

occurred so has the other and indicate this fact using the shorthand $A \cap B$. Then the notation prob $(A \mid B)$ will denote "the probability that event A occurs conditional on the fact that event B has taken place." This is a quantity that is proportional to the probability of $A \cap B$ because B is now where all the action is by which I mean that the original sample space has been reduced to just B, as illustrated in Fig. 1.4. More details are provided in the *Technical Notes*, but one point to be made immediately is that one can equally define the opposite probability of "event B occurs given that event A has taken place." These two conditional probabilities pointing in contrary directions can be and often are quite different. The case cited above regarding sample means is a fairly benign confusion that does little harm, but in general the misuse of conditionals can be more worrying.

The Bayesian approach as used in contemporary practice is to begin with a degree of belief about some hypothesis or supposition denoted by H and that is quantified as a *prior* probability. One then computes the conditional probability that one will observe an outcome B if H is true. This is known as the *likelihood of the evidence B given hypothesis H*. Finally, an adroit use of conditional probabilities, known as Bayes's Theorem, establishes that the *posterior* probability of H based on the known evidence, namely, prob $(H \mid B)$, can be determined by combining prob $(B \mid H)$ with the prior probability of H. I'll spare you the mathematical details of how this comes about (again, these can be found in the *Technical Notes*) Fig. 1.5.

Any number of complications can arise from an inappropriate use of conditional probabilities. There is a class of ill-informed judgments that occur in medical screening and criminal trials, among other settings, in which a conditional probability in one direction is mis-interpreted to mean the

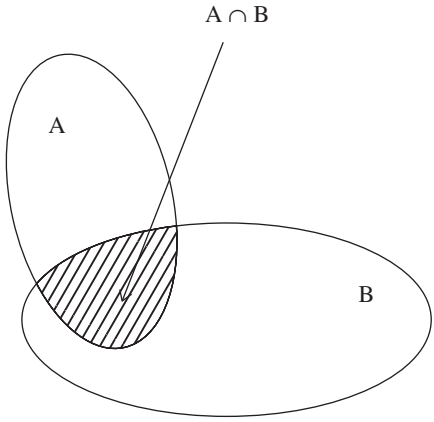

Fig. 1.5 Intersection of events *A* and *B*

opposite conditional. But these two numbers can be vastly different, and this leads to consequences that are at the very least troublesome and often quite serious. This is not the place to explore the egregious uses of probability in everyday affairs, but some useful references to specific examples can be found in the section on *Further Readings*.

Where We Stand Now

In this chapter I took the point of view that randomness is a notion that pertains to an ensemble of possibilities, whose uncertainty is a property of the mechanism that generates these outcomes, and not a property of any individual sequence. Nevertheless we are loath to call a string of 100 digits random if zero is repeated 100 times, and so what we need is a way of looking at what an individual run of zeros and ones can tell us about randomness regardless of where it came from. The shift in emphasis consists in looking at the arrangement of a fully developed string by itself and not

on what generated it. This is one of the tasks before us in the subsequent chapters. In particular, it will be seen that randomness can be rejected when there are simple rules that largely predict the successive digits in a string. Champernowne's number that was discussed in the previous section will fail to qualify as random, for example.

The arguments used in the remainder of the book are quite different from those employed in the present chapter in which the normal law for large sample sizes was invoked to test an a priori assumption about the randomness of a string. Although this approach remains part of the core canon of statistical theory, the subsequent chapters will focus less on the randomness of the generating process and more on the patterns that are actually produced. There are a number of surprises in store, and, in particular, we will again be deluded into believing that a digit string is random when the mechanism that generated is not random and vice-versa.

Let us close with a quote from the mathematician Pierre-Simon Laplace, himself an important contributor to the theory of probability in the early part of the nineteenth century, who unwittingly anticipated the need for a new approach: "in the game of heads and tails, if head comes up a hundred times in a row then this appears to us extraordinary, because after dividing the nearly infinite number of combinations that can arise in a hundred throws into regular sequences, as those in which *we observe a rule that is easy to grasp*, and into irregular sequences, the latter are incomparably more numerous."

2

Uncertainty and Information

Norman…looked at a lot of statistics in his life, searching for patterns in the data. That was something human brains were inherently good at, finding patterns in the visual material. Norman couldn't put his finger on it, but he sensed a pattern here. He said, I have the feeling its not random.

from Sphere, by Michael Crichton

Messages and Information

A half-century ago Claude Shannon, mathematician and innovative engineer at what was then called the Bell Telephone Laboratories, formulated the idea of information content residing in a message, and, in a seminal paper of 1948, he established the discipline that became known as information theory. Though its influence is chiefly in communication engineering, information theory has come to play an important role in more recent years in elucidating the meaning of chance.

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2_2

Shannon imagined a source of symbols that are selected, one at a time, to generate messages that are sent to a recipient. The symbols could, for example, be the ten digits 0, 1, …, 9, or the first 13 letters of the alphabet, or a selection of 10, 000 words from some language, or even entire texts. All that matters for the generation of messages is that there be a palette of choices represented by what are loosely designated as "symbols."

A key property of the source is the uncertainty as to what symbol is actually picked each time. The freedom of choice the source has in selecting one symbol among others is said to be its *information content*.

As this chapter unfolds, it will become apparent that maximum information content is synonymous with randomness and that a binary sequence generated from a source consisting of only two symbols 0 and 1 cannot be random if there is some restriction on the freedom the source has in picking either of these digits. The identification of chance with information will allow us to quantify the degree of randomness in a string and to answer the question "is it random" in a different manner from that of the previous chapter.

The simplest case to consider is a source alphabet consisting of just two symbols, 0 and 1, generating messages that are binary strings. If there is an equal choice between the two alternative symbols, we say that the information in this choice is one *bit*. Just think of a switch with two possible positions 0 and 1, in which one or the other is chosen with probability $\frac{1}{2}$. With two independent switches, the number of equally probable outcomes is 00, 01, 10, and 11, and it is said that there are two bits of information. With three independent switches, there are $2^3 = 8$ possible outcomes 000, 001, 010, 011, 100, 101, 110, and 111 each consisting of three bits of information; in general, n switches result

in an equal choice among 2^n possibilities, each coded as a string of n zeros and ones, or n bits.

You can regard the strings as messages independently generated one at a time by a source using only two symbols or, alternatively, the messages may themselves be thought of as symbols each of which represents one of the 2^n equally probable strings of length n. With messages as symbols, the source alphabet consists of 2^n messages, and its information content is n bits, whereas a binary source has an information content of one bit.

There is uncertainty as to which message string encapsulated by n binary digits is the one actually chosen from a message source of 2^n possible strings. As n increases so does the hesitation, and therefore the information content of the source becomes a measure of the degree of doubt as to what message is actually selected. Once picked, however, the ambiguity regarding a message is dispelled since we now know what the message is. Most strings can be expected to have no recognizable order or pattern, but some of them may provide an element of surprise in that we perceive an ordered pattern and not trash. The surprise in uncovering an unexpected and useful string among many increases as the information content of the source gets larger. However order (and disorder) is in the eye of the beholder, and selection of any other string, patterned or not, is just as likely and just as surprising.

The n bits needed to describe the $m = 2^n$ messages generated by a binary source are related by the mathematical expression $n = \log m$ where "log" means "base 2 logarithm of." The quantity $\log m$ is formally defined as the number for which $2^{\log m} = m$. Since $2^0 = 1$ and $2^1 = 2$, it follows from the definition that $\log 1 = 0$ and $\log 2 = 1$. Additional properties of logarithms can be found in Appendix C (if you are not familiar with logarithms, just think of them as a notational device to represent certain numerical expressions involving

exponents in a compact form). I use logarithms sparingly in this chapter but cannot avoid them entirely because they are needed to formalize Shannon's notion of information content. In fact, the information content of m equally-likely and independent choices is defined by Shannon to be precisely log$m$. With $m = 1$ (no choice), the information is zero, but for $m = 2^n$ the information content is n bits. In the event that a source works from an alphabet of k symbols, all equally-likely and chosen independently, it generates a total of $m = k^n$ message strings of length n, each with the same probability of occurring. In this case the information content per string islog$m = n$ log k, while the information content per symbol is just log$k$. With three symbols A, B, and C, for instance, there are $3^2 = 9$ messages of length 2, namely, AA, AB, AC, BA, BB, BC, CA, CB, and CC. The information content of the source is therefore twice log3 or, roughly, 3.17.

The word "information" as used by Shannon has nothing to do with "meaning" in the conventional sense. If the symbols represent messages, a recipient might view one of them as highly significant and another as idle chatter.

The difference between information and meaning may be illustrated by a source consisting of the eight equally-likely symbols A, D, E, M, N, O, R, and S that generate all words having 10 letters. Most of these words are gibberish, but if randomness should come into sight out of the ensemble of $2^{10} = 1024$ possible words, there is good reason to be startled and feel that perhaps we are recipients of an omen. Nevertheless, the word RRRRRRRRRR is just as likely to appear and should surprise us no less. Another example would be a source whose symbols consist of a set of instructions in English. The information content of this source tells us nothing about the significance of the individual symbols. One message might instruct you to open a drawer and read the contents, which turn out to be a

complete description of the human genome. The unpacked message conveys enormous meaning to a geneticist. The next message directs you to put out the cat, which is arguably not very significant (except, of course, to the cat).

Entropy

After these preliminaries we consider m possibilities each uninfluenced by the others and chosen with perhaps unequal probabilities p_i for the ith symbol, i being any integer from 1 to m. The sample space consists of m elementary events "the ith symbol is chosen." These probabilities *sum to one*, of course, since the m mutually exclusive choices exhaust the sample space of outcomes. The *average information content* of this source, denoted by H, was defined by Shannon to be the negative of the sum of the logarithms of the p_i, each weighted by the likelihood p_i of the ith symbol:

$$H = -(p_1 \log p_1 + p_2 \log p_2 + \ldots + p_n \log p_m)$$

or, in more compact form,

$$H = -\text{the sum of } p_i \log p_i$$

The expression H is called the *entropy of the source* and represents the average information content of the source, in *bits per symbol*. Though this definition seems contrived, it is exactly what is needed to extend the idea of information to symbols that appear with unequal frequencies. Moreover it reduces to the measure of information content considered previously for the case in which the m symbols are all equally-likely to be chosen. When the source consists of

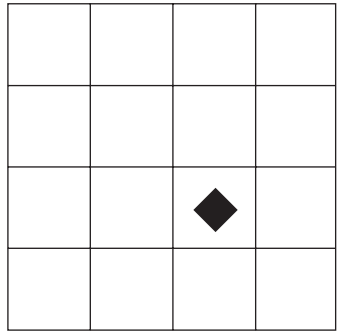

Fig. 2.1 A board divided into 16 squares, all of them empty except for one, which contains an object whose location is to be determined by a blindfolded contestant in a variant of the game "20 questions"

$m = 2^n$ message strings, each having perhaps different probabilities of occurring, H is regarded as the average information content in *bits per message*.

To illustrate the computation of entropy considers a board divided into 16 squares of the same size and suppose you are asked to determine which square (see Fig. 2.1) has some object on it by engaging in a variant of the game "20 questions." You ask the following questions having yes or no answers:

Is it one of the 8 squares on the top half of the board? (No)
Is it one of the 4 squares on the right half of the remaining 8 possibilities? (Yes)
Is it one of the 2 squares in the top half of the remaining 4 possibilities? (Yes)
Is it the square to the right of the 2 remaining possibilities? (No)

Letting one mean yes and zero no, the uncovered square is determined by the string 0110 because its location is determined by no yes yes no. Each question progressively narrows the uncertainty and the amount of information you

receive about the unknown position diminishes accordingly. There are $16 = 2^4$ possible squares to choose from initially, all equally-likely, and therefore 16 different binary strings. The uncertainty before the first question is asked, namely, the entropy, is consequently $\log 16 = 4$ bits per string. The entropy decreases as the game continues.

For a binary source consisting of only two symbols with probabilities p and $1 - p$, the expression for the entropy H simplifies to:

$$H = -\{p \log p + (1-p)\log(1-p)\}$$

Figure 2.2 plots the values of H for a binary source versus the probability p, and we see that H is maximized when

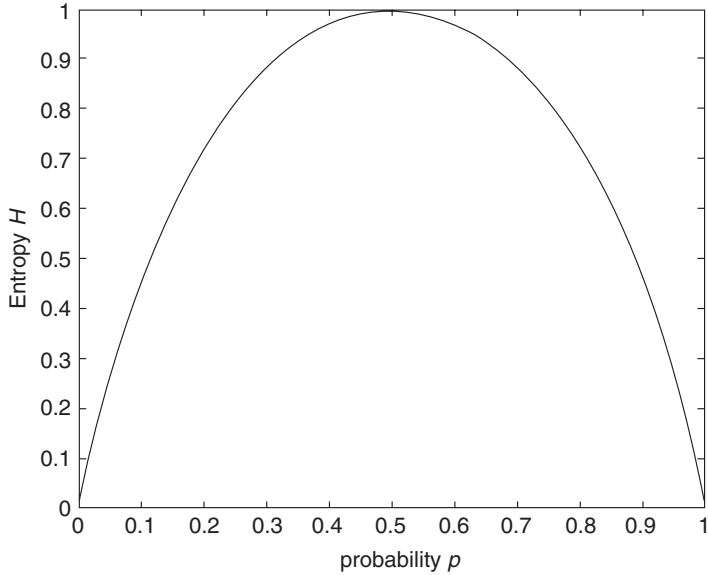

Fig. 2.2 Plot of entropy H versus probability p for a binary source consisting of only two symbols with probabilities p and $1 - p$. Note that H is maximized when p is ½ and is zero when there is no uncertainty (p equal to either 0 or 1)

$p = \frac{1}{2}$ at which point $H = 1$ (since log2 = 1); otherwise H is less than 1. With $p = \frac{3}{4}$, for example, the entropy H is $-$ {.75 log .75 + .25 log .25} which is roughly .81.

In general, for a nonbinary source, it is possible to show that *H is maximized when the m independent choices are equally-likely*, in which case H becomes log$m$, as already noted. In conformity with the discussion in the preceding chapter, *maximum entropy is identified with quintessential randomness.* This is illustrated by the board game of Fig. 2.1, where the entropy is maximum since all four choices are independent and have an equal probability $\frac{1}{2}$ of being true. The entropy concept will allow us to give new meaning in the next section to the question "Is it random?"

It is important to emphasize that information is equated with uncertainty in the sense that the more unaware we are about some observation or fact, the greater is the information we receive when that fact is revealed. *As uncertainty increases so does the entropy* since entropy measures information. A simple choice among two alternative messages is mildly informative, but having one message confirmed out of many has a degree of unexpectedness and surprise that is very informative. With no choice there is no information, or zero entropy. Message content is irrelevant, however, as we have already stressed. The entropy in a binary source is unaffected by the fact that one symbol represents a thousand page textbook and the other a simple "duh."

Messages, Codes, and Entropy

If the probabilities p and $1 - p$ in a binary source are unequal, then patterns of repeated zeros or ones will tend to recur, and the ensemble has some redundancy. If it is possible to reduce the redundancy by compressing the message, then a shorter binary sequence can serve as a code to

reproduce the whole. In a truly random string, all the redundancy has been squeezed out, and no further compression is possible. The string is now as short as possible.

Because of redundancies, a few messages out of the total generated by a binary source are more likely than others due to the fact that certain patterns repeat. There is therefore a high probability that some small subsets of messages are actually formed out of the total that are possible. The remaining (large) collection of messages has a small probability of appearing from which it follows that one can encode most messages with a smaller number of bits. Of all the messages of length n that can conceivably be generated by Bernoulli p-trials, the chances are that only a small fraction of them will actually occur when p is not equal to $\frac{1}{2}$. Instead of requiring n bits to encode each message string, it suffices to use nH bits to represent all but a (large) subset of strings for a message source of entropy H. The required number of bits per symbol is therefore about $nH/n = H$. This compression in message length is effective only when H is less than 1 (p not equal to $1 - p$) since H equals one when $p = \frac{1}{2}$.

Stated in a slightly different way, all messages of size n can be divided into two subsets, one of them a small fraction of the total containing messages occurring a large percentage of the time and a much larger fraction consisting of messages that rarely appear. This suggests that short codes should be used for the more probable subset and longer codes for the remainder in order to attain a lower average description of all messages. The exception to this, of course, is when $p = \frac{1}{2}$ since all messages are then equally probable.

In order to take advantage of message compression, we must find a suitable encoding, but how to do so less than obvious. I will describe a reasonable, but not maximally efficient, coding here and apply it to a specific sequence. Our interest in exploring this now is in anticipation of the

discussion in Chapter 4 where I follow Andrei Kolmogorov and others who say that a string is random if its shortest description is obtained by writing it out in its entirety. There is no compressibility. If, however, a shorter string can be found to generate the whole, a shorter string that encodes the longer one, then there must have been recognizable patterns that reduced the uncertainty. The approach taken by Kolmogorov is different than Shannon's, as we will see, since it depends on the use of computable algorithms to generate strings, and this brings in its wake the paraphernalia of Turing machines, a topic that is explained in Chapter 4.

Take a binary string that is created by Bernoulli p-trials in which the chance of a zero is .1 and that of a one is .9 (p = .9). Break the string up into consecutive snippets of length 3. Since blocks of consecutive ones are more likely to appear than zeros, assign a 0 to any triplet 111. If there is a single zero in a triplet, code this by a 1 followed by 00, 01, or 10 to indicate whether the zero appeared in the first, second, or third position. In the less frequent case of 2 zeros, begin the code with a prefix of 111 followed by 0001 (first and second positions), 0010 (first and third positions), or 0110 (second and third positions). The rare event of all zeros is assigned 11111111. Consider, for example, the string fragment 110 111 101 010 111 110 111 consisting of seven triplets with a total of 21 bits. These triplets are coded, in sequence, by the words 110, 0, 101, 1110010, 0, 110, and 0. The code requires 19 bits, a slight compression of the original message. In general, the reduction is considerably better since a much larger number of ones will occur on average for actual Bernoulli $^{9}/_{10}$-trials than is indicated by the arbitrarily chosen string used here.

Notice that this coding scheme has the virtue of being a *prefix code*, meaning that none of the code words are prefixes of each other. When you come to the end of a code word, you know it is the end, since no word appears at the beginning of any other code word. This means that there is

a unique correspondence between message triplets and the words that code them, and so it is possible to read the code backward to decipher the message. This is evident in the example above, in which none of the seven code words reappears as a prefix of any of the other words. You can now confidently read the fragment 11001110010 from left to right and find that 1 by itself means nothing, nor does 11. However, 110 means 110. Continuing, you encounter 0, which denotes 111. Proceeding further, you need to read ahead to the septuple 1110010 to uncover that this is the code for 010; nothing short of the full seven digits has any meaning when scanned from left to right. Altogether, the deciphered message is 1101010.

Although the discussion of codes has been restricted to only two symbols, zero and one, similar results apply to sources having alphabets of any size k. Suppose that the ith symbol occurs with probability p_i with $i = 1, \ldots, k$. Shannon established that there is a binary prefix coding of the k symbols in which the average length of the code words in bits per symbol is nearly equal to the entropy H of the source. As a consequence of this, a message of length n can be encoded with about nH bits. Consider, for example, a source with four symbols a, b, c, and d having probabilities $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, and $\frac{1}{8}$, respectively. The entropy is computed to be 1.75, and if the code words are chosen as 0 for a, 10 for b, 110 for c, and 111 for d, then this corresponds to code word lengths of 1, 2, 3, 3 whose average length is 1.75 which, in this case at least, equals the entropy. In the example given in the two preceding paragraphs, the source consisted of the eight triplets 000, 001, …, 111 of independent binary digits in which the probability of 111, for example, is $\frac{9}{10}$ multiplied by itself three times, or about .73, with a similar computation for all the other triplets. Although a reasonably effective code was devised for this

source, it is not as efficient as the code established by Shannon.

In order to attain additional insight into randomness, the assumption that successive symbols are independent can be abandoned by allowing each symbol to be contingent on what the previous symbol or, for that matter, on what several of the previous symbols happened to be. More exactly, the probability that a particular symbol takes on any of the k alphabet values is conditioned on which of these values were assumed by the preceding symbol(s).

For a simple illustration, pick the $k = 2$ case where the alphabet is either yes or no. Suppose that the probability of yes is $\frac{1}{3}$ if the preceding bit is yes and $\frac{1}{4}$ if the predecessor happened instead to be no. The conditional probabilities of getting no, by contrast, are $\frac{2}{3}$ and $\frac{3}{4}$, respectively. This means that if the source emits the symbol yes, then the probability of getting no on the next turn is twice that of obtaining yes. There is a built-in redundancy here due to the correlation between successive bits.

A more striking example of sequential correlations is the English language. English text is a concatenation of letters from a total alphabet of 27 (if one counts spaces), or, taking a more liberal viewpoint, a text is a string of words in which the "alphabet" is now a much larger, but still finite, ensemble of possible words.

The frequency with which individual letters appear is not uniform since E, for example, is more likely than Z. Moreover, serial correlations are quite evident: TH happens often as a pair, for example, and U typically follows Q. To analyze a source that spews forth letters, assume that each letter is produced with different frequencies corresponding to their actual occurrence in the English language. The most frequent letter is E, and the probability of finding E in a sufficiently long text is approximately .126, meaning that its

frequency of occurrence is .126, whereas Z has a frequency of only about .001. The most naive simulation of written English is to generate letters according to their empirically obtained frequencies, one after the other, independently. A better approximation to English is obtained if the letters are not chosen independently but if each letter is made to depend on the preceding letters, though not on the letters before that. The structure is now specified by giving the frequencies of various letter pairs, such as QU. This is what Shannon called the "digram" probabilities of the paired occurrences. After a letter is chosen, the next one is picked in accordance with the frequencies with which the various letters follow the first one. This requires a table of conditional frequencies. The next level of sophistication would involve "trigram" frequencies in which a letter depends on the two that come before it.

Generating words at random using trigram frequencies gives rise to a garbled version of English, but as you move forward with tetragrams and beyond, an approximation to the written word becomes ever more intelligible.

The crudest approximation to English is to generate each of the 27 symbols independently with equal probabilities $\frac{1}{2}$. The average entropy per letter in this case is $\log 27 = 4.76$. By the time one reaches tetragram structure, the entropy per symbol is reduced to 4.1, which shows that considerable redundancy has accumulated as a result of correlations in the language induced by grammatical usage and the many habits of sentence structure accrued over time. Further refinements of these ideas led Shannon to believe that the entropy of actual English is roughly one bit per symbol. This high level of redundancy explains why one can follow the gist of a conversation by hearing a few snatches here and there in a noisy room even though some words or entire phrases are drowned out in the din.

Since redundancy reduces uncertainty, it is deliberately introduced in many communication systems to lessen the impact of noise in the transmission of messages between sender and recipient. This is accomplished by what are called *error-correcting codes* in which the message, say one block of binary digits, is replaced by a longer string in which the extra digits effectively pinpoint any error that may have occurred in the original block. To illustrate this in the crudest possible setting, imagine message blocks of length two designated as b_1b_2 in which each b_i is a binary digit. Let b_3 equal 0 if the block is 00 or 11 and 1 if the block is 01 or 10. Now transmit the message $b_1b_2b_3$ which always has an even number of ones. If an error occurs in transmission in which a single digit is altered as, for example, when 011 is altered to 001, the receiver notes that there is an odd number of ones, a clear indication of an error. With an additional refinement to this code, one can locate exactly at what position the error took place and thereby correct it. Compact disk players that scan digitized disks do an error correction of surface blemishes by employing a more elaborate version of the same idea.

Diametrically opposite to error correction is the deliberate insertion of errors in a message in order to keep unauthorized persons from understanding it. The protection of government secrets, especially in time of warfare, is a remarkable tale that reads like a thriller, especially in David Kahn's *The Codebreakers*. A secure method of enciphering messages for secrecy consists in coding each letter of the alphabet by a five-digit binary string. The $2^5 = 32$ possible quintuplets encompass all 26 letters as well as certain additional markers. A message is then some long binary string s. Let v, called the key, designates a random binary string of the same length as s (v is obtained in practice from a "pseudo-random" generator, as described in the next

chapter). The cyphered message # is obtained from the plaintext message s by adding the key v to s, digit by digit, according to the following rule:

$$0+0=0$$
$$0+1=1$$
$$1+0=1$$
$$1+1=0$$

The recipient of # is able to decode the encrypted message by the same procedure: simply add v to # using the same rule for addition as the one above, and this restores s! For example, if s = 10010 and v = 11011, the transmitted message # is 01001. The decoded message is obtained from 01001 by adding 11011 to obtain 10010, namely, the plaintext s (see Fig. 2.3 for a schematic illustration of the entire communication system). Since the key v is random, and known only to sender and receiver, any spy who intercepts the garbled message is foiled from reading it. For reasons of security, the key is usually used only once per message in this scheme in order to avoid having telltale patterns emerge from the analysis of several surreptitious interceptions.

There is a balance between the inherent structure and patterns of usage in a language like English and the freedom one has to spawn a progression of words that still manage to

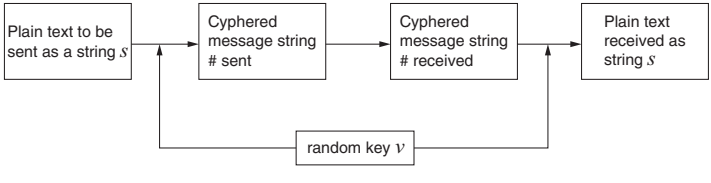

Fig. 2.3 Schematic representation of an encipher-decipher coding scheme for sending messages in a secure manner

delight and surprise. A completely ordered language would be predictable and boring. At the other extreme, the random output of the proverbial monkey banging away at a keyboard would be gibberish. English is rich in nuance and comprehensible at the same time. In Chapter 5 this interplay between chance and necessity becomes a metaphor for the span of human culture and nature at large, in which randomness will be seen as an essential agent of innovation and diversity.

Approximate Entropy

At the end of the previous chapter, I stated the intention of deciding the randomness of a given string regardless of its provenance. We are ignorant of how the string came to be, and we don't care. The mechanism that generated it, whether it be pure happenstance or some hidden design, is irrelevant, and we want to distance ourselves from the source and concentrate on the string itself. Taking a cue from the discussion in the preceding section, the question of randomness will now be made to hinge on the information content of a given binary sequence. A rough and ready tool for assessing randomness in terms of its entropy is called *approximate entropy* (or ApEn, for short).

Suppose that a finite sequence is to some extent patterned by virtue of sequential dependencies as happens when the likelihood of a zero or one is contingent on whether it is preceded by one or more zeros or ones. We calculate the degree of redundancy or, to put it another way, the extent of randomness, by computing an expression analogous to entropy that measures the uncertainty in bits per symbol. If the source generates independent symbols having different probabilities p_i, for $i = 1, 2, \ldots m$, then, as you have already seen, the entropy is given by the

expression $H = -$ the sum of $p_i \log p_i$. However, not knowing the source requires that these probabilities be estimated from the particular sequence before us. The Law of Large Numbers tells us that the p_i is roughly equal to the fraction of times that the ith symbol appears in the given string.

Coming next to *digrams*, namely, pairs of consecutive symbols, an expression for the "entropy" per couple is defined in the same way except that p_i now refers to the probability of obtaining one of the $r = m^2$ possible digrams (where m is the number of symbols and r is the number of digrams formed from these symbols); the expression for H remains the same, but you now must sum over the range $i = 1, 2, \ldots r$. For example, if $m = 3$ with an alphabet consisting of A,B,C, then there are $3^2 = 9$ digrams:

AA BA CA
AB BB CB
AC BC CC

Triplets of consecutive symbols, or *trigrams*, are handled in an analogous manner, allowing for the fact that there are now m^3 possibilities. The same principle applies to *k-grams*, blocks of k symbols, for any k. What I'm striving to establish is that a finite string is random if its "entropy" is as large as possible and if digrams, trigrams, and so forth provide no new clue that can be used to reduce the information content.

For a given string of finite length n, it is necessary to estimate the probabilities p_i for all possible blocks. As before, the Law of Large Numbers assures us that when n is sufficiently large, the value of p_i is roughly equal to the proportion of times that the ith block of length k appears (blocks of length 1 refers to one of the m individual symbols), where i ranges from 1 to m^k. It can be readily determined

that there are exactly $n + 1 - k$ blocks of length k in the string, namely, the blocks beginning at the first, second, …, $(n + 1 - k)$th position. For example, if $n = 7$ and $k = 3$, the blocks of length k in the string ENTROPY are ENT, NTR, TRO, ROP, and OPY, and there are $n + 1 - k = 7 + 1 - 3 = 5$ of them.

Let n_i indicates the number of times that the ith block type occurs among the $n + 1 - k$ successive k-grams in the string. Then the probability p_i of the ith block is estimated as the frequency n_i/N in which we agree to write $(n + 1 - k)$ simply as N. For example, the string

CAAABBCBABBCABAACBACC

has length $n = 21$ and m equals 3. There are $m^2 = 9$ possible digrams, and, since $k = 2$, these are found among the $N = 20$ consecutive blocks of length 2 in the string. One of the 9 conceivable digrams is AB, and this occurs three times, and so the probability of this particular block is 3/20.

It is not hard to see that the n_i must sum to N as i ranges from 1 to m^k. The following approximate expression is then obtained for the "entropy" $H(k)$ per block of size k:

$$H(k) = -\text{the sum of } (n_i / N) \log(n_i / N),$$

where the index i goes from 1 to m^k; $H(1)$ is identical to the usual expression for H. You should be aware, however, that "entropy" is an abuse of language here since entropy as defined earlier for H applies to an alphabet of symbols chosen independently, whereas the blocks of size k may be correlated due to sequential dependencies. Moreover, entropy assumes there is some specific generating source, while now there is only a single string from some unknown source,

and all we can do is estimate the source from the roughly estimated probabilities.

I can now introduce the key notion of *approximate entropy* ApEn(k) as the difference $H(k) - H(k-1)$, with ApEn(1) being simply $H(1)$. The idea here is that we want to estimate the "entropy" of a block of length k conditional on knowing its prefix of length $k-1$. This gives the *new information contributed by the last member of a block given that we know its predecessors within the block*. If the string is highly redundant, you expect that the knowledge of a given block already largely determines the succeeding symbol, and so very little new information is accrued. In this situation the difference ApEn will be small. This can be illustrated for digrams schematically in Fig. 2.4, in which the ovals represent the average entropy of a pair of overlapping (sequentially dependent) symbols in one case and disjoint (independent) symbols in the other. Individual ovals have "entropies" $H(1)$, while their union has "entropy" $H(2)$.

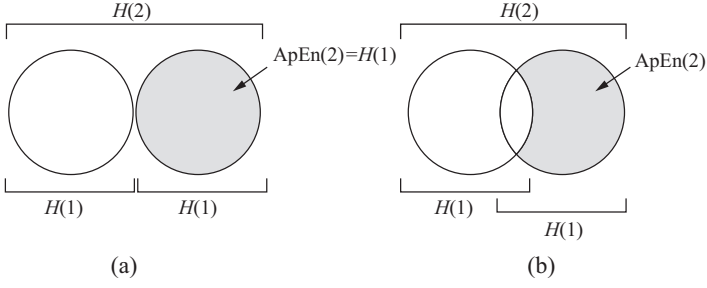

Fig. 2.4 Schematic representation of two successive symbols that are sequentially independent (**a**) and dependent (**b**). The degree of overlap indicates the extent of sequential dependence. No overlap means independence. The average "entropy" per disk (read: symbol) is $H(1)$, and the average "entropy" of the digram is $H(2)$, while the shaded area represents the information contributed by the second member of the digram, which is not already contained in its predecessor, conditional on knowing the first member of the pair, namely, $H(2) - H(1) = \text{ApEn}(2)$

The shaded area in diagram (a) represents the new information contributed by the second member of the digram that is not already contained in its predecessor. This is the approximate entropy ApEn(2) of the digram, conditional on knowing the first member of the pair. The shaded area in diagram (b), and therefore the uncertainty, is maximal whenever the ovals do not overlap.

Another way of seeing why ApEn(2) is close to zero when the second digit in a digram pair is completely determined by the preceding one is that there are about as many distinct digrams in this situation as there are distinct digits and so $H(1)$ and $H(2)$ are about equal; hence their difference is close to zero. By contrast, if a string of length n is random, then $H(1) = \log n$ and $H(2) = \log n^2 = 2 \log n$, since there are n^2 equally probable digrams; therefore ApEn(2) = $2 \log n - \log n = \log n$, which equals ApEn(1).

Again, Is It Random?

For strings that are not too long, it usually suffices to check ApEn (k) for k not exceeding 3 to get an idea of the degree of redundancy. There is a simple algorithm to compute ApEn for binary strings, written as a MATLAB program, and I illustrate it here for three different sequences of length 24 with k equal to 1 and 2. The first one repeats the motif of 01, and so the patterns 00 and 11 never appear. The second sequence is from Bernoulli $\frac{1}{2}$-trials in which some redundancy is expected due to long blocks of consecutive zeros. The last sequence is from Bernoulli $\frac{1}{2}$-trials, and here one may anticipate something closer to maximum "entropy." Since these strings progress from orderly to scrambled, the values of ApEn should reflect this by increasing, and this is precisely what happens. Even though we happen

to know the provenance of each series, they could have appeared to us unannounced, and so it is at least conceivable that each could have arisen from a totally ordered or, perhaps, a hopelessly tangled process. We know not which. The sequences of digits are taken as they are, au naturel, without prejudice as to the source, and we ask to what extent the mask they present to us mimics the face of randomness. The approximate entropy is maximized whenever any of the length k blocks of digits appear with equal frequency, which is the signature requirement of a *normal number* as it was presented in the previous chapter.

It is now clear that in order for an unlimited binary sequence to qualify as random, it must indeed be a normal number. Otherwise, some blocks occur more frequently than others, and, as in the case of Bernoulli p-processes in which p differs from $\frac{1}{2}$, the unequal distribution of blocks results in redundancies that can be exploited by a more efficient coding. In particular, a random sequence has the property that the limiting frequencies of zeros and ones are the same. I must caution, however, that when you look at a string that is finite in length there is no way of knowing for sure whether this snippet comes from a random process or not. The finite string is regarded simply on its own terms. By contrast, the statistical procedures of Chapter 1 tested the null hypothesis that the string is randomly generated, namely, that it comes from Bernoulli $\frac{1}{2}$-trials, but an ineluctable element of uncertainty remains even if the null hypothesis is not rejected since all that it was capable of checking is the frequency of blocks of length one.

Returning to the examples, let the first string consist of 01 repeated 12 times. The computed values of ApEn(1) and ApEn(2) are 1.000 and .001, respectively. Since there is exactly the same number of ones and zeros, ApEn(1) takes on the maximum value of log2 = 1, but ApEn(2) is nearly zero

because the second digit of any digram is completely determined by the first, and there are no surprises.

The next string is

$$000100010000100000100100$$

and we find that ApEn(1) = .738 and ApEn(2) = .684. There are many more zeros than ones which explains why ApEn(1) is less than the theoretical maximum entropy of 1 that would have prevailed if the these digits were equally distributed in frequency. Also, the value of ApEn(2) indicates that the second member of each digram is only partially determined by its predecessor. Zero usually follows a 0, but sometimes there is a surprise; on the other hand, 0 always follows a 1.

The third example, from Bernoulli $\frac{1}{2}$-trials, is

$$111100100010111010110010$$

and here ApEn(1) = .9950, while ApEn(2) = .9777, both close to the maximum entropy of 1. In this example the proportions of ones and zeros are more nearly balanced than they are in the previous example, and this is also the case for each of the digrams 00, 01, 10, and 11.

For purposes of comparison, let us reconsider the string in Chapter 1whose randomness was rejected using De Moivre's theorem. The string consisted of 25 digits

$$1110110001101111111011110$$

Application of ApEn to this sequence gives ApEn(1) = .8555 and ApEn(2) = .7406, leaving questionable again the null hypothesis of randomness. It is worth noting that though the patterned string of repeated 01 fooled de Moivre, ApEn

provides a more stringent test since ApEn(2) is nearly zero, as you saw (actually, the example in Chapter 1 had 25 digits instead of 24, but the conclusion remains the same).

It is only fair to add that the entropy test for randomness may require the application of ApEn for higher values of k in order to detect any latent regularities. For instance, a sequence in which 0011 is repeated six times has the property that 0 and 1 and 00, 01, 10, and 11 all appear with equal frequency and so it fools ApEn(k) into a premature judgment of randomness if k is limited to one or two. It requires ApEn(3) to uncover the pattern. In fact, ApEn(1) is 1.000, ApEn(2) is .9958, but ApEn(3) equals .0018!

As a footnote, it is worth noting that although a random number must be *normal*, not every normal number is necessarily random. Champernowne's example from Chapter 1 is generated by a simple procedure of writing 0 and 1 followed by all pairs 00 01 10 11, followed by all triplets 000 001 etc., and this can be encoded by a finite string that provides the instructions for carrying out the successive steps. This will be made clear in Chapter 4, but for now it is enough to state that a sufficiently long stretch of this normal number fails to be random precisely because it can be coded by a shorter string.

The Perception of Randomness

In the previous chapter, I mentioned that psychologists have concluded that people generally perceive a binary sequence to be random if there are more alterations between zeros and ones than is warranted by chance alone. Sequences produced by a Bernoulli $\frac{1}{2}$ -process, for example, occasionally exhibit long runs that run counter to the common intuition that chance will correct the imbalance by more frequent reversals.

The psychologists Ruma Falk and Clifford Konold put a new spin on these observations in a study they recently conducted. A number of participants were asked to assess the randomness of binary sequences that were presented to them, either by visual inspection or by being able to reproduce the string from memory. It was found that the perception of randomness was related to the degree of difficulty the subjects experienced as they attempted to make sense of the sequence. To quote the authors, "judging the degree of randomness is based on a covert act of encoding the sequence. Perceiving randomness may, on this account, be a consequence of a failure to encode," and, elsewhere, "the participants tacitly assess the sequences's difficulty of encoding in order to judge its randomness." Of the two sequences

1 1 1 1 1 1 0 0 0 1 1 0 0 0 0 0 0 0 1 1 1
1 1 0 1 0 1 0 1 0 1 0 0 0 1 1 0 1 0 1 0 1 ,

the second, with its excess of alternations, is perceived as more random, even though each string departs equally from the number of runs that a truly random string would be expected to have. It is simply more difficult to reproduce the second from memory.

The experimental results are not inconsistent with the idea that a low value of ApEn betrays a patterned sequence whose sequential redundancies can be favorably exploited by employing a suitable code. Actually, the subjects displayed a systematic bias in favor of perceiving randomness in strings with a moderately higher prevalence of alterations than the maximum value of ApEn would indicate.

Falk and Konold conclude their paper with a comment on "the interconnectedness of seeing the underlying structure (i.e., removing randomness) and hitting upon an

efficient encoding. Learning a foreign language is such an instance; forming a scientific hypothesis is often another…once a pattern has been recognized, the description of the same phenomenon can be considerably condensed." In a similar vein, the psychologists Daniel Kahneman and Amos Tversky argue that "random-appearing sequences are those whose verbal descriptions is longest." These assertions echo the sentiments of Laplace quoted at the end of the previous chapter, sentiments that resonate loudly in the remainder of this book. In particular, the complexity of a string as an indication of how difficult it is to encode will ultimately be recast as a precise measure of randomness in Chapter 4.

To summarize, the idea of entropy was introduced in this chapter as a measure of uncertainty in a message string, and we saw that whatever lack of randomness there is can be exploited by a coding scheme that removes some of the redundancy. But the story is far from complete. Entropy and information will reappear in the next chapter to help provide additional insights into the question "what is random?"

3

Janus-Faced Randomness

That's the effect of living backwards, the Queen said kindly: it always makes one a little giddy at first … but there's one great advantage in it that one's memory works both ways

The other messenger's called Hatta. I must have two, you know-to come and go. One to come and one to go… Don't I tell you? the King repeated impatiently. I must have two-to fetch and carry. One to fetch and one to carry

from Through The Looking Glass by Lewis Carroll

Is Determinism an Illusion?

The statistician M. Bartlett has introduced a simple step-by-step procedure for generating a random sequence that is so curious it compels us to examine it carefully since it will bring us to the very core of what makes randomness appear elusive.

Any step-by-step procedure involving a precise set of instructions for what to do next is called an *algorithm*. Bartlett's algorithm produces numbers one after the other

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2_3

using the same basic instruction, and for this reason, it is also referred to as an *iteration*. It begins with a "seed" number u_0 between zero and one and then generates a sequence of numbers u_n, for $n = 1, 2, \ldots$, by the rule that u_n is the previous value u_{n-1} plus a random binary digit b_n obtained from Bernoulli $\frac{1}{2}$-trials, the sum divided by two; the successive values of b_n are independent and identically distributed as 0 and 1. Put another way, u_n can take on one of the two possible values $u_{n-1}/2$ or $\frac{1}{2} + u_{n-1}/2$, each equally probable.

The connection between two successive values of u_n is illustrated in Fig. 3.1, in which the horizontal axis displays the values of u_{n-1} and the vertical axis gives the two possible values of the successor iterate u_n.

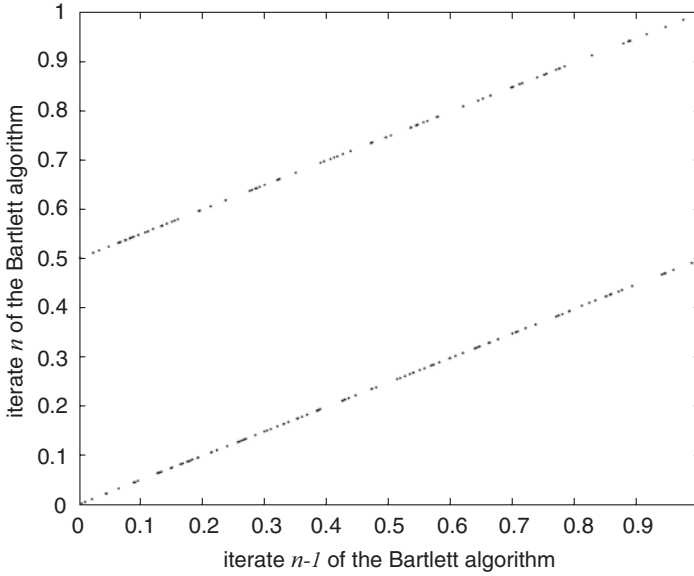

Fig. 3.1 Successive values (u_{n-1}, u_n) of the Bartlett algorithm for 200 iterates. For each value of u_{n-1}, there are two possible values of u_n, each equiprobable

To make further progress with this sequence and to reveal its essential structure, it is convenient, indeed necessary, to represent the successive numbers by a binary string. Any number x between zero and one can be represented as an infinite sum:

$$x = \frac{a_1}{2} + \frac{a_2}{4} + \frac{a_3}{8} + \frac{a_4}{16} + \ldots$$

in which the coefficients a_1, a_2, a_3, … can be either zero or one. Why this is so is discussed in detail in Appendix B, but for now it is enough to give a few examples. The number $^{11}/_{16}$, for instance, can be expressed as the finite sum $^{1}/_{2} + {}^{1}/_{8} + {}^{1}/_{16}$, while the number $^{5}/_{6}$ is an unending sum $^{1}/_{2} + {}^{1}/_{4} + {}^{1}/_{16} + {}^{1}/_{64} + {}^{1}/_{256} + \cdots$ .The coefficients a_1, a_2, … in the infinite sum can be strung out to form a binary sequence, and this is the representation that I am looking for. In the case of $^{11}/_{16}$, the binary sequence is represented as 1011000… with all the remaining digits being 0 (since $a_1 = 1$, $a_2 = 0$, $a_3 = 1$, $a_4 = 1$, and so on), while $^{5}/_{6}$ is represented as 11010101…, in which the 01 pattern repeats indefinitely. In the same manner, any number x in the *unit interval* (meaning the numbers between zero and one) can be represented as a binary sequence $a_1a_2a_3$…; this applies, in particular, to the initial value u_0.

The trick now is to see how the first iterate u_1 of Bartlett's algorithm is obtained from u_0 in terms of the binary representation. The answer is that the sequence a_1a_2… is shifted one place to the *right*, and a digit b_1 is added to the left. In effect, u_1 is represented by $b_1a_1a_2a_3$ … and, mutatis mutandi, by shifting the sequence to the right n places and adding random digits on the left, the nth iterate is expressed as $b_nb_{n-1}\ldots b_1a_1a_2a_3\ldots$. A simple example shows how this works: pick the initial seed number u_0 to be $^{1}/_{4}$, and

suppose that the random digits b_1 and b_2 are, respectively, 0 and 1. Then $u_1 = \frac{1}{8}$ and $u_2 = \frac{9}{16} = \frac{1}{2} + \frac{1}{16}$. The binary representations of u_0, u_1, and u_2 are now found to be 01000…, 001000…, and 1001000… in accord with what was just described.

The clever scheme of replacing the iterations of u_n by the easier and more transparent action of shifting the binary sequence that represents u_n is known as *symbolic dynamics*. Figure 3.2 is a schematic representation.

At this point we should pause and note a most curious fact: Bartlett's randomly generated sequence is a set of numbers that, when *viewed in reverse*, is revealed as a deterministic process! What I mean is that if you start with the last known value of u_n and compute u_{n-1} in terms of u_n and then u_{n-2} in terms of u_{n-1} and so on, there is an unequivocal way of describing how the steps of Bartlett's iteration can be traced backward. The iterates in reverse are obtained by shifting the binary sequence that represents u_n to the *left* one step at a time and truncating the left-most digit: the sequence $b_n b_{n-1} \ldots b_1 a_1 a_2 \ldots$ therefore becomes $b_{n-1} b_{n-2} \ldots b_1 a_1 a_2 \ldots$ this is shown schematically in Fig. 3.3.

This shifting operation does not entail the creation of random digits as it does in Bartlett's iteration, but, instead,

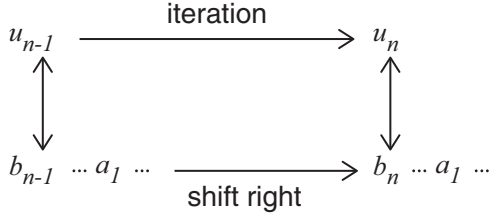

Fig. 3.2 Representation of Bartlett's algorithm using symbolic dynamics: Iteration from u_{n-1} to u_n is tantamount to a shift of binary sequence $b_{n-1} \ldots a_1 \ldots$ to the right and then adding the digit b_n to the right and then adding the digit b_n to the left

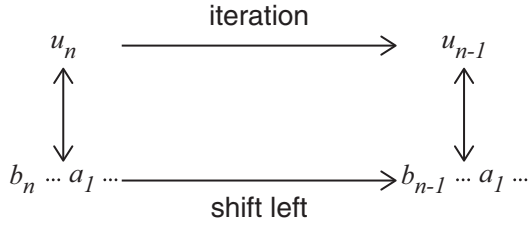

Fig. 3.3 Representation of the inverse of the Bartlett algorithm using symbolic dynamics: Iteration from u_n to u_{n-1} is tantamount to a shift of the binary sequence $b_n \ldots a_1 \ldots$ to the left and then deleting the leftmost digit

it simply deletes existing digits, and this is a perfectly mechanical procedure.

We can now step back a moment to see what this inverse operation actually amounts to. Pick a new seed number v_0 in the unit interval and now generate a sequence v_n for $n = 1, 2, \ldots$ by the iterative rule that v_n is the *fractional part of twice* v_{n-1}. Taking the fractional part of a number is often expressed by writing "mod 1" and so v_n can also be written as $2v_{n-1} \pmod 1$. For instance, if v_{n-1} equals $7/8$, then v_n is obtained by doubling $7/8$, which yields $14/8$, and taking the fractional part, namely, $3/4$. Now suppose that v_0 has, as u_0 did earlier, an infinite binary representation $q_1 q_2 q_3 \ldots$. It turns out that the action of getting v_n from its predecessor is represented symbolically by lopping off the left-most digit and then shifting the binary sequence to the left: $q_1 q_2 q_3 \ldots$ becomes $q_2 q_3 q_4 \ldots$. This is illustrated by the sequence of three Bartlett iterates given earlier, namely, $1/4$, $1/8$, and $9/16$. Beginning with $9/16$ as the value v_0 and working backward by the "mod 1" rule you get $v_1 = 18/16 \pmod 1 = 1/8$ and $v_2 = 1/4 \pmod 1 = 1/4$. In terms of binary representations, the sequence of iterates is 1001000…, 001000…, and 01000….

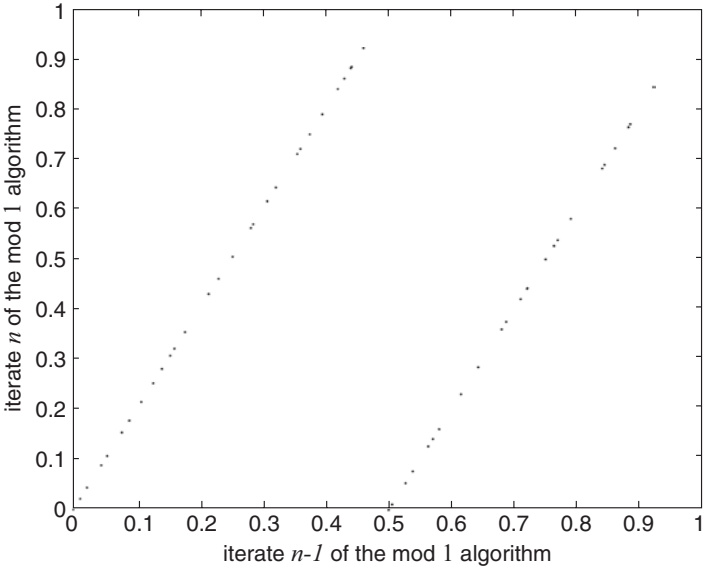

Fig. 3.4 Successive values (v_n, v_{n-1}) of the inverse Bartlett algorithm, namely, the mod 1 algorithm, for 200 iterates. For each value of v_n, there is a unique value of v_{n-1}, a deterministic relationship

It is now apparent that the mod 1 iteration is identical in form to the inverse of the Bartlett iteration and you can again ascertain that this inverse is deterministic since it computes iterates by a precise and unambiguous rule.

Figure 3.4 plots the relation of v_n to v_{n-1}, and by tilting your head, it becomes clear that this picture is the same as that in Fig. 3.1 viewed in reverse. All you need to do is to relabel v_n, v_{n-1} as u_{n-1}, u_n in that order, as shown in Fig. 3.5.

So now we have the following situation: whenever Bartlett randomly generates a binary string $b_n \ldots b_1 a_1 \ldots$, the mod 1 algorithm reverses this by deterministically deleting a digit to obtain the sequence $b_{n-1} \ldots b_1 a_1 \ldots$. In effect the future unfolds through Bartlett by waiting for each new zero or one event to happen at random, while the inverse

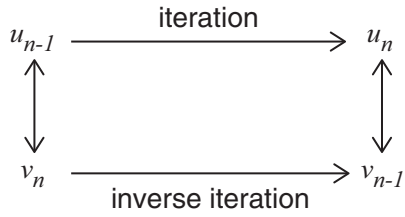

Fig. 3.5 The pair (v_n, v_{n-1}) of Fig. 3.4 corresponds to the pair (u_{n-1}, u_n) of Bartlett's algorithm (Fig. 3.1), whose iterates constitute the *Janus sequence*. The iterates defined by the inverse Bartlett algorithm, namely, the mod 1 algorithm, define the *inverse Janus sequence*

forgets the present and retraces its steps. The uncertain future and a clear knowledge of the past are two faces of the same coin. This is why I call the Bartlett iterates a *Janus sequence*, named after the Roman divinity who was depicted on coins with two heads looking in opposite directions, at once peering into the future and scanning the past.

Although this Janus-faced sequence appears to be a blend of randomness and determinism, a more careful scrutiny reveals that what poses as order is actually disorder in disguise. In fact the string in reverse generates what has come to be known as *deterministic chaos* in which any uncertainty in initial conditions inevitably translates into eventual randomness. In order to see this, let us return to the way the mod 1 iterates shift a binary sequence $q_1q_2\ldots$ to the left. If the string $q_1q_2\ldots$ corresponding to the initial condition v_0 happens to be known in its entirety, then there is no uncertainty about the number obtained after each shift—it is the number that masquerades as the sequence $q_2q_3\ldots$.

The problem arises from what is called *coarse graining*, namely, when v_0 is known only to limited precision as a *finite string* $q_1\ldots q_n$. The reason is that all infinite binary strings can be regarded as outputs of an unending Bernoulli

$\frac{1}{2}$ -trials (Appendix B elaborates on this correspondence), and, therefore, if v_0 is known only to the finite precision $q_1 \ldots q_n$, the remaining digits of the sequence comprise the undetermined remnants of Bernoulli $\frac{1}{2}$ -trials. After n iterates the first n digits have been truncated, and the string now looks like $q_{n+1}q_{n+2}\ldots$, and this consists of a sequence of zeros and ones about which all we know is that they are *independent and equally probable*; from here on out the mod 1 iterates are unpredictable. If q_{n+1} is zero, then the n + 1st iterate is a number in $\left[0, \frac{1}{2}\right)$, whereas if q_{n+1} equals one, this tells you that the iterate lies somewhere within $\left[\frac{1}{2}, 1\right]$; each event has the same probability of occurring.

The seemingly deterministic behavior of the reverse Janus iterates (mod 1 iteration) is evidently compromised unless one knows the present value v_0 exactly as an infinite sequence $q_1q_2\ldots$. Moreover, a subtle change in the initial conditions, a simple flip from 0 to 1 in one of the digits, can result in a totally different sequence of iterates. Dynamical theorists call the mod 1 algorithm, namely, the reverse of Janus, *chaotic* because of this sensitivity to initial conditions: a small change is magnified by successive iterates into unpredictability; *ignorance now spawns randomness later.*

One can produce a random sequence of iterates r_n, n = 1, 2, …, using the mod 1 algorithm directly. Starting with a v_0 whose binary expansion corresponds to some random Bernoulli sequence, set r_n to 0 if the nth iterate is a number in $\left[0, \frac{1}{2}\right)$, and put it equal to 1 if the number is within $\left[\frac{1}{2}, 1\right]$. A little reflection shows that even though the successive values of r_n are generated by a purely mechanical procedure, they are identical to the digits q_n given earlier. The iterates thus define a random process. This is a startling revelation because it suggests that determinism is an illusion. What is going on here?

The solution to the apparent paradox has already been given: if v_0 is known completely in the sense of knowing all its binary digits q_n, then the "random" process generated by the mod 1 algorithm collapses into a futile enumeration of an unending and predestined supply of zeros and ones, and determinism is preserved. The problem, as I said earlier, is to truly possess the digits of v_0's binary representation in their entirety, and, as a practical matter, this eludes us; the curse of coarse graining is that randomness cannot be averted if v_0 is imperfectly known. On a more positive note, however, it may take many iterates before the unpredictability manifests itself, long enough to provide some temporary semblance of order.

Generating Randomness

The deterministic nature of mod 1 is unmasked in other ways, the most convincing of which is the plot in Fig. 3.4, which shows that successive iterates follow each other in an orderly pattern, a consequence of the fact that v_n is dependent on v_{n-1}. By contrast, the successive values of a random sequence are uncorrelated and appear as a cloud of points that scatter haphazardly. They can be obtained by using what is known as a *random number generator* that comes bundled today with many software packages intended for personal computers. They supply "random" numbers by an algorithm not much different in principle from the one employed by the mod 1 procedure: an initial seed is provided as some *integer* x_0, and then integers x_n for $n = 1, 2, \ldots$ are produced by a deterministic iterative scheme in which the updated integer is taken mod m. Since the generated integers don't exceed m, it is fairly evident that at most m distinct integer values of x_n can be produced and so the

iterations must eventually return to some previous number. From here on the procedure repeats itself.

Each integer iterate x_n produced by the random number generator is converted to a fraction within the unit interval by dividing x_n by m. Numbers x_n/m, which lie between 0 and 1, are represented by finite length binary strings, and these may be thought of as truncations of an infinite string whose remaining digits are unknown. This coarse graining, a necessary shortcoming of the limited span of computer memory, is what gives the algorithm its cyclical nature since it is bound to return to a previous value sooner or later. The idea is to obtain a supply of digits that don't cycle back on themselves for a long time, and at the very least, this requires that m be large enough. In many versions of the algorithm, m is set to $2^{31} - 1$, indeed quite large, and if the iterative scheme is chosen judiciously, every integer between 0 and $m - 1$ will be obtained once before recycling.

The recurrence of digits shows that the iterates are less than random. In spite of this inherent flaw, the sequence of x_n values typically manages to pass a battery of tests for randomness. The numbers thus qualify as *pseudo-random*, satisfactory enough for most applications. Many users are lulled into accepting the faux unpredictability of these numbers and simply invoke an instruction like "x = rand" in some computer code (such as MATLAB) whenever they need to simulate the workings of chance.

Even more sophisticated random number generators betray their inherent determinism, however. If one plots successively generated values as pairs or triplets in a plane or in space, orderly spatial patterns appear as striations or layers. "Random numbers fall mainly in the planes" is the way mathematician George Marsaglia once put it. This is discussed further in the *Technical Notes*.

Just how random, incidentally, is the Janus algorithm? Though the successive iterates are generated by a chance mechanism, they are also sequentially correlated. This implies a level of redundancy that the ApEn test of the previous chapter should be able to spotlight. Twenty distinct iterates of the Janus, namely, Bartlett's, algorithm starting with $u_0 = .15$ were used, and an application of ApEn(1) to this string gives 4.322, which is simply the entropy of a source alphabet of 20 equally-likely symbols ($\log 20 = 4.322$). However ApEn(2) is merely .074, another clear indication that substantial patterns exist among the iterates.

A truly random sequence of Bernoulli $\frac{1}{2}$-trials consists of equiprobable blocks of binary digits of a given size. This would also have to be true in reverse since a simple interchange between 0 and 1 results in mirror images of the same blocks. *Therefore, any test for randomness in one direction would have to exhibit randomness when viewed backwards.*

Janus and the Demons

The "demons" in question are hypothetical little creatures invoked in the last century to grapple with paradoxes that emerged during the early years of thermodynamics.

The nineteenth-century dilemma was that an ensemble of microscopic particles moving deterministically in some confined vessel according to Newton's laws of motion needed to be reconciled with Ludwig Boltzmann's idea that the entire configuration of particles moves, on the average, from highly ordered to disordered states. This inexorable movement toward disorder introduces an element of irreversibility that seemingly contradicts the reversible motion of individual particles, and this observation was at first seen as a paradox. In the heated (no pun intended) debate that

ensued, the physicist James Clerk Maxwell introduced a little demon in 1871 who allegedly could thwart irreversibility and thereby dispel the paradox.

The demon, whom Maxwell referred to as a "very observant and nimble-fingered fellow," operates a small doorway in a partition that separates a gas in thermal motion into two chambers. The tiny creature is able to follow the motion of individual molecules in the gas and permits only fast molecules to enter one chamber while allowing only slow ones to leave. By sorting the molecules in this manner, a temperature difference is created between the two portions of the container, and, assuming that initially the temperature was uniform, order is created from disorder (by this I mean that a temperature difference can be exploited to generate organized motion, whereas the helter-skelter motion of molecules at uniform temperature cannot be harnessed in a useful manner). In 1929 the physicist Leo Szilard refined this fictional character and made a connection to the idea of information and entropy which, of course, dovetails with one of the themes in this book.

Before seeing how Szilard's demon operates, we need to backtrack a little. Classical physics tells us that each molecule in the container has its future motion completely determined by knowing exactly its present position and velocity. For simplicity I confine myself to position only and divide up the space within the box containing the billions of molecules of the gas into a bunch of smaller cells of equal volume and agree to identify all molecules within the same cell. In this manner a "coarse graining" is established in which individual particles can no longer be distinguished from one another except to say that they belong to one of finite number N of smaller cells within the container.

As the molecules bounce off of each other and collide with the walls of the box, they move from one cell to

another, and any improbable initial configuration of molecules, such as having all of them confined to just one corner of the enclosure, will over time tend to move to a more likely arrangement in which the molecules are dispersed throughout the box in a disorderly fashion. Although it is remotely conceivable that just the opposite motion takes place, Boltzmann established, as mentioned earlier, that on average the motion is always from an orderly to less orderly arrangement of molecules. This is one version of the *Second Law of Thermodynamics*, and it is supported by the observation that an ancient temple neglected in the jungle will, over time, crumble into ruins, a heap of stones.

The probability p_i of finding a molecule in the ith cell at some particular time is, for $i = 1, 2, \ldots, N$, very nearly n_i/T according to the Law of Large Numbers, for N large enough, where n_i is the number of molecules actually located in that cell and T is the total number of molecules within all N cells. In the previous chapter, we found that the entropy H per cell is the negative of the sum over all cells of $p_i \log p_i$. What Boltzmann established, in effect, is that H can never be expected to decrease and its maximum is attained when the molecules are equi-distributed among the cells ($p_i = 1/N$ for all); total disorder is identified with maximum entropy and randomness. This "one-way street" is the paradox of irreversibility.

The exact location of each molecule at some particular time is specified by its three position coordinates, and these numbers can be represented by infinite binary strings, as we've seen. However, the coarse graining of molecules into N cells means that the exact position is now unknown, and the location of a molecule is characterized instead by a binary string of finite length which labels the particular cell it happens to find itself in, much the same as in the game of "20 questions" of Chapter 2 where an object that is put on a board divided into 16 squares is determined by 4 binary

digits. It is this imprecision that undermines the determinism; there is no uncertainty without coarse graining.

An analogy with the random arrangement of binary strings may be helpful here. Think of the zeros and ones as representing the heads and tails of a tossed coin. At the macroscopic level of an individual string, we know only the total number of heads and tails but not their arrangement within the string. This is the analogy: the molecules within a cell can be arranged in any number of ways, but at the level of cells, we only know how many there are but not their whereabouts. Suppose one is initially given a string of n zeros out of an ensemble of $N = 2^n$ possible binary strings of length n. This unlikely arrangement can happen in only one way and corresponds to an improbable and uniquely distinguishable configuration. However the number of cells that have an equal number of heads and tails is a very large quantity, even for moderate values of n. Essentially it comes to this: there are many more ways of getting strings with an equal number of 0 and 1 than there is in finding a string of all zeros. With n equal to 4, for example, there is a single string 0000, but six strings are found in which 0 and 1 are in the same proportion, namely, 0011, 0101, 0110, 1001, 1010, and 1100. As n increases the discrepancy between ordered (patterned) and disordered (random) strings grows enormously. Just doubling n to 8, for instance, results in 70 strings that are macroscopically indistinguishable since the arrangement of heads and tails is blurred at this level.

However, if the molecules begin naturally in a more probable configuration, then irreversibility is less apparent; indeed it is possible to observe a momentary move toward increased order! It is the artifice of starting with a contrived (human made) and highly unusual configuration that creates the mischief. In the normal course of un-manipulated events, it is the most probable arrangements that one expects to see. Humpty-dumpty sitting precariously on a wall

is a rare sight; once broken into many fragments, there is little likelihood of his coming together again without some intervention, and even all the King's men may not be able to help.

This brief discussion of irreversibility provides another illustration that a deterministically generated sequence can behave randomly because of our ignorance of all but a finite number of binary digits of an initial seed number. In the present case, this is caused by truncating the binary sequences that describe the exact position of each molecule, herding all of them into a finite number of cells.

Let's now return to Szilard and his demon and finally make a connection to the Janus algorithm. A single molecule is put in a container whose two ends are blocked by pistons and that features a thin removable partition in the center (Fig. 3.6). Initially the demon observes which side of the partition the molecule is on and records this with a binary digit, 0 for the left side and 1 for the right. The piston on the side not containing the molecule is gently pushed in toward the partition, which is then removed. This can be accomplished without an expenditure of energy since the chamber through which the piston is compressed is empty.

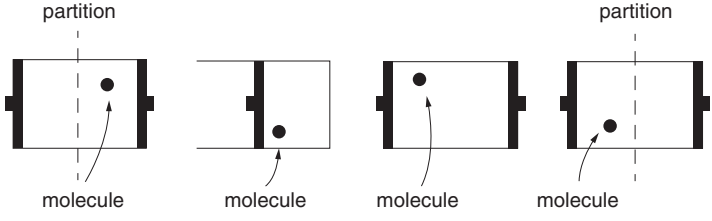

Fig. 3.6 The Szilard engine. A demon observes a molecule in the chamber on the right and gently pushes the piston on the left toward the center partition, which is then removed. The molecule in motion impinges on the piston, sliding it back to the left. This creates useful work and lowers the entropy. The cycle then begins anew

Moreover, Szilard reasoned that moving the partition can be done with negligible work. The energetic molecule now impinges on the piston and moves it back to where it was. Thermal motion creates useful work, a reversal of the usual degradation of organized motion into the disordered motion of heat, and a seeming violation of the Second Law of Thermodynamics. Szilard figured that this decrease in entropy must be compensated by an equivalent increase in entropy due to the act of measurement by the demon to decide which side of the partition the molecule is on. The contemporary view is that what raises the entropy is the act of erasing the measurement after the piston has returned to its original position. Consider this: when one bit is erased, the information, and therefore the entropy, is increased since there is now uncertainty as to whether the digit was 0 or 1 prior to erasure.

If the demon does not erase after each movement of the piston, the entropy certainly decreases, but one is left with a clutter of recorded digits that appears as a disordered binary string of length n. To return the system to its original state requires that this string be blotted out, and, in so doing, n bits are lost, which is tantamount to an increase of uncertainty measured by an entropy of $\log n$. In this manner the Second Law of Thermodynamics maintains its integrity, and it reveals why Maxwell's "little intelligence" cannot operate with impunity. In his stimulating book *Fire In The Mind*, George Johnson tells of an imaginary vehicle powered by a "Szilard Engine" whose piston is cycled back and forth by the demon using information as a fuel. The exhaust, a stream of binary digits, pollutes the environment and raises the entropy.

The operation of the demon is mimicked by the action of Janus, shifting a binary sequence (representing an initial configuration) to the right and inserting a binary digit to the left. After n iterations a binary string of length n is

created to the left of the initial sequence, and it is read from right to left. Initially the n digits are unknown and can be ordered in any one of the 2^n possibilities. Each iterate corresponds to a measurement that decreases the uncertainty and lowers the entropy. After k iterates of Janus have taken place, the information content of the string has been reduced to $n - k$ for $k = 1, 2, \ldots, n$ because only 2^{n-k} strings remain to be considered. At the end, the inverse to Janus deletes the string by repeatedly shifting to the left and chopping off the leftmost digit, and this is read from left to right. Uncertainty is now increased due to the ignorance of what was erased. The information content of the string, after k iterates of the inverse have been applied, is k (i.e., $\log 2^k$) since 2^k possible strings have been eliminated and are now unknown.

The Janus algorithm has now been given a new twist it as a device that reduces randomness through the act of measuring the future, while its inverse increases uncertainty because it discards acquired digits. Put another way, the entropy decreases with each measurement since it lessens any surprise the future may hold, but, in its wake, there is a trail of useless digits that now serve only as a record of what once was. These junk digits increase entropy since they represent disorder and randomness. Janus hits its stride when it is combined with its inverse. Abolishing the past after recording the future brings one back to the beginning; entropy is preserved, and the Second Law of Thermodynamics is not violated. The demon has been exorcized by reconciling the two faces of ignorance now and disorder later. Figure 3.7 shows the sequence of steps schematically starting, for simplicity with an initial value of zero.

In the next chapter Janus returns to establish a similar reconciliation between the dual views of randomness as information and complexity.

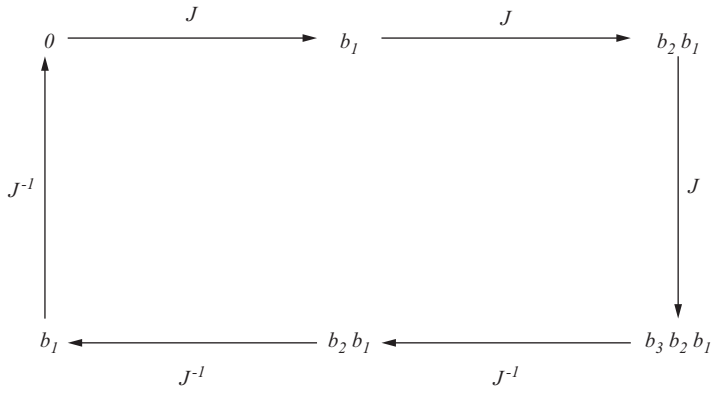

Fig. 3.7 The action of the Janus algorithm on the initial value zero (chosen for simplicity of representation), shifting one place to the right. This shift is denoted by *J*, for Janus, and it accrues a randomly generated digit b_1. Repeated action of *J* adds digits b_2 and b_3 to the left. The deterministic inverse, denoted by J^{-1} for Janus inverse, shifts one place to the left and chops off the leftmost digit, so that $b_3b_2b_1$ becomes b_2b_1, and then b_2b_1 becomes b_1. *J* decreases entropy, while J^{-1} increases it. Since *J* and J^{-1} cancel each other in pairs, the total action is to return to the starting value zero, and entropy is conserved

Quantum Indeterminacy

The uncertainty that we have seen in Boltzmann's model is due to coarse graining that makes the positions of individual molecules look fuzzy. However when we speak of what happens in the subatomic world, it appears that chance is built into its very fabric. Few, if any, claim to understand the bizarre doings of particles subject to the rules of quantum mechanics. Many physicists simply go along with the unfathomable behavior simply because it works. Quantum theory is a successful tool for predicting what happens, but any understanding of why things are the way they are is elusive. Perhaps there is some hidden mechanism

underlying quantum phenomena, but so far this is pure speculation. What will be done here, instead, is to give a brief and somewhat heuristic explanation of how probability enters into the quantification of quantum chance.

In order to motivate the discussion is may be helpful to begin with an analogy. It's the familiar idea that any vector v in the plane can be represented as the sum of its projection onto orthogonal, namely, perpendicular, x and y coordinate directions. Think of the vector, which for simplicity we assume has unit length, as an arrow with one end at the origin of the plane and pointing in some arbitrary direction as in Fig. 3.8 in which we see the orthogonal projections of the arrow onto the two separate coordinate directions. The vector (or arrow) v is a superposition of v_1 and v_2, namely, $v = \alpha_1 v_1 + \alpha_2 v_2$ where v_1 and v_2 are arrows of unit length in the x and y directions, respectively, and α_1 and α_2 are the lengths of the separate projections. Note that since v has unit length, the familiar Pythagorean theorem of elementary geometry tells us that length squared of v is the sum of

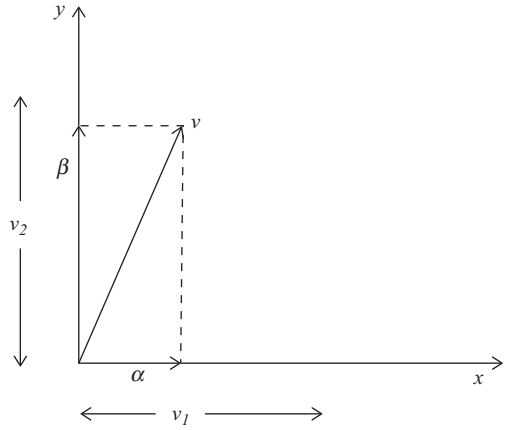

Fig. 3.8 The orthogonal components αv_1 and βv_2 of the vector v

the squared lengths of the projections, namely, $\alpha_1^2 + \alpha_2^2$. Something similar prevails in the quantum world as will be seen.

Let's choose the simplest situation in which there are only two possible measurement outcomes, also referred to as *states*, labeled *spin up* and *spin down*. In quantum theory the idea of a particle's spin is as different from the ordinary concept of spin as one can imagine and so we need to think of it abstractly. It is what one observes when an apparatus is oriented in a given direction to measure spin, and it obtains one and only one of these two outcomes; up and down are mutually exclusive states. Quantum systems with only two possible outcomes are often called *qubits* (quantum bits), and the state S of an arbitrary qubit is a superposition of spin up and spin down each weighted by a complex number α_1 and α_2: $S = \alpha_1$ spin up + α_2 spin down. You may recall that a complex number z is of the form

$$z = a + i\,b$$

where are a and b are real numbers and i is the imaginary quantity which satisfies $i^2 = -1$. Each z has a squared magnitude R, a real and positive number defined by the product of z with its complex conjugate $z^* = x - ib$, namely, $R = zz^*$.

The state S is indeterminate; Can we assign a probability to its various outcomes? Yes, but to do this, we need to remember that probabilities must be real and positive numbers, which is not generally the case with complex numbers. The trick then is to take the squared magnitudes of α_1 and α_2, call them R_1 and R_2, in which case the probability of observing a particle in state S to be spin up is R_1 and the probability of observing it spin down is R_2. Moreover, since there are only two possible outcomes, these probabilities must sum to one. If state S is normalized to have length

one, we have an *analogy* with the Pythagorean theorem that was invoked earlier which tells us that the square of the length of *S*, which evidently equals one, is the sum of the squares of its two orthogonal projections onto states spin up and spin down. That is, $S = \alpha_1$spin up + α_2 spin down with $\alpha_1\alpha_1^* + \alpha_2\alpha_2^* = 1$. To say that these physically distinct states are orthogonal is not meant in the usual geometrical sense but rather that each of them has no component in terms of the other.

Let me say a little more about this. The spin up and spin down states are mutually exclusive in the sense that if we orient our measuring apparatus along some specific direction *D* in three dimensional space and spin up is recorded, say, then repeating this observation under identical conditions will again record spin up with certainty. Rotating the apparatus in a diametrically opposite direction then unambiguously records spin down. In either case we have fixed the state to be up with probability one (and zero probability for down) or conversely. In the parlance of quantum theory, these two states are said to be *orthogonal*. However, if the apparatus is now oriented along a different direction, then what it records is either up or down at random. What this means is by repeating the procedure of first fixing an up or down state along direction *D* and then rotating the apparatus to the new alignment the outcomes up and down will now be haphazardly recorded independent of what was observed previously. It's like tossing a coin successively to obtain a random string of heads and tails. In general the probabilities of one outcome or the other are not the same (the coin is biased) except when the new alignment is *spatially orthogonal* to *D* in which case they are each equal to $\frac{1}{2}$. What I have done here is to paraphrase what was stated previously: the state of the particle's spin in the new orientation is a superposition of the orthogonal states spin up and

spin down, and it assumes one of these states at random with probability $\alpha_1\alpha_1^*$ or $\alpha_2\alpha_2^*$.

An unusual notion of chance presides in the subatomic world to be sure, different in flavor that one is accustomed to, but there you have it.

4

Algorithms, Information, and Chance

The Library is composed of an ...infinite number of hexagonal galleries...(it) includes all verbal structures, all variations permitted by the twenty five orthographical symbols, but not a single example of absolute nonsense. It is useless to observe that the best volume of the many hexagons under my administration is entitled The Combed Thunderclap *and another* The Plaster Cramp *and another* Axaxacas mlö. *These phrases, at first glance incoherent, can no doubt be justified in a cryptographical or allegorical manner; such a justification is verbal and, ex hypothesi, already figures in the Library. I cannot combine some characters dhcmrlchtdj which the divine library has not forseen and which in one of its secret tongues do not contain a terrible meaning*

The certitude that some shelf in some hexagon held precious books and that these precious books were inaccessible seemed almost intolerable. A blasphemous sect suggested that...all men should juggle letters and symbols until they constructed, by an improbable gift of chance, these canonical books...but the Library is...useless, incorruptible, secret

from The Library of Babel by Jorge Luis Borges

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2_4

Algorithmic Randomness

During the decade of the 1960s, several individuals independently arrived at a notion that a binary string is random if its shortest description is the string itself. Among the main protagonists in this story is the celebrated Russian mathematician Andrei Kolmogorov, whom we met earlier, and Gregory Chaitin, information theorist and computer scientist.

The *algorithmic complexity* of a binary string s is formally defined as the length of the *shortest program*, itself written as binary string $s^\star$, that reproduces s when executed on some computer (see Fig. 4.1). A "program" in this context is simply an algorithm, a step-by-step procedure, which has been coded into binary form.

The idea behind the definition of algorithmic complexity is that some strings can be reproduced from more concise descriptions. For example, if s is a thousand-fold repetition of 01 as 010101…, this can be condensed into the statement "copy 01 a thousand times," and as will be seen, this can be coded by a short binary program $s^\star$ that squeezes out the redundancy. This program requires a fixed number of bits to accommodate the instructions "copy 01 a thousand times," and the number 1000 additionally demands another 10 bits (since 1000 can be written in binary form as 1111101000). Altogether, the program $s^\star$ is much smaller than s, and when the computer acts on $s^\star$, it produces s as its output. The string s therefore has low algorithmic complexity.

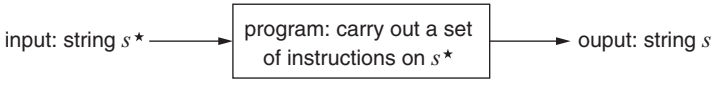

Fig. 4.1 Representation of a computer acting on an input string s^* to produce an output string s

On the other hand, a sequence of zeros and ones generated by Bernoulli $\frac{1}{2}$-trials gives a binary string s that in all likelihood cannot be described by any method shorter than simply writing out all the digits. In this case, the requisite program "copy s" has roughly the same length as s itself, because what one needs to do is to supply the computer with s, and it then displays the result. The string has maximum algorithmic complexity.

A string s is said to be algorithmically random if its algorithmic complexity is maximal, comparable to the length of the string, meaning that it cannot be compressed by employing a more concise description other than writing out s in its entirety. The code that replicates the string is the string itself and, *a fortiori*, the string must replicate the code.

If a long string s of length n has a fraction p of ones (and, therefore, a fraction $1 - p$ of zeros), then recalling the arguments in Chapter 2, there is a binary string $s^\star$ of length roughly nH that replicates s by exploiting the redundancies in the string. The string $s^\star$ can be constructed as a Shannon code: if p is not equal to $\frac{1}{2}$, then the entropy H of the source is less than one and $s^\star$ is shorter than s. In order to place this coding within the framework of algorithmic complexity, all that is needed is a program that performs the decoding of $s^\star$. As you may recall, this decoding can be done uniquely in a step-by-step manner, since Shannon's procedure is a prefix code, meaning that none of the code words are prefixes of each other. The string s has moderate algorithmic complexity. Figure 4.2 illustrates strings of low, moderate, and maximum complexity.

With $p = \frac{9}{10}$, for instance, the entropy H is about .47, and the algorithmic complexity of s, being the size of the shortest program, does not exceed .47 n. From this it is evident that for s to be considered algorithmically random, the proportions of 0 and 1 must be very nearly equal. One can

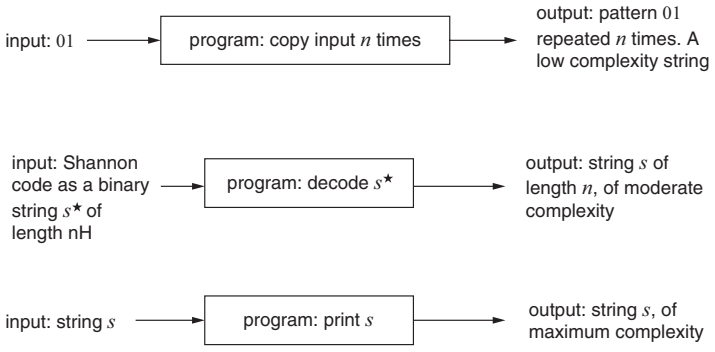

Fig. 4.2 Representations of computer algorithms for strings of low, moderate, and maximal algorithmic complexity

say more. Every possible block of 0 and 1 of length k in a random sequence must also appear with roughly equal frequency for each $k = 1, 2, \ldots$; otherwise patterns can be detected that can be used to shorten the description, in a similar manner to a Shannon coding. For infinite strings, you then reach a familiar conclusion: random sequences must define *normal numbers*. Algorithmic randomness is therefore consonant with randomness in the usual sense of maximum entropy and of normal numbers.

Incidentally, though the sine qua non of randomness of a string is that it be the binary representation of a *normal* number, not every normal number is random. In Chapter 1 you encountered the only well-documented example of such a number, due to D. Champernowne in 1933, and it is characterized by a simple description of its binary expansion: choose each digit singly, then in pairs, then in triplets, and so on. At the nth step list, in order, all possible 2^n strings of length n, giving rise to the sequence 0100011011000001…. Any sufficiently long but finite segment of this number cannot be random since the algorithm for generating it is fairly short.

Remarkably, if every book written were to be coded in binary form, each of them would sooner or later surface within a long enough segment of Champernowne's number, since every possible string eventually appears and reappears indefinitely. Your favorite novel or recipe or even the book you are currently reading will pop up somewhere along the way. Champernowne's number contains the vast storehouse of written knowledge in a capsule.

Randomness in the information-theoretic sense of maximum entropy was framed in Chapter 2 in terms of the probabilities of the source symbols. By contrast, the complexity definition of randomness makes no recourse to probability, and it depends only on the availability of a mechanical procedure for computing a binary string. Moreover, the complexity definition is independent of the provenance of a string s and at looks only at the string itself, merely as a succession of digits. In this way it overcomes the embarrassment of having to contemplate a not so very random looking output of some allegedly random mechanism; remember that Bernoulli $\frac{1}{2}$-trials can crank out a string of all zeros or a totally un-patterned string with the same insouciance. Another difference between the two approaches is that Shannon ignores the semantic aspects of a generated string, whereas Kolmogorov/Chaitin is concerned with the information encoded in a string. As an extreme example, suppose the source generates strings from just two symbols, for a meager information content of one bit. However each message could be a long book, and in spite of redundancies in the English language, the algorithmic complexity of each book would be substantial with many bits needed to encode the book's content.

In the previous chapter, it was mentioned that a random string is random when viewed in reverse. This can be seen from the complexity definition as well: if *s* taken backward

can be reproduced by a shorter string $s^\star$, then the relatively short program "read $s^\star$ in reverse" will reconstruct s, and this contradicts the putative randomness of s (Fig. 4.3).

A similar argument shows that the minimal program $s^\star$ must itself be random. Otherwise there is a shorter string $s^{\star\star}$ that reproduces $s^\star$ and a new program can be created that concatenates $s^\star$ and $s^{\star\star}$ with the command: "use $s^{\star\star}$ to generate $s^\star$ and follow the instructions of $s^\star$." After the machine executes $s^{\star\star}$, a few more bits of program are needed to instruct the computer to position itself at the beginning of $s^\star$ and continue until s appears. The new program that spawns s is shorter than s and so $s^\star$ is not minimal. This contradiction establishes that $s^\star$ must indeed have maximum complexity (Fig. 4.3).

Randomness of s has been defined by the algorithmic complexity of s matching the length of s. It is comforting to know that there is at least one string of length n. We can show this by counting: there are 2^k strings of length k for any k less than n, and the sum of these strings for $k = 1, 2, \ldots, n-1$ is $2^n - 1$ (Appendix A explains how this sum is arrived at). Therefore at least one string must have complexity no less than n. A slight modification of this argument establishes something deeper. Let's stipulate that a string of

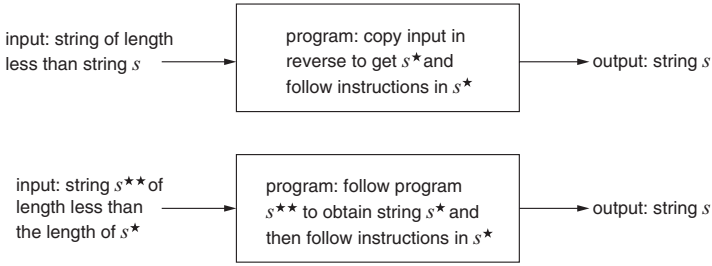

Fig. 4.3 Representations of computer programs for establishing that the converse of a random string is again random, and to show that a minimal program $s^\star$ is itself random

length n is *d-random* if its algorithmic complexity is greater than or equal to $n - d$ for some small integer d. In effect complexity, no less than $n - d$ is good enough for it to be virtually indistinguishable from random, an idea that makes sense for long enough strings. There are at most 2^k distinct programs of length k, for $k = 1, 2\ldots, n - d - 1$, and therefore at most

$$2 + 2^2 + \ldots + 2^{n-d-1} = 2^{n-d} - 2$$

or fewer than 2^{n-d}, programs that describe strings of length n having complexity less than $n - d$. Since there are 2^n strings of length n, the fraction of strings that are not d-random is less than $2^{n-d} / 2^n = 1/2^d$. With $d = 10$, for instance, fewer than one in a thousand strings are not d-random, since $1/2^{10} = 1/1024$. From now on, the randomness of long strings is identified with d-randomness in which d is negligible relative to the length of the string. With this proviso, *most strings are random because their algorithmic complexity is close to their length.*

There is a link between algorithmic complexity and the *Janus algorithm* of the previous chapter. Assume, for simplicity, that the Janus iteration begins with the seed value u_0 equal to zero, namely, a string of zeros, and let us watch as it successively computes n digits of Bernoulli $1/2$ -trials from right to left. Initially the unknown string is one of 2^n equally-probable strings of length n, and its information content is n bits. After k iterates the information content of the remaining portion of the string is $n - k$, since only $n - k$ bits remain to be specified. However, if the string is incompressible, the first k bits represent a block of complexity roughly k, and as k increases, for $k = 1, 2, \ldots, n$, the uncertainty diminishes while the complexity grows. Now apply the inverse to Janus to erase the digits created, one

step at a time, from left to right. After k erasures the information content rises to k, since k bits have been lost, but the complexity of the remaining portion of the string has dwindled to roughly $n - k$. The sum of the two terms, entropy and complexity, remains constant at each step.

There is a duality here between the increase in entropy in one direction and an increase in complexity in the opposite direction; *information content and complexity are seen as complementary ways of representing randomness* in an individual string.

If Janus is modified to produce digits from Bernoulli p-trials, then some redundancies can possibly be squeezed out, and so after k steps, the remaining uncertainty is only $(n - k)H$, where H is the entropy of the source. The complexity of the remaining portion is roughly kH using a Shannon encoding, as you saw above. Once again, the sum of the two terms is nearly constant at each iteration.

The self-reflection of Janus mirrors the hallucinatory nature of randomness in which ignorance begets disorder which begets ignorance. The Argentine writer Jorge Luis Borges captured this narcissism in an essay in which he taunts us with "why does it disturb us that the map is included in the map and a thousand and one nights in the book of the *Thousand and One Nights* ? Why does it disturb us that Don Quixote be a reader of the *Quixote* and Hamlet a spectator of *Hamlet* ? I believe I have found the reason: these inversions suggest that if the characters of a fictional work can be readers or spectators, we, its readers or spectators, can be fictitious." A more light-hearted version of these musings is captured by *The New Yorker* cartoon in which an announcer steps through the parted curtain and notifies the theater-goers that "tonight the part normally played by the audience will be played by the actors playing the part of the audience."

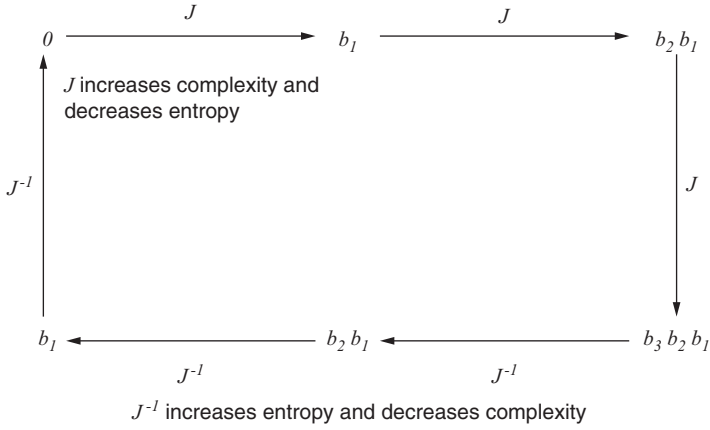

Fig. 4.4 Schematic representation of how entropy and complexity as two sides of the same Janus-faced coin. This figure is a variant of Fig. 3.7 and uses the same notation adopted there

Figure 4.4, a variant of Fig. 3.7 in the previous chapter, gives a schematic if somewhat prosaic representation of Janus's self-reflection.

With the above definition of complexity definition in hand, randomness of a binary string can now be understood in three senses:

- A string is random if each digit is generated by some mechanism in an unpredictable manner. The randomness resides in the disorder of the generating process. The only way the string can be reproduced is for a monkey working at a binary switch to accidently hit upon its pattern.
- A string is random because it is completely unanticipated; its entropy is maximal. Recognition of the string elicits total surprise.
- A string is random because no prescribed program of shorter length can regurgitate its successive digits. Since

it is maximally complex in an algorithmic sense, the string cannot be reproduced except by quoting itself, a futile circularity.

Algorithmic Complexity and Undecidability

The notion of algorithmic complexity was defined in terms of a program executed on a computer. To make further progress with this idea and its connection to randomness, it is necessary to clarify what is meant by a computer.

In 1936, well before the advent of electronic computers in the 1940s, the mathematician Alan Turing reduced the idea of computation to its bare bones by envisioning a simple device that carries out in rote fashion a finite number of commands, one at a time, following the precise and unambiguous steps of an algorithm or, in computer jargon, a *program*. Turing's conceptual device incorporated the essential ingredients of a modern computer and was a precursor to the actual computing machines that were to be built a decade later.

His imaginary machine consists of a scanner that moves along a tape partitioned into cells lined up in a single row along a strip. Each cell contains either a one or a zero, and a portion of the tape consists of the binary expansion of some number which represents the input data, one digit per cell. Depending on what the scanner reads on the first cell it is positioned at, it carries out one of a small set of moves. It can replace what it reads with either a 0 or 1, it can move one cell right or left, or it can go to one of the finite number of steps in the algorithm and carry out whatever instruction it finds there, and this might include the command "stop." The finite set of instructions is the machine's program.

A simple example of a Turing computation is the following three steps:

Step 1	Move one cell to the right.
Step 2	If the cell contains a 0, go to step 1.
Step 3	If the cell contains a 1, go to step 4.
Step 4	Stop.

Here is a key observation: the instructions in the program may themselves be coded in binary form. There are seven code words:

Code	Instruction
000	Write 0
001	Write 1
010	Go left
011	Go right
1010…01	Go to step *i* if 0 is scanned (there are *i* zeros to the right of the leftmost 101)
1011…10	Go to step *i* if 1 is scanned (there are *i* ones to the right of the leftmost 101)
100	Stop

An example of the fifth instruction is 10100001, which says "go to step 4 if 0 is scanned." The simple program of four steps given earlier can therefore be coded as the string 01110101101111110100 since 011 is step 1, 10101 is step 2, 10111110 is step 3, and 100 is the final step "stop." In effect, if a 1 is encountered after step 1, then the program moves to step 4 and then stops.

The translation of a program into binary form is evidently a prefix code and is therefore immediately decipherable by reading left to right. Suppose that a Turing machine, labeled *T*, carries out some computation on a given input data string and stops when some output string *s* appears on the tape. Turing conceived another machine, labeled *U*,

that systematically decodes the binary instructions in the program for T and then mimics the action of T on its input data. This mechanism, the *universal Turing machine*, operates on a tape whose initial configuration is the coded program for T followed by the input data. All that U has to do is unwaveringly decode each command for T and then scuttle over to the input string and carry out the operation that T would have itself performed. It then returns to the next command in T's program and repeats this scurrying back and forth between program and data. T's program is stored on the tape as if it were input data for U and it passively cedes its role to U.

The computers in commercial use conform to the same general scheme: you write a program in some language like Fortran, and the computer then interprets these commands using its own internal language that forges an output using the binary logic inscribed on its chips. Figure 4.5 is a schematic representation of a universal Turing machine.

One of the benefits of a universal Turing machine U is that it removes an ambiguity that lurked in the background of the definition of algorithmic complexity of a string s. The length of the shortest program $s^\star$ that replicates s on a given computer depends on the idiosyncrasies of that machine's hardware. To remove any dependency on the particular machine in use, it is convenient to define the complexity in

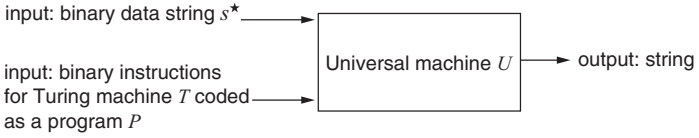

Fig. 4.5 Schematic of a universal Turing machine U that acts on the binary instructions of a Turing machine T coded as a binary program p. Turing machine U simulates the behavior of T on an input data string $s^\star$ to produce an output string s. This may be expressed symbolically as $U(p, s^\star) = T(s^\star)$

terms of a universal computer that simulates any other device. This, of course, introduces some additional overhead in terms of the instructions needed to code *U*'s internal interpreter. This overhead increases the program by a fixed number of bits but is otherwise independent of the string *s* that is the desired output. Any two machines that compute *s* have programs whose lengths can differ at most by a constant number of bits since each of them can be simulated by the same *U*, and so the complexity of *s* is effectively machine independent. The additional "overhead" becomes less significant as *s* gets longer, and it is useful to think of lengthy strings when talking about complexity so that machine dependency can be neglected.

Since Turing's original work, it has become an entrenched dogma, the *Church-Turing Thesis* (so-called because Turing's work shadowed that of the American logician Alonzo Church that was carried out independently and at about the same time), stating that anything at all computable using some algorithm, either mechanically or by a human, can also be performed by a universal Turing machine. This stripped-down, almost toy-like, caricature of a computer has the remarkable property of being able to realize, in principle at least, any program whatever. We are talking here of the ability to carry out the steps of any program but an actual computation on a Turing machine, if it could be realized in practice, would be woefully slow.

In what follows, a *Turing computation* or, equivalently, a *Turing program* is an algorithm, coded in binary form, that is executed on a Turing machine.

Not every Turing computation halts when presented by some input string. The four-step program given above halts after a finite number of moves if the input contains some cell with a one on it; for an input consisting solely of zeros, it grinds away without ever stopping. A glance at this

program reveals immediately whether it stops or not for a given input, but in general it is not obvious when or whether a given program will halt. Turing established the startling result that there is no program P crafty enough to determine whether any given program T will eventually halt when started with some input data string. The inability to decide whether a Turing computation will halt is closely linked to Godel's notorious incompleteness theorem, and it is mirrored by an equally striking assertion regarding algorithmic complexity due to Gregory Chaitin. I relegate Turing's argument to the *Technical Notes* and move instead directly to what Chaitin showed.

Let us regard a proof of an assertion as a purely mechanical procedure using precise rules of inference starting with a few unassailable axioms. This means that an algorithm can be devised for testing the validity of an alleged proof simply by checking the successive steps of the argument; the rules of inference constitute an algorithm for generating all the statements that can be deduced in a finite number of steps from the axioms. There is, in effect, a Turing program that can work through the chain of inferences to test the validity of an alleged proof. If the axioms and proofs are coded as binary strings, the program begins by checking binary proof strings of length 1, then all such strings of length 2, then those of length 3, and so on, and whenever a valid proof is encountered, namely, a proof that follows from the axioms using the given rules of inference, it prints it out. Eventually any valid proof will be found in this manner by the Turing program.

Consider now the claim that there is a proof of the statement "for any positive integer n there is a binary string whose complexity exceeds n." The Turing program is an algorithm that tests all possible proofs in order of increasing length until it finds the first one able to establish that some

string has complexity greater than n. It then writes out this string and stops.

The program has a length of k bits, where k is the fixed length of the binary string that encodes the particular choice of axioms as well as the instructions required to check the details of the proof, plus about log$n$ bits that are needed to code the integer n in binary form (see Appendix C to verify the last assertion). Though log$n$ increases as n gets larger, it does so more slowly than n, and, therefore, a large enough n will eventually exceed the length of the program, namely, $k + \log n$. Here you have a contradiction: the program supposedly *computes the first string that can be proven to have complexity greater than n, but this string cannot possibly be calculated by a program whose length is less than n*. The contradiction can only be avoided if the program never halts!

Chaitin's proof is related to a paradox posed by Oxford librarian G. Berry early in the twentieth century that asks for "the smallest positive integer that cannot be defined by an English sentence with fewer than 1000 characters." Evidently the shortest definition of this number must have at least 1000 characters. However, the sentence within quotation marks, which is itself a definition of the alleged number, is less than 1000 characters in length! This paradox appears to be similar to that of "the first string that can be proven to be of complexity greater than n." Although the Berry paradox is a verbal conundrum that does not seem to be resolvable, in Chaitin's case the paradox dissolves by simply observing that no such proof exists. That is, it is not possible to ascertain whether a given binary string has complexity greater than some sufficiently large n for if such a proof existed, then, as we saw, there would be an algorithm that uses less than n bits to carry out the proof, and that contradicts the meaning of complexity.

A paraphrase of Chaitin's result is that there can be *no formal proof that a sufficiently long string is random* because there is no assurance that its complexity can ever be established, since the complexity of a random string is comparable to its length. No matter how one devises the axioms and an algorithm for testing valid proofs, there is always some long enough string that eludes a proof of randomness, even though it may in fact be random.

In response to the persistent question *"Is it random?"* the answer must now be "probably, but I'm not sure"; "probably" because most numbers are in fact random, as we have seen, and "not sure" because the randomness of long strings is essentially undecidable. A monkey is likely to generate a long string that is in fact random, but it is unlikely that we would be able to recognize it as such.

Algorithmic Probability

We are given a binary string s whose complexity l is determined by the shortest binary computer program, namely, a string s^* of length l, that outputs s. For each s let's assign a probability, the *algorithmic probability of s*, to be 2^{-l}. Among all 2^l computer programs of length l, this is the likelihood that some specific program is chosen to generate s. Since probabilities of disjoint events must sum to something not exceeding one, we need to impose a technical condition that the shortest length programs for each s constitute a prefix-free set. This is explained more fully in the *Technical Notes*, and for our purposes here, this requirement can be safely ignored.

Because of the way algorithmic probability is defined, it follows that low complexity strings are more likely than those of high complexity, and it ensures that if one is given

two strings s_1 and s_2, then the one of *shortest length is the more probable of the two*.

It is part of the lore of science that the most parsimonious explanation of observed facts is to be preferred over convoluted and long-winded theories. Ptolemaic epicycles gave way to the Copernican system largely on this premise, and, in general, scientific inquiry is governed by the oft-quoted dictum of the medieval cleric William of Occam that "numquam ponenda est pluralitas sine necessitate" which may be paraphrased as "choose the simplest explanation for the observed facts." This fits in with the conclusion reached in the previous paragraph that if two hypotheses are framed as binary strings, the most likely hypothesis is the shorter one.

The mathematician Laplace has already been quoted in the first chapter regarding the need to distinguish between random strings and "those in which we observe a rule that is easy to grasp," a prescient remark that anticipates algorithmic complexity. Laplace precedes this statement by asserting that "if we seek a cause whenever we perceive symmetry, it is not that we regard a symmetrical event as less probable than the others but, since this event ought to be the effect of a regular cause or that of chance, the first of these suppositions is more probable than the second. On a table we see letters arranged in this order, *constantinople*, and we judge that that this arrangement is not the result of chance, not because it is less possible than the others, for if this word were not employed in any language we would not suspect that it came from any particular cause, but with the word in use among us, it is incomparably more probable that some person has arranged the aforesaid letters than that this arrangement is due to chance." Later he continues "being able to deceive or to have been deceived, these two causes are as much more probable as the reality of the event is less."

A string s of length n generated by a *chance* mechanism has a probability $1/2^n$ of occurring but a *cause* for s would be a much shorter string s^* of length m that outputs s on a universal computer U starting on a blank tape. The probability of getting $s^\star$ is $1/2^m$ out of all equally-likely programs of length m where m is, of course, the complexity of s. The ratio of 2^{-m} to 2^{-n} is 2^{n-m}, and whenever m is much less than n, it is then 2^{n-m} times *more probable that s arose from a cause than from chance.* Short patterns are the most convincing, perhaps because their brevity gives them the feel of truth. That may be why we are so attracted to the pithy assertion, the succinct poem, and the tightly knit argument. The physicist Werner Heisenberg wrote "if nature leads us to mathematical forms of great simplicity and beauty...we cannot but help thinking that they are true."

5

The Edge of Randomness

And so one may say that the same source of fortuitous perturbations, of "noise", which in a nonliving (i.e., non-replicative) system would lead little by little to the disintegration of all structure, is the progenitor of evolution in the biosphere and accounts for its unrestricted liberty of creation, thanks to the replicative structure of DNA: that registry of chance, that tone deaf conservatory where the noise is preserved along with the music

from Chance and Necessity, by Jacques Monod

Valentine: It's all very, very noisy out there. Very hard to spot the tune. Like a piano in the next room, it's playing your song, but unfortunately it's out of whack, some of the strings are missing, and the pianist is tone deaf and drunk—I mean, the noise! Impossible!

Hannah: What do you do?

Valentine: You start guessing what the tune might be. You try to pick it out of the noise. You try this, you try that, you start to get something- it's half baked, but you start putting in notes that are missing or not quite the right notes... and bit by bit...the lost algorithm !

from Arcadia, by Tom Stoppard

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2_5

Between Order and Disorder

Up to now I have been trying to capture the elusive notion of chance by looking at binary strings, the most ingenuous image of a succession of sensory events. Since most binary strings cannot be compressed, one would conclude that randomness is pervasive. However, the data streams of our consciousness do, in fact, exhibit some level of coherence. The brain processes sensory images and unravels the masses of data it receives, somehow anchoring our impressions by allowing patterns to emerge from the noise. If what we observe is not entirely random, it doesn't mean, however, that it is deterministic. It is only that the correlations that appear in space and time lead to recognizable patterns that allow, as the poet Robert Frost puts it, "a temporary stay against confusion." In the world that we observe, there is evidently a tension between order and disorder, between surprise and inevitability.

I wish to amplify these thoughts by returning to binary strings, thinking of them now as encoding the fluctuations of some natural process. In this context the notion of algorithmic complexity that preoccupied us in the previous chapter fails to capture the idea of complexity in nature or in everyday affairs, as it is usually perceived. Consider whether the random output of a monkey at a keyboard is more complex than a Shakespearean sonnet of the same length. The tight organizational structure of the sonnet tells us that its algorithmic complexity is less than the narration produced by the simian. Evidently, we need something less naive as a measure of complexity than simply the length of a program which reproduces the sonnet. Charles Bennett has proposed instead to measure the complexity of a string as the *time required, namely, the number of computational steps needed*, to replicate its sequence from some other

program string. Recall the algorithmic probability that was defined in Chapter 4 as a sum over all programs that generate binary strings *s*. This probability is evidently weighted in favor of short programs since they contribute more to the sum than longer ones. If most of the probability is contributed by short programs for which the number of steps required to compute *s* is large, then the string *s* is said to have *logical depth*.

The lengthy creative process that led to Shakespeare's sonnet endows it with a logical depth that exceeds by far the mindless key tapping of the monkey even though the sonnet has less algorithmic complexity because of redundancies, some obvious and others more subtle, that crop up in the use of language. Although the information content of the sonnet may be limited, it unpacks a one of a kind message that is deep and unexpected.

Allowing that a long random string *s* can be replicated only by a program of roughly the same length as *s*, the instructions "copy *s*" can nevertheless be carried out efficiently in a few steps. Similarly, a totally ordered string such as 01010101… requires only a few computational steps to execute "copy 01 *n* times" since a simple copy instruction is repeated many times over and again. Clearly, strings like these that entail very large or very small algorithmic complexity are shallow in the sense of having meager logical depth: little ingenuity is needed to execute the steps of the program. Therefore, strings that possess logical depth must reside somewhere between these extremes, *between order and disorder*. As Charles Bennett put it: "the value of a message thus appears to reside not in its information (its absolutely unpredictable parts), not in its obvious redundancy (verbatim repetitions, unequal digit frequencies), but rather in what might be called its buried redundancy--parts predictable only with difficulty."

Another striking example of logical depth is provided by DNA sequences. This familiar double-stranded helix is made up of nucleotides consisting of sugars and phosphates hooked to one of four different bases designated simply as A, C, G, T. Each triplet from this alphabet of symbols codes for one of the 20 amino acids that are linked together to form proteins. Some of these proteins are enzymes that, in a self-referential way, regulate the manner in which DNA unfolds to replicate itself, and other enzymes regulate how DNA transcribes its message to make more proteins.

Writing each of the four symbols in binary form as 00, 01, 10, 11 exhibits DNA as a long binary string (about 3 billion symbols in humans). Since there are $4^3 = 64$ possible triplets of nucleotides A, C, G, T, called codons, and only 20 amino acids, there is some duplication in the way transcription takes place. Moreover, some fragments of the DNA strand repeat many times, and there also appears to be long-term correlations between different portions of the string, and this suggests that there is considerable redundancy. On the other hand, some codon sequences appear to be junk, having no recognizable role, possibly caused by chance accretions over the span of evolutionary time. It follows that DNA lies betwixt randomness and structure, and its logical depth must be substantial, since the evolution of the code took place over several million years. A DNA string s can therefore be replicated by a shorter program string s^*, but the blueprint of the more succinct code s^* is likely to be complicated, requiring a lot of work to unpack it into a full description of the string s.

The genetic machinery of a cell provides not only for the copying of DNA and, indirectly, for its own replication, but it controls the onset of an organism's growth through the proteins coded for by DNA. The cells combine to form organisms and the organisms then interact to form

ecosystems. The mechanical process of DNA replication is disrupted from time to time by random mutations, and these mutations function as the raw material for natural selection; evolution feeds on the fortuitous occurrence of mutations that favor or inhibit the development of certain individuals.

Chance also intrudes beyond the level of genes as organisms have unexpected and sometimes disruptive encounters with the world around them, affecting the reproduction and survival of individuals and species. Some of these contingent events, what physicist Murray Gell-Mann calls "frozen accidents," have long-term consequences because they lock in certain regularities that persist to provide succeeding organisms, individually and collectively, with characteristics that possess some recognizable common ancestry. The accumulation of such frozen accidents gives rise to the complexity of forms that we observe around us. The helical structure of DNA and of organic forms like snails may be a consequence of such accidents. Jacques Monod expressed the same idea most evocatively as "randomness caught on the wing, preserved, reproduced by the machinery of invariance and thus converted into order, rule, necessity."

Some accidents of nature permit existing parts to be adapted to new functions. Paleontologist Stephen Jay Gould comments on how happenstance provides an opportunity for selection when he argues that the complexity of forms is due to "poor fit, quirky design, and above all else, redundancy...pervasive redundancy makes evolution possible. If animals were ideally honed, with each part doing one thing perfectly, then evolution would not occur, for nothing could change and life would end quickly as environments altered and organisms did not respond." Fully formed parts are unlikely to be made from scratch but are a product of nature's fortuity. Catastrophic events, such as large

meteor impacts, can also alter the course of evolution by extinguishing some species and bestowing a selective advantage to other, previously marginal, dwellers.

Redundancy in DNA and in the organisms it spawns makes them less fragile to disruptions. The same is true at the level of ecosystems, which apparently can achieve species diversity because of spatial heterogeneity and fluctuations in the environment. Disturbance of an ecological community allows new species to colonize areas that would normally be inhabited by a more aggressive competitor, and sporadic incursions such as floods, storms, or fires can leave in their wake a patchy landscape co-inhabited by more species than would otherwise be possible in a more stable environment in which predation and competition would ensure the dominance of just a few species. The ecologist G. Evelyn Hutchinson once pondered "the paradox of the plankton" whereby a number of competing species of plankton are able to co-exist, rather than one species surviving at the expense of the others, and he similarly concluded that this was possible because turbulence in the waters dislodges the community structure from equilibrium.

Whether it be at the level of cells, organisms, or ecosystems, chance and order co-mingle to unfold the vast panorama of the living world. This underscores the utility of randomness in maintaining variability and innovation while preserving coherence and structure.

A number of thinkers have gone further in suggesting how some semblance of order and coherence can arise from irregular and accidental interactions within biological systems. The idea is that large ensembles of molecules and cells tend to organize themselves into more complex structures, hovering within the extremes of, on one hand, totally random encounters where turmoil reigns and no order is possible and, on the other, tightly knit and rigidly regulated

interactions where change is precluded. Cells and organisms in isolation, shut off from the possibility of innovation, veer toward decay and death. Biologist Stuart Kauffman believes that "selection... is not the sole source of order in biology, and organisms are not just tinkered-together contraptions, but expressions of deeper natural laws...profound order is being discovered in large, complex, and apparently random systems. I believe that this emergent order underlies not only the origins of life itself, but much of the order seen in organisms today."

For physicist Per Bak, emergent order occurs not only in the realm of biological phenomena, but it is rampant in the worlds of physical and social experience: "complex behavior in nature reflects the tendency to evolve into a poised critical state, way out of balance, where minor disturbances may lead to events, called avalanches, of all sizes...the evolution to this very delicate state occurs without design from any outside agent. The state is established solely because of the dynamical interactions among individual elements of the system: the critical state is *self-organized*."

Kauffman underscores these sentiments when he says that "speciation and extinction seem very likely to reflect the spontaneous dynamics of a community of species. The very struggle to survive, to adapt to small and large changes... may ultimately drive some species to extinction while creating novel niches for others. Life, then, unrolls in an unending procession of change, with small and large bursts of speciations, small and large bursts of extinctions, ringing out the old, ringing in the new...these patterns ...are somehow self organized, somehow collective emergent phenomena, somehow natural expressions of the laws of complexity."

Chemist Ilya Prigogine had, even earlier, given vent to similar ideas when he advocated a view of nature far from

equilibrium in which evolution and increasing complexity are associated with the self-organization of systems that feed on the flux of matter and energy coming from the outside environment "corresponding to a delicate interplay between chance and necessity, between fluctuations and deterministic laws." He contrasts these pools of decreasing entropy with the irrevocable degradation and death implied by the Second Law of Thermodynamics. Complex structures offset the inevitable downhill slide into disintegration by dumping their own decay into the open system that nourishes them. Recall the Szilard demon of Chapter 3 who is able to decrease entropy and increase order, bit by bit, by accumulating a record of junk digits that ordinarily need to be erased to restore the loss in entropy. But if these digits are unceremoniously wasted into the environment, a pocket of order is created locally even as disorder is increased globally.

The thesis of self-organized complexity is a controversial idea that finds its most ardent voice today at the Santa Fe Institute in New Mexico, and it is not my intention to engage in the ongoing polemic regarding the validity of this and competing ideas of complexity that are vexing these thinkers. This is best reviewed in a spate of books that have appeared in the last few years, not the least of which is Bak's effort in *How Nature Works*. Instead, I accept this idea as a provocative metaphor of how chance and order conspire to provide a view of complexity in nature, and in the artifacts of man.

A closely related paradigm of how nature organizes itself between order and disorder is that symmetry, a manifestation of order, tends to be countered in biological systems by asymmetric organic molecules including mirror asymmetry as discovered by Louis Pasteur and others in the nineteenth century. Much later Erwin Schrodinger was prescient in intuiting that the structure of genetic molecules would have

to be a-periodic and not periodic crystals. Asymmetry as a form of disorder initiated by chance events is ubiquitous in biological systems.

That randomness gives rise to innovation, and diversity in nature is echoed by the notion that chance is also the source of invention in the arts and everyday affairs in which naturally occurring processes are balanced between tight organization, where redundancy is paramount, and volatility, in which little order is possible. One can argue that there is a difference in kind between the unconscious, and sometimes conscious, choices made by a writer or artist in creating a string of words or musical notes and the accidental succession of events taking place in the natural world. However, it is the perception of ambiguity in a string that matters, and not the process that generated it, whether it be man-made or from nature at large.

In Chapter 2, we saw that the English language is neither completely ordered, which would render it predictable and boring, nor so unstructured that it becomes incomprehensible. It is the fruitful interplay between these extremes that gives any language its richness of nuance. The same is true of the music of Bach, to mention just one composer, which is poised between surprise and inevitability, between order and randomness. Many architects attempt to combine wit with seriousness of design to create edifices that are playful and engaging while meeting the functional requirements dictated by the intended use of these buildings.

In a book about mystery and romance John Cawelti states "if we seek order and security, the result is likely to be boredom and sameness. But, rejecting order for the sake of change and novelty brings danger and uncertainty...the history of culture can be interpreted as a dynamic tension between these two basic impulses...between the quest for order and the flight from ennui."

A final example of how chance intrudes to provide an opportunity for novelty and complexity and the formation of patterns far from equilibrium is based on the cliché, popularized by John Guare in his play "Six Degrees of Separation," that everyone is connected to everyone else in the world through at most six intermediate acquaintances. A study by mathematicians Duncan Watts and Steven Strogatz shows that an ensemble of entities tightly woven into parochial clusters can rapidly expand into a global network as soon as a few links are randomly reconnected throughout the network. Examples abound to show that structures as diverse as the global community of humans, electric power grids, and neural networks have all evolved to reside somewhere between a crystalline structure of local connectedness and random disarray.

All these examples are reminiscent of a delightful watercolor by the eighteenth-century artist Pietro Fabris, one of a series used to illustrate a work called Campi Phlegraei by the urbane scholar and diplomat Sir William Hamilton, British envoy to the then Kingdom of Naples. In the water color Hamilton is leaning on his staff below the crater of Vesuvius, viewing the sulfurous pumice being hurled haphazardly from the belching vent reflecting, it seems, on the delicate balance between the tumult and anarchy of the untamed volcano and the unruffled azure sky beyond, between the capriciousness of ordinary life and the world of reason, an exquisite portrayal of life poised between order and disorder.

Complexity and Power Laws

As we saw in the very first chapter, the central organizing principle in conventional probability theory is the Normal Law which tells us, roughly, that sums of independent

random samples with a common average tend to be distributed as a bell-shaped normal curve. For example, if the samples are the heights of people, then the average represents a characteristic size for the population, and most individuals don't deviate too far from that typical scale length. On the other hand, there are numerous ensembles of variables that have no typical scale since they range over a wide span and there is no characteristic value for their distribution. One thinks of the size of cities which ranges from small towns of several thousand people to that of large metropolis whose population is measured in millions (a number of other examples will be given later). In a sense to be made precise momentarily, being scale-free is an attribute of what is called a *Power Law* in which there many small to medium-sized events interspersed by a smaller number of extreme happenings. The study of such scale-free distributions provides a statistical anchor for a large array of data that lacks a characteristic size and, as such, serves as an organizing principle that is similar, in some respects, to that of the normal curve in the traditional theory. However, while the conventional theory is set on a sound mathematical footing, it must be admitted that some of what is reported about Power Laws is speculative even though there is certainly a core of mathematical validation and at least some of it is partly supported by empirical evidence.

A number of mechanisms have been proposed for the existence of power laws, one of which I'll review later, but essentially what seems to be at work here is that a large number of variables representing contingent events interact over a wide range of temporal and spatial scales. This manifests itself in a distribution of the frequency of sizes that possess what are generally called fat tails. This means that the occurrence of extreme events, though they are not frequent, is much more common that would be expected from the normal curve. As we will see, many natural processes

follow such a distribution, as do man-made patterns like the financial markets. Catastrophic occurrences such as large earthquakes and financial meltdowns can be anticipated with more regularity than conventional probability models would lead one to believe.

A relation that assigns a value to some positive quantity x is said to be a power law if it is proportional, for all sufficiently large x, to the reciprocal of some power b of x, namely, $1/x^b$ with b a number that is usually between one and three.

An important attribute of power law relations is that they are *scale-invariant* meaning that if one stretches or contracts the variable x by some factor s, so that we now measure it on a different scale, then the shape of the relationship remains essentially unaltered (this is illustrated for the power law $1/x$ in Fig. 5.1).

If the relation is between the magnitude of an event and the rate of occurrence x of that event then, for example, there is just as much energy dissipated in many small earthquakes as in a few large seismic events, and the same can be said for the energy in wind turbulence where there are a few large gusts interspersed among many smaller puffs. The sizes of moon craters are another instance because there are small and large meteor impacts, and there are as many small species extinctions in the geologic record as large ones. I'm not making this up (see, for instance, M. Browne, *Many Small Events May Add Up To One Mass Extinction*, NY Times, Sept 2, 1997). These instances of self-similarity and many others in nature have been verified empirically as in the plot of plankton data exhibited in Fig. 5.2 in which there are many high frequency oscillations of plankton mass of moderate size and a few large magnitude oscillations of lower frequency. This figure does not look like a power law in its present form; what is missing is the conversion of the plankton data into a squared magnitude of the oscillation,

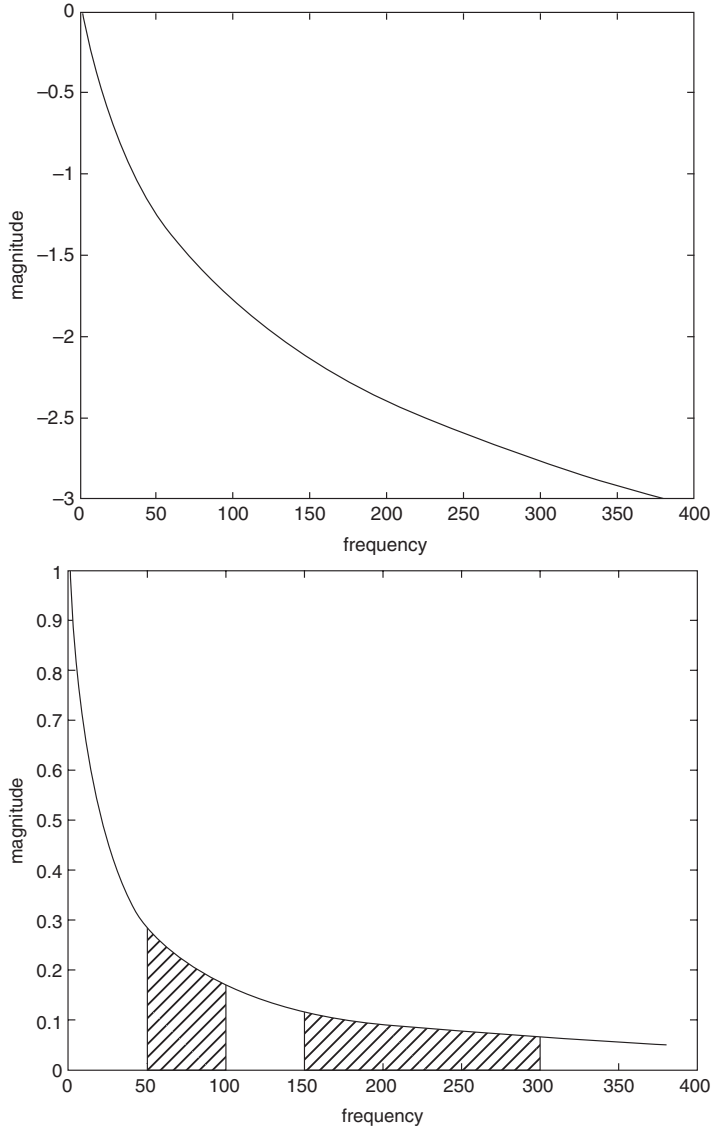

Fig. 5.1 Power Law 1/*x*. The shaded portions have equal areas indicating that many small events are equivalent to a few larger events

often referred to as the spectrum, corresponding to each frequency of oscillation, a step that is omitted here since it involves technicalities that we need not get into. After this conversion, a plot of magnitude against frequency turns out to be a power law resembling Fig. 5.1. Going forward I'll use the term magnitude instead of spectrum since this will not affect our discussion.

To paraphrase what I said earlier, the most plausible explanation for Power Law behavior is that the events we measure are due to a large ensemble of mechanisms, some large and others small, and it is the confluence of these many factors that results in a lack of prejudice with respect to scale. Individual occurrences are often multiplicative and contingent on each other and the data spreads out considerably.

Scale invariance is often associated with the idea of *fractals* in which the statistical properties of a distribution appear the same at different levels of magnification. "Clouds are not spheres, mountains are not cones, coastlines are not circles, and bark is not smooth, nor does lightning travel in a straight line" asserts Benoit Mandelbrot in his influential and idiosyncratic book on fractal geometry. A fractal structure is not smooth or homogeneous. When one looks at a fractal more and more closely, greater levels of detail are revealed, and, at all the different scales of magnification, the structure exhibits more or less similar features. An example often mentioned is the coastline of Britain whose perimeter gets longer the closer it is examined. On a large scale, it has a perimeter that misses many tiny inlets and coves, but these make their appearance when measured with a smaller measuring stick, and all the additional indentations uncovered at this scale serve to increase the length. The crinkliness of the shoreline continues until one gets down to the dimension of individual pebbles. The plankton data of Fig. 5.2 is fractal, and it begets a power law. Another

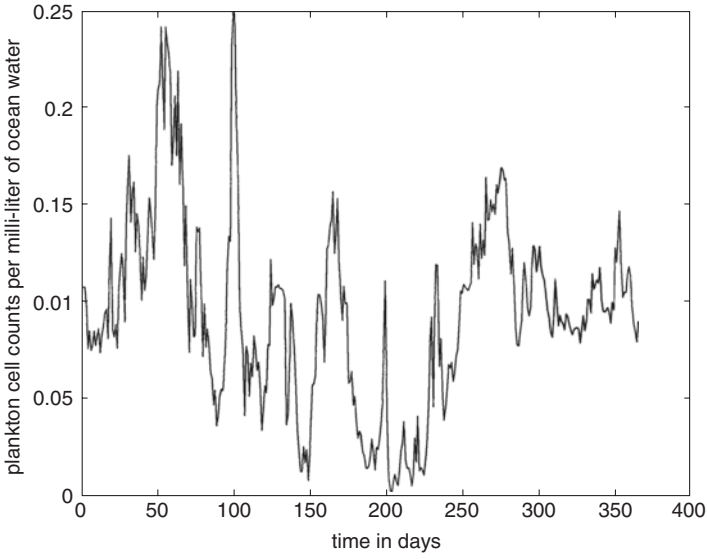

Fig. 5.2 Plankton cell counts per milliliter of ocean water as a daily average, over a nearly 1-year span. The data were obtained at a mooring off the continental shelf of the east coast of the United States

illustration is familiar from Chapter 1 where we considered the fluctuations of accumulated winnings of Peter and Paul in a game of heads and tails. If a plot of 5000 tosses is blown up by a factor of 10, so that only 500 tosses now come into view, a similar pattern of fluctuations appears at this new scale, and the cluster of points now show additional clusters within them that were not visible before. Clusters are actually clusters within clusters, down to the level of a single toss (Fig. 5.3). Looking at the gaps between returns to zero gains, we again find a power law relating the magnitude (length) of the gaps to their frequency: many small gaps interspersed by a few longer ones. Two other examples are the branching of bronchial tubes in the lung and the branching of a river basin into smaller rivulets.

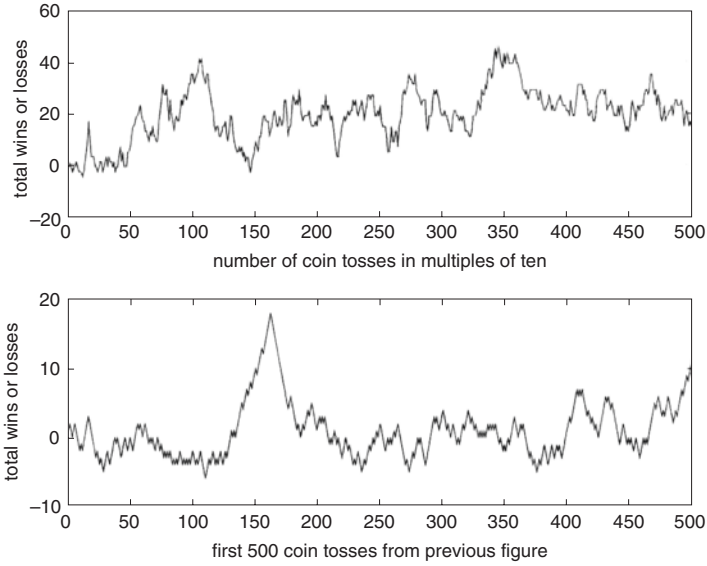

Fig. 5.3 Fluctuation in total winning (or losses) in a game of heads and tails with 4000 tosses, plotted every tenth value (upper half), and a close-up of the same figure from 1 to 500 (lower half). The same amount of detail is visible in the plot magnified 10 times as there is in the original

The self-similarity of fractal structures implies that there is some redundancy because of the repetition of details at all scales. Even though some of these structures may appear to teeter on the edge of randomness, they actually represent complex systems at the interface of order and disorder. The reason for discussing fractals and power laws is that, as you will soon see, there is an identifiable signature of self-similarity that allows us to make a connection to the notions of self-organization of complex systems discussed in the previous section.

The best way to understand power law distributions is to give more examples, which I do now. Begin with the fraction of people having income x in some large population.

Vilfredo Pareto established in 1897 that the income distribution is, at least for the more affluent segment of the population, a power law with exponent b hovering in most cases about the value two, *more or less*. What applies to income applies to wealth more generally or to the wages paid. If x_0 is the minimum annual income, say about $\$10,000$, then the percentage of people having an annual income at least ten times that, namely, $\$100,000$, is $1/10^{b-1}$. If we identify the fraction of people as the probability of an individual having a given income x, then what I am saying is that the probability of an income greater or equal to x is $(x_0/x)^{b-1}$. Thus, if $b = 2$, say, then only .1 of the population has an income of at least $\$100,000$. When x equals ten times that, a million dollars, then one percent has at least that income, and for ten times that only a fraction .001, one-tenth of a percent, have incomes exceeding that amount. Thus the fraction of people enjoying a certain level of income above the minimum is inversely proportional to the income level. This can also be interpreted as the conditional probability of having an income greater than x given that a person's income is greater than x_0.

An appealing conceptualization of these computations is called the *80–20 rule* as first observed by Pareto who stated, as variant to the income-wealth label, that 20% of the people (in Italy) held 80% of all the land. The same observation applies to a wide array of instances, indeed whenever there is a power law. Needless to say, *80–20* is just a convenient mnemonic for what in different settings might actually be 70–30 or 90–10, but, in all cases, the top-heavy power function tells a story about some level of imbalance in society, a few more outrageous than others. Examples abound, though some may be anecdotal: 20% of the people generate 80% of the insurance claims, 20% of the workers in a firm do 80% of the work (an example that has been vouched for is that in which 4 people in a realty firm of 20 generate 85%

of all sales), 20% of patients utilize 80% of the health-care resources, 20% of a firm's customers account for 80% of sales volume, 20% of all firms employ 80% of the workforce, and so forth. An oft-quoted power law, due to George Zipf from 1949, is the distribution of the frequency of words in some text, such as Moby Dick. He prepared a ranking of words from the most common, starting with the word *the*, followed by *of*, down to those least used, and then plotted the frequency with which each word appears in the text and he found that this is very nearly inversely proportional to its rank. To say that the *r*th most frequently used word has rank r is equivalent to asserting that *r* words have rank greater than or equal to *r* and so the Zipf plot is actually a power law of the Pareto type in which the fraction of words with rank no less than *r* is plotted against *r*.

Based on the 2000 census, we can plot the population of US cities having more than a thousand people. Drawing a smooth curve through the data gives a curve describing the fraction of cities whose population exceed *x*. This distribution is far from the normal. Though one can compute an average size of US cities, the individual sizes do not cluster about this mean value; there are many more small to medium cities than larger ones, and there is no tendency to cluster about this central value. We can apply the 80–20 rule to say that 20% of the cities harbor 80% of the US population. There are many other examples. In all cases the word fraction is interchangeable with percentage or frequency or, better, probability. To mention a few of more prominent:

The fraction of websites with *x* or more visitors (Google has many hits while most sites languish in obscurity), the fraction of earthquakes with a Richter scale magnitude of *x* (many small tremors and a few large upheavals), the sale of books (in any time period, many books have a paucity of

sales but just a few bestsellers sell millions of copies), the fraction of all US firms of size x or more, the fraction of all meteor craters exceeding a diameter x, and the fraction of animals of a certain size versus metabolic rate (George Johnson, *Of Mice and Elephants*, NY Times, Jan 12, 1999). It has even been suggested that shots of scenes in Hollywood films satisfy a power law with many short sequences interspersed by longer ones, perhaps as an attempt to relieve tedium and increase attention span.

We could go on, but you get the idea. In all cases there are infrequent events having a large impact coupled with many events having little or no consequence. We can say, in short, that it's normal not to be normal!

One rationale for power laws is known as *self-organized criticality* and is often explained in terms of the sand pile model. Here grains of sand are dropped onto a flat surface until the sloping sides of the pile reach a certain critical incline beyond which any new grain begins to slip a little or a lot. At some point one falling grain results in an avalanche. It is the confluence of myriads of small contingent events that give an avalanche of any size, and it cannot be attributed to any single movement within the pile. Thereafter, when the tremor subsides, the pile starts to build up again until it reaches a critical state once more, and it is in this sense that it is self-organized. The distribution of sizes is shown to be a power law. The disturbance caused by additional grains are contingent events, and it is argued (Bak, *How Nature Works*, 1996) that the pile is a paradigm of many natural processes that organize themselves to a poised critical state at which minor disturbances can trigger a large avalanche. Earthquakes, forest fires, and species extinctions are all cited as examples of this. In each instance the footprint of a scaling law indicates a complex process organized to the brink of randomness simply as a result of the

interactions among individual elements of the system. Many systems in nature that follow a power law hover between order and disorder and every movement is the confluence of many effects, some small and others large, with no characteristic spatial or temporal scale, leading to changes for no discernible cause. They just happen. Significant external prodding can shake up a system and get it moving but what occurs next is unanticipated. Another manifestation of randomness.

Using $1/x^b$ as a statistical anchor to organize a mass of seemingly unrelated phenomena finds an echo in the preceding two centuries when an attempt was first made to tame chance by appealing to the "law of errors," the Gaussian law. There are some striking differences, however. The Gaussian bell-shaped curve describes the distribution of superpositions of unrelated events, such as the sample average, in which independent e outcomes are added, and these totals spread themselves about the population average, with most near the average and less of them further out. By contrast, power law phenomena hinges on contingency.

If one employs a *logarithmic scale* on both the horizontal and vertical axis, the $1/x^b$ curve will appear as a *straight line* sloping downward from left to right (this is shown in the *Technical Notes*). On this scale the decrease in volatility from large to small magnitude events, as the frequency increases, becomes an instantly recognizable signature of how the magnitude of fluctuations is caused by the confluence of many contingent events that distribute themselves by a rule of self-similarity. Indeed the appearance of such a straight line on a log-log scale is often the first clue that one is looking at a Power Law. To consider just one example, Fig. 5.4 exhibits the log-log plot of the plankton data of Fig. 5.2 (after it has been suitably converted as explained

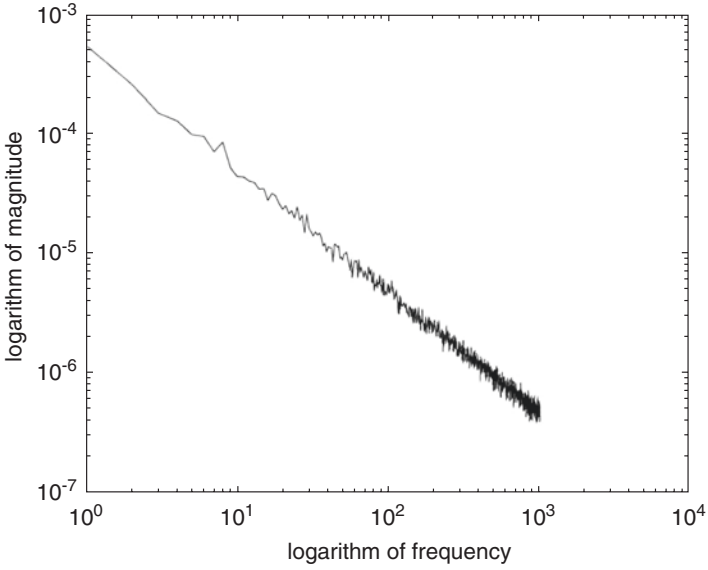

Fig. 5.4 The magnitude of a signal (Fig. 5.4) versus frequency, plotted on a logarithmic scale

earlier). It is not quite a straight line because of inherent sampling errors, but it nonetheless reveals its power law origins.

The data that conforms to $1/x$ laws are decidedly non-Gaussian in several other respects. Because of self-similarity, there are many more small fluctuations than very large ones as we have seen, and so the average fluctuation will depend on the ratio of large to small components. The sample average will tend to increase without bound or shrink to zero as more and more pieces of data are included, and the Law of Large Numbers, which held data in its grip in the preceding chapters, is no longer operable. Moreover, the spread about the population average in the Gaussian case is now poorly defined since dispersion increases as time is extended due to the fact that ever larger fluctuations get included.

Much as the Gaussian law establishes a form of order in its domain, the $1/x$ laws provide a signature organizing principle for a different category of events. The appearance of a $1/x$ footprint indicates a complex process organized at the brink of randomness. Physicist Per Bak's book, mentioned earlier, is a paean to the ubiquity of power laws in nature. In an unfettered moment, he exclaims "self-organized criticality is a law of nature from which there is no dispensation." This can be compared to the equally unbridled comment by Francis Galton, a century earlier regarding the Normal Law that we quoted in the first chapter in which he ends by writing: "The larger the mob, and the greater the apparent anarchy, the more perfect is its sway. It is the supreme law of unreason." Wow!

Near-random processes endowed, as they are, with shifting patterns of order, may conceivably be the consequence of relatively short codes that provide rules of replication and redundancy as a consequence of statistical self-similarity over a wide range of temporal and spatial scales. The rules at one scale beget similar rules at other scales that lead to the emergence of ever more complex structures. This is parodied by Jonathan Swift's verse in reverse: "So, naturalists observe, a flea hath smaller fleas that on him prey, and these have smaller fleas to bit'em, and so proceed ad infinitum."

What Good Is Randomness?

Our romp through the many nuances of randomness in this book comes to a close in the present chapter by proposing that you think of uncertainty as something more than an elusive nuisance or an oddity of mathematics.

In antiquity randomness was seen as a cause of disarray and misfortune, disrupting what little order individuals and

society had managed to carve out from their surroundings. The world today still contends with uncertainty and risk in the form of natural disasters, acts of terrorism, market downturns, and other outrageous slings of fortune, but there is also a more uplifting view of randomness as the catalyst for life-enhancing changes. Uncertainty is a welcome source of innovation and diversity, providing the raw material for the evolution and renewal of life. Because stasis leads to decay and death, chance fluctuations may guarantee the viability of organisms and the resilience of ecosystems. And, just possibly, chance offers a temporary refuge from the inexorable Second Law of Thermodynamics by permitting complex structures to emerge through the exploitation of fortuitous accidents. Finally, chance events relieve the tedium of routine and furnish the element of surprise that makes languages, the arts, and human affairs in general, a source of endless fascination.

6

Fooled by Chance

In the play *Rosencrantz and Guildenstern* by Tom Stoppard, a streak of 76 consecutive heads occur in a coin tossing game. The loser remarks that *"A weaker man might be moved to re-examine his faith, if in nothing else than in the laws of probability"*. And, later, *"the equanimity of your average tosser of coins depends on a tendency which ensures he will not upset himself by losing too much nor upset his opponent by winning too often."*

Binary Strings, Again

In Chapter 1, we raised the question of how people perceive randomness. The tendency is for individuals to reject patterns such as a long run of heads in a binary sequence as not typical of randomness and to compensate for this by judging frequent alternations between zeros and ones to be more typical of chance; experiments that ask subjects to either produce or evaluate a succession of digits reveal a bias in favor of more alternations than an acceptably random string can be

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2_6

expected to have; people tend to regard a clumping of digits as a signature pattern of order when in fact the string is randomly generated. Closely related to this cognitive flaw is the Gambler's Fallacy of supposing that after a long run of failures, there is a higher probability of success than is warranted by Bernoulli trials. We quoted the comment by psychologists Maya Bar-Hillel and Willem Wagenaar that an individual's assessment of randomness in tosses of a fair coin seems to be based on the "equi-probability of the two outcomes together with some irregularity in the order of their appearance."

There is an oft-told illustration of how people misconstrue *success runs* in coin tossing. On the first day of a probability course, the instructor asks his students to record 200 flips of a fair coin. Of course some may cheat and simply make up a random looking sequence of tosses. The next day the instructor amazes the class them by glancing at each of the student's papers and reporting, in nearly all cases, which are true coin tosses from faked data. His ability to do so is based on the surprising fact that in a sequence of 200 tosses, it is extremely likely that a run of six or more consecutive heads or tails will occur, as I show below. However, the typical person will rarely include runs of that length.

Nevertheless a more careful look at these sequential runs indicates that some vestige of validity can be attributed to these fallacies as I'll discuss momentarily. But first I need to establish a formula for the probability of success runs of a given size. I'll give the main ingredients of the argument here, due to Berresford [13]. Define a run, clump, or streak of size k in a string of n coin tosses as a sequence of *exactly k successive heads*. We want to compute the probability $P(n, k)$ of finding runs of size no less than k. To do this we consider two mutually exclusive events whose separate probabilities are then added:

(i) There are runs of size no less than k among the first $n - 1$ tosses.

(ii) There are no runs of size no less than k in the first $n-1$ tosses, but the last k tosses out of n do form a clump.

Note that (ii) is the intersection of two independent events, the first being that the last $k+1$ tosses are of the form tail followed by k heads (if these $k+1$ tosses were all heads, there would be a run of size no less than k among the first $n-1$ tosses, which we exclude), and the second event is that there is no run within the first $n-k-1$ tosses. Because of independence, the probabilities of these events are multiplied. The complete proof Berresford's result is left to the *Technical Notes*.

Now let's compute $P(200, 6)$ to give an answer to the question posed at the beginning of this section, the one in which the instructor asked his students to toss a coin 200 times. The likelihood that there is a clump of heads of size equal or greater than 6 is .8009, a better than 80% chance, and there is a better than even chance, .5437, of a head run of length no less than 7. These unexpected results are what confounded the students.

To put the formulas in to their simplest context, I list here the sixteen strings associated with $n = 4$ and look at runs of the digit one of various lengths:

1 1 1 1	1 0 1 1	0 1 1 1	0 0 1 1
1 1 1 0	1 0 0 1	0 1 1 0	0 0 1 0
1 1 0 1	1 0 1 0	0 1 0 1	0 0 0 1
1 1 0 0	1 0 0 0	0 1 0 0	0 0 0 0

From the formula for $P(n, k)$, we find that $P(4, 2) = .5$ and $P(4, 3) = .1875$ and these values are immediately verifiable from the above table since there are eight runs of size no less than 2, giving a probability of $\frac{8}{16} = .5$, and three runs of size no less than 3, leading to a probability $\frac{3}{16} = .1875$.

I stated that a random binary sequence is expected to have runs of a length longer than what would be anticipated in an invented sequence. There is a caveat however. An equally valid calculation establishes that on average about a quarter of all tosses are alternating head-tail or tail-head flips (Bloom [14]). But a sequence with too many singles tends to have a smaller number of gaps available for long runs, so where does the failure of true randomness reside, in a scarcity of long success runs or a paucity of singles? It turns out that these two requirements are in fact compatible and you can have a quarter of tosses as singletons without compromising the occurrence of long streaks of heads and tails.

If I ask for the probability that the immediate successor of a randomly chosen head in a sequence of tosses is also a head we know, from the independence of tosses, that either a head or a tail occurs with equal frequency. Now I'll introduce a conundrum which I take from a striking paper by Miller and Sanjurjo [54] in which we find some fellow Jack tossing a balanced coin in which there is an equal chance of getting a head or tail on each flip. He generates a relatively short sequence of such tosses and then looks at those flips that follow a head. For example, if we have THTH then T follows H, whereas in HHTH, a head follows a head once and tail once. The flips that immediately follow a head run of length one must be a tail. One expects a considerable number of such singletons as was just noted. For head runs of length two a head succeeds a head just once and a tail once and so on for longer (but also rarer) runs. This leads to an overrepresentation of tails following short runs. Nonetheless, our tosser expects that the proportion of heads that follow a head remains $\frac{1}{2}$. Miller and Sanjurjo write "shockingly, Jack is wrong; the average proportion of heads is less than $\frac{1}{2}$."

To decipher what is going on, let's compute the percentage of heads that follow a head, the *HH percentage* or, equivalently, the *average HH w*hich is the fraction of HHs among all the flips that follow a head in a given sequence of tosses. One ignores sequences with no Hs or with a single H in the last position since in these cases, there is no possibility for a HH . So, for example, in the sequence HTTH, there is no HH (zero percentage), while for HHTT there is a single HH among the two heads (average $\frac{1}{2}$ or 50%). With n flips, there are 2^n possible sequences, all having the same probability of occurrence since the coin is fair. For each of these, one calculates the HH percentage. Dividing by $n-2$ (we exclude the sequence with all tails and the one with just one head in the last position, as already noted), we obtain the average number of heads that follow a head, an estimate of the probability of HH. The entire procedure is tantamount to flipping a fair coin n times, computing the average HH, and then repeating this procedure a large number of times; each of the 2^n sequences will appear roughly an equal number of times if n is large enough. By forming the sample mean of the individual averages, one obtains the approximate probability of HH, just as before. Sadly, this probability will be less then $\frac{1}{2}$. Something is terribly wrong since *we know* that in independent tosses of a fair coin, an H or a T follows an H with equal probability.

The puzzle can be illustrated with a simple example by considering the case of three tosses leading to eight possible sequences. In the table below, we see these 8 sequences followed, in the second column, by the number of flips that follow a head. The third column records the HH frequency proportion or average HH among the flips noted in the second column. The expected value, or proportion, of the individual frequencies is $2+\frac{1}{2}$ divided by 6 eligible sequences equals $\frac{5}{12}=.416$ and this is less than .5.

A homely example illustrates what is going on. Suppose that in a certain community, 10, 000 homeowners own a single residence but that another 300 homeowners have a second residence as well. 40% of the single homeowners have annual family incomes of $100, 000 or more, whereas *all* the multiple homeowners have such incomes. So, for the community at large the un-weighted average of average family incomes of the two groups is the sum of 40% and 100% divided by 2, namely 70%. This gives the absurd result that 7, 210 families have incomes of $ 100,000 or more when we know that only 4000 plus 300 equals 4300 have such incomes. The correct number is found by giving equal weight to each household which entails taking a weighted average of 0.4 × 10, 000 plus another 1.0 × 300 to obtain 4, 300 families with incomes exceeding $100,000. That's much better.

So, returning to coin tosses giving equal weight to sequences of flips rather than to the flips themselves is problematic because taking *an un-weighted average of averages is an unsound statistical procedure.* Think of the individual home owners as coin flips and the groups into which they were divided as two sequences. Giving equal weight to each group of homeowners corresponds to giving equal weight to each of the six sequences of flips. So if one focuses on the sequences and not on the individual flips, we obtain a biased average. The reason is that, as with the homeowners, some sequences have more heads than others. Note, however, that although our estimate for the probability of HH is biased, it does not negate the irrefutable fact that H will follow a head just as often as T does. In the table above, there are an equal number of HH and HT, six of each. In fact, if one had picked a flip at random instead of a sequence then, since the distribution of heads is uneven, the sequences would not have been chosen with equal

likelihood, and the HH frequency would have been a weighted average; in this case the probability of HH is indeed $\frac{1}{2}$.

3-flip sequence	# of recorded flips	proportion of Hs in column 2
TTT	0	
TTH	0	
THT	1	0
HTT	1	0
THH	1	1
HTH	1	0
HHT	2	$\frac{1}{2}$
HHH	2	1
Expected proportion:		$\frac{5}{12}$

The contrived HH computation leads to an interesting interpretation of the *Gambler's Fallacy*, the belief that a succession of heads in coin tossing will soon be reversed. The long-run frequency of heads/tails will tend to equalize as discussed earlier, but this long-term equalization is incorrectly viewed to hold even in relatively short runs. In the Miller and Sanjurjo paper, the authors suggest that what seems to be going on is that gamblers observe many short sequences of complete games applying an equal weight to each of these sequences or games obtaining what is in effect an un-weighted average of the percentages of a head following a head. This under-estimates the true probability that an H follows an H or, equivalently, an over-estimate that a tail will follow a run of heads, thus providing support for the intuitive feeling that after a succession of heads, there is an increased probability the next flip will be a tail. Quoting Miller and Sanjurjo again, *this confirms a belief in a reversal of fortune that is consistent with the gambler's fallacy*.

Poisson's Model of Randomness

To give more insight into the nature of randomness, let's introduce something called the *Poisson distribution* named after the early nineteenth-century French mathematician Simeon Denis Poisson who launched the essential concept in a book published in 1837 in which he discusses jury selection in criminal cases together with other topics regarding legal matters.

Imagine events taking place unpredictably during a span of time or over an expanse of space, call them "arrivals," with the property that the number of such arrivals over disjoint times periods (or distinct regions of space) are independent of each other and, moreover, does not depend on which temporal or spatial interval one happens to look at. When I say "arrivals," this is generally another way of talking about the occurrence of a "success" or a "hit" of some kind in which an occasional success happens sporadically at random among a bunch of "failures." The examples to follow will bring this terminology to life.

Define $N(T)$ to be the number of so-called arrivals during some time period of length T; the arrivals are sequential in time, and $N(T)$ counts how many actually appeared during T. We assume $N(0) = 0$ and that the arrivals are isolated events; at most one event takes place at any particular instant, and, as already mentioned, the number of arrivals in non-overlapping time intervals is independent in the sense that the number of onsets in one interval says nothing about what happened in a different interval.

Although I do not show it here (there is always the *Technical Notes* as backup), there is an expression for the probability of exactly k successes in a given interval, and it defines what is known as the Poisson distribution, denoted by P_k. Put another way, prob $(N(T) = k) = P_k$, for $k = 0, 1,$

2, …. There is a corresponding expression $N(A)$ which counts haphazard occurrences within a spatial region (we could also consider volumes but area suffices here), and P_k is now the probability of k arrivals in a region of area A. Let me point out that P_k depends on a parameter λ that designates the average number of arrivals, as will become apparent in the examples below.

The Poisson distribution is useful for several reasons, one of which is that it makes precise the notion that things are happening in a totally haphazard manner without any discernible cause. The idea is this. Imagine a square surface (it can be any shape, actually, or simply an interval of time). You scatter a bunch of tiny pellets on it at random, meaning that each throw is independent of the previous toss and with no bias in favor of one part of the surface or another. Put in a slightly different way, each point on the surface has the same probability of being hit as any other. What do you see? One might think the pellets are spread out pretty uniformly on the square, but, in fact, the distribution of points would look very irregular with clusters of them here and there, some small and others large, interspersed by some blank spots, something like Fig. 6.1.

Let's say you dispersed n pellets helter-skelter, where n can be any number. Now divide the square (or any region or time span for that matter) into a bunch M of smaller squares all of the same size, and ask for the probability that any one of these sub-squares has k pellets on it, where k can be 0, or 1, or 2, or any number up to the total n. This is conveyed by the Poisson quantity P_k that was defined above. For reasonably large n, it gives a close approximation to the probability of a sub-square having exactly k pellets sprinkled on it. Let M_k denote the number of these sub-regions containing exactly k pellets so that the sum of the M_k over all k, for $k = 0, 1, 2, \ldots, n$, equals M. By the Law of Large Numbers, P_k is roughly equal to the sample mean M_k/M.

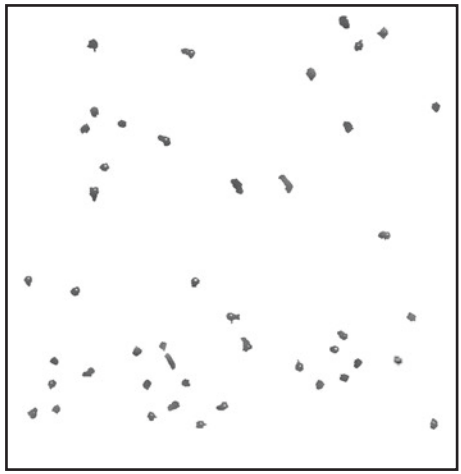

Fig. 6.1 Spatial dispersal of "arrivals" from a Poisson process

Now "pellets" is of course a euphemism for any quantity, such as the location of auto accidents in a city during the month or craters on the surface of the moon. It doesn't have to be space. Arrivals of buses at a particular stop in a busy city during peak rush hour times, and even bizarre events like shark attacks along the Florida coast over the summer and the freakish case of deaths from horse kicks in the Prussian cavalry during a 10-year period, the vacancies of Supreme Court justices from 1790 through 1932, and the number of daily homicides in England and Wales from 2014 to 2016 would be all be verifiable examples. And, if it is spatial, it can be any dimension, not just a surface, such as raisins in a cake. You get the idea.

A striking application of the spatial Poisson approximation is provided by William Feller in his classic book on probability theory in which one finds the data on where $n = 537$ flying bombs (rockets) fell on London during the Second World War. To test whether the hits occurred at random in accordance with the Poisson distribution, the

entire area of south London was divided into $M = 576$ enclaves of $\frac{1}{4}$ square kilometers each, and the table below gives the number of sub-areas M_k that sustained exactly k hits:

k	0	1	2	3	4	5 and over
M_k	229	211	93	35	7	1

Using the Poisson distribution with $M = 576$ trials and 537 hits and letting λ be the number of hits per unit area, namely, $\lambda = n / M = 537/576 = .9323$, and recalling that P_k is approximately M_k/M, it was found that

k	0	1	2	3	4	5 and over
$M \times Pk$	226.7	211.4	98.5	30.6	7.1	1.6

The close agreement between theory and data exhibited here suggests that a hypothesis of random hits cannot be excluded even though a resident in one of the hard hit areas might have felt it a strange coincidence that his neighborhood was singled out while other parts of London went unscathed, whereas a resident of one of these more fortunate areas might have reasoned that her vicinity was spared because that is where the enemy agents were hiding. The unexpected clustering of hits may appear suspicious even though such bursts of activity are characteristic of random processes. What a close-fit to the Poisson distribution shows is that one should be disposed to accept the propensity for bombs to hit one part of the city to be the same as for any other part. The clustering one observes is simply a signature of chance.

The same reasoning shows that it is not inconsistent with randomness to have regions in the United States in which there is an unusually high incidence of some specific cancer. These cancer clusters occur in various parts of the nation

and lead to a belief among some residents of these communities that there must be an unnatural source for the higher than usual rate of malignancies, such as toxic wastes secretly dumped into the water supply by local industries or government agencies who connive in a conspiracy of silence. It seems like just too much of a coincidence. Public health agencies are often cajoled into investigating what the residents of such a targeted township regards with suspicion, disregarding the fact that there are many such locales throughout the nation some of which may not even be aware that some cancer rate is above average. Once again it is a case of ignoring that the probability of such a cluster, somewhere, not just in your backyard, may not be all that small. As the Poisson approximation shows, a cluster is likely to happen somewhere and will affect someone other than you and your community. When it happens to you, it induces skepticism and distrust (see Atul Gawande, *The Cancer Cluster Myth*, *The New Yorker*, February 8, 1999).

One more example may be of interest since it involves temporal rather than spatial occurrences. It originates from a RAND study of Fire Department operations in New York City around 1970. Fire alarms typically vary by time of day and by season. The authors of the study chose five consecutive Friday summer evenings in a region of the Bronx between 8 and 9 pm during which there were $n = 55$ alarms. The five hour period was divided into 15 minute segments for a total of $M = 20$ intervals so that λ can be estimated as $n/M = 2.75$, the number of alarms per unit time. To check whether the data is consistent with a Poisson distribution, the following table gives the number of intervals M_k having exactly k alarms followed by the quantities $M \times P_k$:

k	0–1	2	3	4 or greater
M_k	5	5	4	6
$M \times P_k$	4.80	4.83	4.43	6.94

We see another close-fit suggesting that the fire alarms arrived at random times. This was useful to the investigators since they were attempting to study delays in Fire Department response times, and this depends on having Poisson arrivals of alarms.

It may be easier to intuit the parameter λ by thinking of randomly distributing *n* balls within *M* urns with each urn having the same probability of being chosen. Then λ is the average number of balls per urn.

Some applications combine *chance with skill* as in the study of baseball player Joe DiMaggio's remarkable streak of 56 consecutive games in the 1941 season in which he got at least one hit. Was this feat, unequalled in the annals of the game, due to chance or was it an unusual display of skill? The Poisson distribution was applied to DiMaggio's *anno mirabilis* to establish that it was a combination of both factors as reported in a paper referenced in *Further Readings.*

Cognitive Illusions

The spooky quality of coincidences rarely fails to fascinate and confound people who experience them. What I hope to show is that many, perhaps most, unexpected coincidences are less amazing than they first appear to be. The problem is that there are many possible events and not just those that catch our attention. We tend to focus on those meaningful to us. In effect, when a coincidence appears that we happen to notice, what is being ignored here is the larger number of other events that also lead to striking coincidences but that we failed to detect. The source of wonder in an uncanny coincidence is our selectivity in picking those events that catch our fancy.

For a simple illustration, if you toss two identical balanced die (a cube with sides numbered from one to six), the space of possibilities consists of 36 possible equi-likely outcomes for the number of dots that appear on each die. For some reason or other, you are fixated on the number three maybe because your daughter turned three today. The dies are tossed. We ask, what is the probability of getting the pair (3,3). There is just one possibility here, and so the probability of this happening is $1/36$. However, this is different from asking for the probability of the same number coming up on each die, *any* number from one to six, because there are now six possibilities to consider: (1,1), (2,2), …, (6,6), and so the probability of this event is $6/36 = 1/6$. There are six coincidences here any one of which could be special to someone. Looked at this way you realize that the likelihood of two faces coming up the same is really not so special after all.

Selective reporting is a source of coincidences. To quote Cohen and Stewart (*It's Amazing, Isn't It?*, New Scientist, Jan 17, 1998): "The human brain just can't resist looking for patterns, and seizes on certain events it considers significant, whether or not they really are. And in so doing, it ignores all the neighboring events that would help it judge how likely or unlikely the perceived coincidence really is."

To provide a guidepost to the phenomena of coincidence, I consider first a generalization of the familiar birthday problem in which a group of k people are assembled quite arbitrarily (i.e., at random) and one inquires what the probability is that at least two of these individuals have the same birth date. A modest assumption is made here, one that is not quite true, that all birthdays are equally likely to occur on any day of the year, and that a year consists of 365 days. To make the problem more interesting, we extend this question to ask, in addition, what the probability is that at

least two individuals have a birthday no more than one day apart (near coincidence).

These probabilities can be computed explicitly (I give more details in the *Technical Notes*), and here is what we find: When k is as small as 23, there is a better than even chance that two or more of these 23 individuals will report the same birthdate! On the several occasions in which I've tried this experiment in a classroom of about 30 students, only once did I fail to get an agreement on birth dates from two or more people in the class. The surprise here is that most students believe that it would require a much larger population of individuals to achieve such concordance. It is also true that there is a better than even odds that at least two individuals out of a small group of 14 have the same birthday or a birthday one day apart.

The birthday problem may appear surprising because some people hearing it for the first time are responding to the wrong question, one that sounds superficially like the birthday problem, namely, "what is the probability that someone else has the same birthday as *mine*?". The real issue is whether any two people in a room have the same birthday, and there are many more possibilities for this to occur; we are fooled into thinking of the coincidence as something that happens between us and someone else rather than between any two randomly chosen.

For the sake of demonstrating how different the probability that at least one of n randomly chosen people has the same birthday as *myself*, note that it takes $n = 253$ people before obtaining an even chance of a match, and this number fits in better with the intuition of most individuals who first encounter the birthday problem (details of why this so are in the *Technical Notes*).

Another seemingly paradoxical puzzle is the notorious *Monty Hall problem* that is adapted from the TV program

Let's Make a Deal I draw this account from Jason Rosenhouse's book *The Monty Hall Problem*.

A contestant on a game show is offered a choice of three doors; behind one is a car, behind the others are goats. The show's host knows, of course, what is behind each door. After the contestant chooses a door and before she can open it, the host, Monty, opens one of the other doors to reveal a goat and then asks the contestant if she wants to switch her choice to the remaining door. The unspoken assumption here is that Monty opens a door having a goat, choosing at random between doors if both have goats.

Most individuals who react to the question of whether she should switch will argue that it doesn't matter since there are now only two doors with a goat behind one and a car behind the other and so there is an even chance of finding a car behind either door. However, a careful analysis of the choices, using the notion of conditional probability introduced in the first chapter, reveals the totally unexpected and counterintuitive result that by switching the contestant increases her chances of winning from $\frac{1}{3}$ to $\frac{2}{3}$, as can be seen from Fig. 6.2 in which there are, to begin with, three equally likely choices for doors to pick. The contestant chooses one of the doors at random with probability $\frac{1}{3}$. Behind two of these doors are goats, and her options are either to accept Monty's offer and switch or to stick with the initial decision. It is apparent that by switching, she gets a car $\frac{2}{3}$ of the time, whereas by sticking she wins only $\frac{1}{3}$ of the time. Amazing! So, what happens is that having chosen one of three doors at random, the prior probability of getting a car is only $\frac{1}{3}$, but the new evidence provided by the host opening a door and showing a goat supplies conditional information that now alters the probabilities. It's a subtle problem that has been widely studied since it is

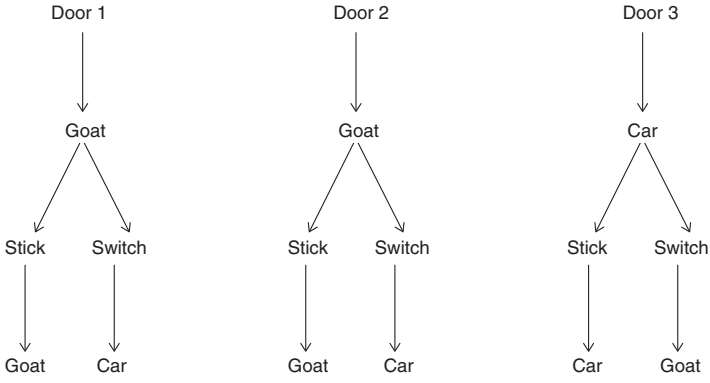

Fig. 6.2 The Monty Hall door options illustrating the advantage of a contestant switching doors

so bizarre. Chance has deceived us into having a moment of cognitive dissonance!

In Chapter 1, there is a brief mention of conditional probability. Recall that for two events A and B in the sample space, the probability of A given that B has taken place is indicated by Prob (A given B) which is shortened to Prob ($A|B$). I want to use this concept to give a mathematically more satisfying treatment of the Monty Hall problem. To do this, it is convenient to first introduce the notion of odds as an alternative and sometimes more intuitive way of framing questions about probability. The *odds* in favor of an event A is the ratio of Prob (A) to Prob ($\sim A$), where $\sim A$ indicates the negation of A, the set of all possibilities in which A never occurs. Therefore Odds (A) = Prob (A)/Prob ($\sim A$).

A few examples of odds: if the sample space contains 8 equi-likely binary strings of length 3 and A denotes the event that 010 or 101 transpires then though the probability of A is $2/8 = 1/4$, the odds of A is $1/4$ divided by $6/8 = 3/4$, namely, $1/3$ indicating one favorable outcome for three unfavorable. (Note: these odds are read as "one to three" and

are sometimes written "1 : 3".) It is easy to compute that odds of $\frac{1}{1}$ corresponds to a probability of $\frac{1}{2}$, odds of $\frac{2}{1}$ matches up with a probability of $\frac{2}{3}$, and, for one more example, an odds of $\frac{3}{7}$ tallies with a probability of $\frac{3}{10}$.

In the *Technical Notes* for the first chapter, there is a proof of Bayes' Theorem. I reproduce it here:

$$\text{Prob}\left(A|B\right) = \text{Prob}\left(B|A\right) \times \text{Prob}\left(A\right) / \text{Prob}\left(B\right)$$

There is a similar expression for Prob ($\sim A \mid B$). Dividing the left side of Bayes' formula for A by the same expression with $\sim A$ in the place of A and then similarly dividing the right side by the expression in which A is replaced by $\sim A$, furnishes us with a corresponding Bayes' blueprint for A in odds form:

$$\text{Odds}\left(A|B\right) = BF \times \text{Odds}\left(A\right)$$

where BF is the *Bayes factor* Prob ($B|A$)/Prob (B)$|\sim A$) and

$$\text{Odds}\left(A\right) = \text{Prob}\left(A\right) / \text{Prob}\left(\sim A\right).$$

It is customary to remark that a *posterior odds* of A given the observation B equals the Bayes factor times the *prior odds* of A.

Now return to Monty Hall who has just opened door one to reveal a goat just after the contestant had signaled her willingness to open door three. Writing G for goat and C for car, there were three equally-likely scenarios before Monty acted, GCG, GGC, and GCG, but after opening the door, this shrinks to two possibilities of GCG and GGC. As far as the contestant is concerned, these two options have the same chance of occurring and so the prior odds of the event

A = GCG being true is one. Monty will only open a door hiding a goat, and if two goat doors are available to him, he chooses between them at random. So, given event A is true the probability he opens door one is certain since he has no other choice; letting B denote the event that door one is opened this implies that Prob $(B|A) = 1$ whereas Prob $(B|\sim A) = 1/2$ because under this scenario (~A = GGC) he has two options and, picking at random, the probability is $\frac{1}{2}$ as stated. It follows that the Bayes factor is 1 divided by $\frac{1}{2}$ or 2. It is now immediate that the posterior odds of A is 2/1. This corresponds to a probability of $\frac{2}{3}$, as we saw. It is evidently in the contestant's best interest to switch since her probability of getting a car has jumped from $\frac{1}{3}$ to $\frac{2}{3}$. Put differently, the odds of GCG is two in favor to one against.

This analysis and a number of similar examples can be found in another interesting paper of Miller and Sanjurjo [55].

One can encapsulate the confounding properties of randomness by the somewhat trenchant aphorism that "chance and order are in the eye of the beholder." In the first chapter, we saw that chance events can, in the large, array themselves into the orderly pattern of a normal curve and, in the very next chapter, a random process can spew out orderly and predictable binary strings. By contrast a deterministic computer algorithm is able to forge binary strings that to all appearances behave randomly (Chapter 3), and, somewhat analogously, in Chapter 4, computer codes were revealed that spawn algorithmically random strings. Then, in Chapter 5, we witnessed order and disorder comingling promiscuously. Finally, in this last chapter, an observer is left to unravel the quirky idea that random is not always what we think it should be.

One concludes that chance is not only ubiquitous and essential but also profoundly illusive.

Sources and Further Readings

Chapter 1: The Taming of Chance

The introduction to probability and statistics in this chapter is thoroughly elementary and intuitive. To my mind the most authoritative and provocative book on probability remains the classic work *An Introduction to Probability Theory and its Applications* of William Feller [32], where among other topics one will find proofs of Bernoulli's Law of Large Numbers and de Moivre's theorem regarding the normal, or Gaussian, curve. Although Feller's book is very engaging, it is not always easy reading for the beginner. There are two delightful articles by Mark Kac called "What Is Random?" [42], which prompted the title of this book, and "Probability" [43]. There are several additional articles on probability in the collection *The World of Mathematics,* edited by James Newman some years ago [57]. I especially recommend the passages translated from the works of James Bernoulli, Pierre Simon Laplace (from which I also cribbed a quotation), and Jules Henri Poincare. The Francis Galton

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2

quotation also comes from Newman's collection, in an essay by Leonard Tippett called "Sampling and Sampling Error," and Laplace's words are from a translation of his 1814 work on probability in the same Newman set. A philosophical perspective on probability, called simply "Chance," is by Alfred J. Ayer [4].

Technical accounts of statistical tests of randomness, including the runs test, are in the second volume of Donald Knuth's formidable magnum opus *Seminumerical Algorithms* [48].

The idea of chance in antiquity is documented in the book *Gods, Games, and Gambling* by Florence David [25], while the rise and fall of the goddess Fortuna is recounted in a book by Howard Patch [60], from which I took a quotation. Oystein Ore has written an engaging biography of Girolamo Cardano, including a translation of *Liber de Ludo Aleae*, in *Cardano: The Gambling Scholar* [59]. A spirited and very readable account of how the ancients conceptualized the idea of randomness and how this led to the origins of probability is provided in Deborah Bennett's attractive book called *Randomness* [10].

The early history of the rise of statistics in the service of government bureaucracies during the eighteenth and nineteenth centuries is fascinatingly told by Ian Hacking [38] in a book whose title I shamelessly adopted for the present chapter. The work of early probability theorists is also admirably covered in the detailed account by Stephen Stigler *The History of Statistics* [69].

More recent work about randomness that is not treated in the present volume can be found in *Ten Great Ideas About Chance* by Diaconis and Skyrms [27] including a good discussion of the contributions of Bernoulli and Bayes.

The perception of randomness as studied by psychologists is summarized in an article by Maya Bar-Hillel and Willem Wagenaar [7]. More will be said about this topic in the next chapter.

Chapter 2: Uncertainty and Information

The story of entropy and information and the importance of coding is the subject matter of information theory. The original paper by Claude Shannon is tough going, but there is a nice expository article by Warren Weaver that comes bundled with a reprint of Shannon's contribution in the volume *The Mathematical Theory of Communication* [68]. There are several other reasonably elementary expositions that discuss a number of the topics covered in this chapter, including the one by John Pierce [62]. A fascinating account of secret codes is David Kahn's *The Codebreakers* [44]. There are readable accounts of the investigations of psychologists in the perception of randomness by Ruma Falk and Clifford Konold [31] and Daniel Kahneman and Amos Tversky [45], in addition to the previously cited paper by Maya Bar-Hillel and Willem Wagenaar [7].

Approximate entropy is discussed in an engagingly simple way by Fred Attneave in a short monograph called *Applications of Information Theory to Psychology* [3]. Extensions and ramifications of this idea, as well as the term ApEn, can be found in "Randomness and Degrees of Irregularity" by Steve Pincus and Burton Singer [63] and "Not All (Possibly) 'Random' Sequences Are Created Equal" by Steve Pincus and Rudolf Kalman [64]. Among other things, these authors do not restrict themselves to binary sequences, and they permit arbitrary source alphabets. An elementary account of this more recent work in John Casti's "Truly, Madly, Randomly" [18].

Chapter 3: Janus-Faced Randomness

The Janus algorithm was introduced in Bardett's paper "Chance or Chaos" [8]. A penetrating discussion of the mod 1 iterates, and chaos is available in Joseph Ford's "How Random Is a Coin Toss?" [33].

Random number generators are given a high-level treatment in Donald Knuth's book [48], but this is recommended only to specialists. A more accessible account is in Ivar Ekeland's charming book *The Broken Dice* [29]. The rise of chaos in mod 1 algorithms is discussed by Robert May [52].

The quotation from George Marsaglia is the title of his paper [51].

Maxwell's and Szilard's demons are reviewed in two papers having the same tide, "Maxwell's Demon," by Edward Daub [24] and W. Ehrenberg [28]. There is also a most informative paper by Charles Bennett called "Demons, Engines, and the Second Law" [11].

George Johnson's fast-paced and provocative book *Fire in the Mind* [40] also covers many of the topics in this and the next two chapters and is especially recommended for its compelling probe of the interface between science and faith. Susskind and Friedman's book on quantum mechanics [71] is the most lucid treatment of this subject known to me.

Chapter 4: Algorithms, Information, and Chance

I have glossed over some subtleties in the discussion of algorithmic complexity, but the dedicated reader can always refer to the formidable tome by Li and Vitanyi [49], which also includes extensive references to the founders of this field several decades ago, including the American engineer Ray Solomonoff, who, some believe, may have launched the entire inquiry.

There is an excellent introduction to Turing machines and algorithmic randomness in "What Is a Computation," by Martin Davis [26], in which he refers to Turing-Post programs because of the related work of the logician Emil

Post. An even more detailed and compelling account is by Roger Penrose in his book *Shadows of the Mind* [61], and my proof that the halting problem is undecidable in the technical notes is taken almost verbatim from this source.

The best introduction to Gregory Chaitin's work is by Chaitin himself in an article called "Randomness and Mathematical Proof" [20] as well as the slightly more technical "Information-Theoretic Computational Complexity" [21].

The distinction between cause and chance as generators of a string of data is explored more fully in the book by Li and Vitanyi mentioned above.

The words of Laplace are from a translation of his 1814 *Essai philosophique sur les probabilites*, but the relevant passages come from the excerpt by Laplace in Newman's *World of Mathematics* [57]. The Heisenberg quotation is from the article "Beauty and the Quest for Beauty in Science," by S. Chandrasekhar [22].

Quotations from the writing of Jorge Luis Borges were extracted from his essays "Partial Magic in the Quixote" and "The Library of Babel," both contained in *Labyrinths* [15].

Chapter 5: The Edge of Randomness

Charles Bennett's work on logical depth is found in a fairly technical report [12]. A sweeping overview of many of the topics in this section, including the contributions of Bennett, is *The Quark and the Jaguar,* by Murray Gell-Mann [35]. Jacques Monod's influential book is called *Chance and Necessity* [50]. Robert Frost's arresting phrase is from his essay "The Figure a Poem Makes," and Jonathan Swift's catchy verse is from his "On Poetry, a Rhapsody" and is quoted in Mandelbrot's book [50].

The quotations from Stephen Jay Gould come from several of his essays in the collection *Eight Little Piggies* [37].

His view on the role of contingency in the evolutionary history of living things is summarized in the article "The Evolution of Life on the Earth" [36]. The other quotations come from the ardently written books *How Nature Works* [5], by Per Bak, and *At Home in the Universe,* by Stuart Kauffman [47]. Ilya Prigogine's thoughts on nonequilibrium phenomena are contained in the celebrated *Order Out of Chaos,* which he wrote with Isabelle Stengers [65]. Cawelti's book is *Adventure, Mystery, and Romance* [19]. A useful resume of the work of Watts and Strogatz paper *Collective Dynamics of Small World Networks* [74] is in "It's a Small World" by James Collins and Carson Chow [23]. More thorough going is the admirable book *Sync* [70] by Steven Strogatz.

Benoit Mandelbrot's astonishing book is called *The Fractal Geometry of Nature* [50].

Other, more accessible, sources to self-similarly, $1/f$ laws, and details about specific examples are noted in Bak's book mentioned above: the article "Self-organized Criticality," by Per Bak, Kan Chen, and Kurt Wiesenfeld [6]; "The Noise in Natural Phenomena," by Bruce West and Michael Shlesinger [75]; and "Physiology in Fractal Dimensions," by Bruce West and Ary Goldberger [76]. All of the quotations come from one or the other of these sources.

Plankton dynamics as a power law is discussed in Ascioti et al., "Is There Chaos in Plankton Dynamics" [2]. An overview of Power Laws can be found in Montroll and Schleslinger's "On $1/f$ Noises" [56] and Newman's "Power Laws, Pareto Distributions, and Zipf's Law" [58]

Chapter 6: Fooled by Randomness

Near coincidence in birthdays is discussed in Abraham and Moser's "More Birthday Surprises" [1]. Also relevant to the study of coincidences is Falk's "Judgment of

Coincidences" [30]. Essential references to coin tossing are Berreford's "Runs in Coin Tossing" [13] and Bloom's "Singles in a Sequence of Coin Tosses" [14].

An excellent reference for the Poisson distribution is Feller's book [32] mentioned earlier. The Poisson process in Fire Department operations comes from the book edited by Walker et al.'s *Fire Department Deployment Analysis* [73].

Two important papers by Miller and Sanjurjo that are referenced in this chapter are [53] and [54]. For the Monty Hall problem, see the article [72] by Tierney that appeared in the *New York Times* and the book by Jason Rosenhouse called *The Monty Hall Problem* [66]. A nice account of this conundrum can also be found in the enjoyable novel *The Curious Incident of the Dog in the Night-Time* by Mark Haddon [39].

Excellent coverage of cognitive illusions may be found in the book *Thinking, Fast and Slow* by Nobel laureate Daniel Kahneman [46].

Technical Notes

The following notes are somewhat more technical in nature than the rest of the book and are intended for those readers with a mathematical bent who desire more details than were provided in the main text. The notes are strictly optional, however, and in no way essential for following the main thread of the discussion in the preceding chapters.

Chapter 1: The Taming of Chance

1.1.

The material under this heading is optional and is intended to give the general reader some specific examples to illustrate the probability concepts that were introduced in the first chapter.

In three tosses of a coin, the sample space S consists of eight possible outcomes HHH, HHT, HTH, THH, THT, TTH, HTT, TTT, with H for head and T for tail. The triplets define mutually exclusive events and, since they are all

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2

equally-likely, each is assigned probability $\frac{1}{8}$. The composite event "two heads in three tosses" is the subset of S consisting of HHT, HTH, THH (with probability $\frac{3}{8}$), and this is not mutually exclusive of the event "*at least* two heads in three tosses" since this event consists of the three triplets HHT, HTH, THH plus HHH (with probability $\frac{4}{8} = \frac{1}{2}$).

In the example above, HHT and HTH are mutually exclusive events, and the probability of one or the other happening is $\frac{1}{8} + \frac{1}{8} = \frac{1}{4}$. By partitioning S into a bunch of exclusive events, the sum of the probabilities of the union of all these disjoint sets of possibilities must be unity since one of them is certain to occur.

The independence of events A and B is expressed mathematically by stipulating that the event "both A and B take place" *is the product of the separate probabilities* of A and B. Thus, for example, in three tosses of a fair coin, the events "first two tosses are heads" and "the third toss is head," which are denoted, respectively, by A and B, are intuitively independent since there is a fifty-fifty chance of getting a head on any given toss regardless of how many heads preceded it. In fact, since A consists of HHH and HHT while B is HHH, HTH, THH, TTH, it follows that the probability of A equals $\frac{2}{8} = \frac{1}{4}$ and the probability of B equals $\frac{4}{8} = \frac{1}{2}$; however the probability that "both A and B takes place" is $\frac{1}{8}$ since this event consists only of HHH. Observe that the product of the probabilities of A and B is $\frac{1}{4}$ times $\frac{1}{2}$, namely, $\frac{1}{8}$.

One can also speak of events conditional on some other event having taken place. When the event F is known to have occurred, how does one express the probability of some other event E in the face of this information about?

To take a specific case, let us return to the sample space of three coin tosses considered in the previous paragraph, and let F represent "the first toss is a head." Event F has probability $\frac{1}{2}$, as a quick scan of the 8 possible outcomes shows. Assuming that F is true reduces the sample space S to only four elementary events HTT, HTH, HHT, HHH. Now let E denote "two heads in 3 tosses." The probability of E equals $\frac{3}{8}$ if F is not taken into account. However, the probability of E given that F has in fact taken place is $\frac{1}{2}$ since the number of possibilities has dwindled to the two cases HHT and HTH out of the four in F. The same kind of reasoning applies to any finite sample space and leads to the *conditional probability* of E given F. Note that if E and F are independent, then the conditional probability is simply the probability of E because knowledge of F is irrelevant to the happening or non-happening of E.

In the text conditional probability of E knowing that event F has taken place is denoted by prob $(E|F)$ and we saw that it must be proportional to prob $(E \cap F)$. In fact if the constant of proportionality is chosen to be prob (F) then one can write

$$\text{prob}\left(E|F\right) = \text{prob}\left(E \cap F\right) / \text{prob}\left(F\right)$$

That this choice of normalizing constant is appropriate can be seen by choosing E to be equal to F since in this case $E \cap F = F$ and the right side of the equation above is simply one; this is correct because F now constitutes the entire sample space under consideration and its probability should be unity. Note that from this same equation, we get another way of expressing independence of events E and F as prob $(E|F)$ = prob (E). This is evidently equivalent to pr ob $(E \cap F)$ = prob (E) · prob (F).

Since, once again using the equation above,

$$\text{prob}\left(E \cap F\right) = \text{prob}\left(E|F\right)\cdot\text{prob}\left(F\right),$$

then, interchanging the roles of E and F, one can also write

$$\text{prob}\left(E \cap F\right) = \text{prob}\left(F|E\right)\cdot\text{prob}\left(E\right).$$

The right sides of these two relations must then equal each other, and this gives us what is known as *Bayes' Theorem*:

$$\text{prob}\left(E|F\right) = \text{prob}\left(F|E\right)\cdot\text{prob}\left(E\right) / \text{prob}\left(F\right)$$

1.2.

The set of all numbers in the unit interval between zero and one can be identified in a one-to-one manner with the set of all possible non-terminating strings of zeros and ones, provided that strings with an unending succession of ones are avoided. The connection is that any x in the unit interval can be written as a sum

$$x = a_1 / 2 + a_2 / 2^2 + \frac{a_3}{2^3} + \cdots$$

where each a_k, k = 1, 2… denotes either 0 or 1. For example, $3/8 = 0/2 + 1/2^2 + 1/2^3$, with the remaining terms all zero. The number x is now identified with the infinite binary string $a_1a_2a_3$… . With this convention, $x = 3/8$ corresponds to the string 01100…. Similarly, $1/3 = 01010101\ldots$. In some cases two representations are possible, such as $1/4 = 01000\ldots$

and $1/4 = 00111\ldots$ and when this happens, we opt for the first possibility. So far, this paraphrases Appendix B, but now here comes the connection to probability.

Each unending sequence of Bernoulli $1/2$-trials gives rise to an infinite binary string and therefore to some number x in the unit interval. Let S consist of all possible strings generated by this unending random process. Then an event E in S is a subset of such strings, and the *probability of E is taken to be the length of the subset of the unit interval that corresponds to E*. Any two subsets of identical length correspond to events having the same probability.

Conversely, any x in the unit interval gives rise to a string generated by a Bernoulli $1/2$-process. In fact, if x corresponds to $a_1a_2a_3\ldots$, then $a_1 = 0$ if and only if x belongs to the subset $\left[0, 1/2\right)$ having length $1/2$. This means that the probability of a_1 being zero is $1/2$; the same applies if $a_1 = 1$. Now consider a_2. This digit is 0 if and only if x belongs to either $\left[0, 1/4\right)$ or $\left[1/2, 3/4\right)$. The total length of these disjoint intervals sum to $1/2$, and since they define mutually exclusive events, the probability of a_2 being zero is necessarily $1/2$. Obviously the same is true if $a_2 = 1$. Proceeding in this fashion, it becomes evident that each a_k, for $k = 1, 2, \ldots$, has an equal probability of being zero or one. Note that regardless of the value taken on by a_1, the probability of a_2 remains $1/2$ and so a_1 and a_2 are independent. The same independence applies to any succession of these digits. It follows that the *collection of numbers in the unit interval is identified in a one-to-one fashion with all possible outcomes of Bernoulli $1/2$-trials.*

If s is a binary string of length n, let Γ_s denote the set of all infinite strings that begin with s. Now s itself corresponds to some terminating fraction x, and since the first n positions of the infinite string have already been specified, the remaining digits sum to a number whose length is at most $1/2^n$. To see this first note that any sum of terms of the form $a_k/2^k$ can never exceed in value a similar sum of terms $1/2^k$ since a_k is always less than or equal to one. Therefore the infinite sum of terms $a_k/2^k$ starting from $k = n + 1$ is at most equal to the corresponding sum of terms $1/2^k$, and as Appendix A demonstrates, this last sum is simply $1/2^n$.

Therefore, the event Γ_s corresponds to the interval between x and x plus $1/2^n$, and this interval has length $1/2^n$. It follows that the event Γ_s has probability $1/2^n$. Thus, if E is the event "all strings whose first three digits are one," then E consists of all numbers that begin with $1/2 + 1/4 + 1/8 = 7/8$, and this gives rise to the interval between $7/8$ and 1 whose length is $1/8$. Therefore the probability of E equals $1/8$.

1.3.

The simple MATLAB m-file below is for those readers with access to MATLAB who wish to generate Bernoulli p-trials for their own amusement:

File generates a binary string of n Bernoulli trials with a probability p of success (occurrence of digit 1). Entering Bernoulli in the MATLAB workspace returns a prompt for the string length n and success probability p. The output is a string of length n

```
n=input('string size is...');
p=input('probability of a success is...');
ss=[];
for k=1:n
if rand<p
s=1;
else s=0;end
ss=[ss,s];
end
ss
```

1.4.

The next MATLAB m-file allows one to generate the fluctuations of a random walk whose positive and negative excursions correspond to the winnings and losses of Peter and Paul in a coin-tossing game of head and tails as described in the text.

The program simulates the accumulative gains and displays the results in graphic over n tosses (typically 5000 or more) plotted every tenth toss. It also prints out the average gain (which should be close to zero) in contrast to the accumulated gain as well as fraction of time one player is in the lead. Entering coin toss in the MATLAB workspace returns a prompt for the number of tosses *n*.

```
ss=[];feller=[];
n=input('number of trials is    ');
for i=1:n
if rand<.5
s=1;
else s=-1;
end
ss=[ss,s];
gain=sum(ss);
```

```
feller=[feller, gain];
end
z=feller;
plot(feller(1:10:n))
xlabel('number of coin tosses, p=1/2')
ylabel('fluctuation of wins and losses')
for k=1:n
if feller(k)<0
feller(k)=0;else feller(k);end
end
fprintf('the fraction of tosses in which one player
is in the lead:%g.\n',nnz(feller)/n)
fsprintf('the average gain of a
player:%g.\n',mean(ss))
```

Chapter 2: Uncertainty and Information

2.1.

The following comments are intended to clarify the discussion of message compression when there are redundancies.

The Law of Large Numbers discussed in the previous chapter asserts that in a long enough sequence of n Bernoulli p-trials with a likelihood p of a success (namely, a one) on any trial, the chance that the proportion of successes S_n/n deviates from p by an arbitrarily small amount is close to unity. What this says, in effect, is that in a long binary string, it is very likely that the number of 1's, which is S_n, will nearly equal np, and, of course, the number of 0's will nearly equal $n(1 - p)$. The odds are then overwhelmingly in favor that such a string will be close to a string in which the digit one appears np times and the digit zero appears $n(1 - p)$ times. This is because successive Bernoulli trials are independent and, as we know, the probability of a

particular string of independent events, namely, the choice of zero or one, must equal the product of the probabilities of the individual events. The product in question is

$$p^{np}\left(1-p\right)^{n(1-p)}$$

and it is equal to $\frac{1}{2}^{nh}$ for some suitable h yet to be determined. Taking logarithms to the base two of both sides (see Appendix C), we obtain

$$np\log p + n\left(1-p\right)\log\left(1-p\right) = -nh\log 2$$

One can now solve for h and obtain, noting that log2 = 1,

$$h = -\left\{p\log p + \left(1-p\right)\log\left(1-p\right)\right\}$$

which is the entropy H of the source. It is therefore likely that a binary string of length n has very nearly a probability 2^{-nH} of occurring. Stated differently, the fraction of message strings having a probability 2^{-nH} of appearing is nearly one. It follows that there must be altogether about 2^{nH} such message strings and it is highly improbable that an actual message sequence is not one of them. Even so, these most likely messages constitute a small fraction of the 2^n strings that are possible, since $2^{nH}/2^n$ is tiny for large n. An exception to this is when $p = \frac{1}{2}$ because H equals unity in that case.

2.2.

This note expands on the Shannon coding scheme.

An alphabet source has k symbols whose probabilities of occurrence p_1, p_2, …, p_k are assumed to be listed in

decreasing order. Let P_r denote the sum $p_1 + \cdots + p_{r-1}$, for $r = 1, \ldots, k$, with P_1 set to 0. The binary expansion of P_r is a sum $a_i/2^i$, for $i = 1, 2, \ldots$ (Appendix B). Truncate this infinite sum when i is the first integer, call it l_r, that is no smaller than $-\log p_r$ and, by default, less than $1 - \log p_r$. The binary code for symbol r is then the string $a_1 a_2 \ldots a_{l_r}$.

For example, for a source alphabet a, b, c, and d with corresponding probabilities $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, and $\frac{1}{8}$, $P_1 = 0$, $P_2 = \frac{1}{2} + \frac{0}{2^2}$, $P_3 = \frac{1}{2} + \frac{1}{2^2} + \frac{0}{2^3}$, and $P_4 = \frac{1}{2} + \frac{1}{2^2} + \frac{1}{2^3}$. Now $-\log \frac{1}{2} = 1$, $-\log \frac{1}{2^2} = 2$, and $-\log \frac{1}{2^3} = 3$, and so, following the prescription given above, the code words for the source symbols a, b, c, and d are 0, 10, 110, and 111, respectively, with lengths 1, 2, 3, and 3.

The Shannon code assigns short words to symbols of high probability and long codes to symbols of low probability.

2.3.

In defining ApEn(k), an alternate approach would be to formally compute the *conditional entropy* of a k-gram given that we know its first $k - 1$ entries. This expression, which is not derived here, can be shown to equal $H(k) - H(k - 1)$, and this is identical to the definition of ApEn(k) in the text.

2.4.

A simple program called atneave and written as a MATLAB m-file implements the ApEn algorithm and is included here for those readers who have access to MATLAB and would like to test their own hand-tailored sequences for randomness. Entering atneave into the MATLAB workspace

returns a prompt for the size m of the pattern length (not to exceed four). The output is the value of ApEn for each m. to use the code one must input the binary string u that is being tested in the workspace, written as a column vector u'. Also one inputs the length n of u. For example, enter

```
u = [1 0 1 0 1 0 1 0]'; n= 8;
A = [1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
     1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0
     1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0
     1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0];
%
m=input('pattern length     ');
ApEn=[];h=[];
for k=1:m
N=n-k+1;
p=2^k;
Q=A(1:k,1:2^(4-k):16);
U=zeros(N,p);
r=zeros(1,p);
for j=1:p
for i=1:N
U(i,j)=norm(u(i:i+k-1)-Q(:,j),inf);
end
r(j)=sum(spones(find(U(:,j)==0)));
end
rr=find(r>0);r=r(rr);
h(k)=log2(N)-sum(r.*log2(r))/N;
end
ApEn(1)=h(1);
fprintf('value of ApEn(1)      :%g.\n',ApEn(1))
% if m is greater than one :
for s=2:m
ApEn(s)=h(s)-h(s-1);
end
ApEn
```

Chapter 3: Janus-Faced Randomness

Some of the missing details regarding the use of symbolic dynamics are filled in here. Our concern is how to represent the action of the Janus algorithm and its inverse.

3.1.

Begin by writing u_0 as a sum $\frac{a_1}{2}+\frac{a_2}{4}+\frac{a_3}{8}+\ldots$ as described in the text, which implies that $a_1a_2a_3\ldots$ represents u_0. Then u_1 is either $u_0/2$ or $\frac{1}{2}+u_0/2$. In the first instance, the sum becomes $\frac{a_1}{4}+\frac{a_2}{8}+\frac{a_3}{16}+\cdots$, and in the second, it is $\frac{1}{2}+\frac{a_1}{4}+\frac{a_2}{8}+\cdots$.

In either case, you obtain u_1 as $\frac{b_1}{2}+\frac{a_1}{4}+\frac{a_2}{8}+\ldots$ with b_1 being zero or one. Therefore u_1 is represented by the string $b_1a_1a_2\ldots$. Similarly, u_2 becomes $\frac{b_2}{2}+\frac{b_1}{4}+\frac{a_2}{8}+\ldots$, and so u_2 is identified by the sequence $b_2b_1a_1a_2\ldots$.

3.2.

By contrast with the previous note, the mod 1 scheme starts with a seed v_0, which can be expressed as $\frac{q_1}{2}+\frac{q_2}{4}+\ldots$, and so the binary sequence $q_1q_2q_3\ldots$ represents v_0. If v_0 is less than $\frac{1}{2}$, then $q_1 = 0$, and q_1 is 1 when v_0 is greater than or equal to $\frac{1}{2}$.

In the first case, $v_1 = 2v_0 \pmod 1$ equals $\frac{q_2}{2}+\frac{q_3}{8}+\ldots(\mathrm{mod}\,1)$, while

in the second, it is $1+\frac{q_2}{2}+\frac{q_3}{8}+\ldots(\text{mod } 1)$. By definition of mod 1, only the fractional part of v_1 is retained, and so 1 is deleted in the last case, while mod 1 is irrelevant in the first because the sum is always less than or equal to 1 (see Appendix A). In either situation, the result is the same: v_1 is represented by the sequence $q_2q_3q_4\ldots$.

3.3.

I wish to sketch a proof of the result of George Marsaglia about random numbers in a plane. Denote two successive iterates of the mod 1 algorithm by $x_k = 2^k x_0 \pmod 1$ and $x_{k+1} = 2^{k+1}x_0 \pmod 1$, where x_0 is a fraction (a truncated seed value) in lowest form p/q for suitable integers p, q. The pair (x_{k+1}, x_k) is a point in the unit square. Now let a, b denote two numbers for which $a + 2b = 0 \pmod q$. If q is 9, for instance, then $a = 7$, $b = 10$ satisfy $a + 2b = 27$, which is a multiple of 9 and therefore equal to zero (mod 9). There are clearly many possible choices for a and b, and there is no loss in assuming that they are integers. It now follows that $ax_k + bx_{k+1} = 2^k x_0(a + 2b) \bmod 1$, and since $a + 2b$ is some multiple m of q, we obtain $ax_k + bx_{k+1} = mp2^k \pmod 1$, since $x_0 = p/q$ and the two q's cancel. But $mp2^k$ is an integer, and so it must equal zero mod 1. This means that $ax_k + bx_{k+1} = 0, \pm 1, \pm 2, \pm 3, \ldots$, which are the equations of straight lines in the plane. What this says, in effect, is that the "random number" pair (x_{k+1}, x_k) must lie on one of several possible lines that intersect the unit square. The same argument applies to triplets of points, which can be shown to lie on planes slicing the unit cube, and to k-tuples of points which must reside on hyper-planes intersecting k-dimensional hyper-cubes.

3.6.

Among the algorithms that are known to generate deterministic chaos, the most celebrated is the algorithm defined by

$$x_n = r\, x_n \left(1 - x_n\right)$$

for $n = 1, 2, \ldots$ and a seed x_0 in the unit interval. The parameter r ranges in value from 1 to 4, and within that span, there is a remarkable array of dynamical behavior. For r less than three, all the iterates tend to a single value as n gets larger. Thereafter, as r creeps upward from three, the successive iterates become more erratic, tending at first to oscillate with cycles that get longer and longer as r increases, and, eventually, when r reaches four, there is full blown chaos. The intimate details of how this all comes to pass have been told and retold in a number of places and I'll not add to these accounts. My interest is simply to mention that in a realm where the iterates are still cycling benignly, with r between 3 and 4, the intrusion of a little bit of randomness jars the logistic algorithm into behaving chaotically. It therefore becomes difficult, if not impossible, to disentangle whether one has a deterministic algorithm corrupted by noise or a deterministic algorithm that, because of sensitivity to initial conditions, only mimics randomness. This dilemma is a sore point in the natural sciences whenever raw data needs to be interpreted either as a signal entwined with noise or simply as a signal so complicated that it only appears noisy.

Chapter 4: Algorithms, Information, and Chance

4.1.

I provide a succinct account of Turing's proof that the halting problem is undecidable. Enumerate all possible Turing programs in increasing order of the size of the numbers represented by the binary strings of these programs and label them as T_1, T_2, … . The input data, also a binary string, is similarly represented by some number $m = 1, 2, \ldots$, and the action of T_n on m is denoted by $T_n(m)$. Now suppose that there is an algorithm for deciding when any given Turing program halts when started on some particular input string. More specifically, assume the existence of a program P that halts whenever it has demonstrated that some Turing computation never halts. Since P depends on both n and m, we write it as $P(n, m)$; $P(n, m)$ is alleged to stop whenever $T_n(m)$ does not halt.

If m equals n, in particular, then $P(n, n)$ halts if $T_n(n)$ doesn't halt. Since $P(n, n)$ is a computation that depends only on the input n, it must be realizable as one of the Turing programs, say $T_k(n)$, for some integer k, because T_1, T_2, … encompass all possible Turing machines. Now focus on that value of n that equals k to obtain $P(k, k) = T_k(k)$, from which we assert that if $P(k, k)$ stops, then $T_k(k)$ doesn't stop. But here comes the denouement: $P(k, k)$ equals $T_k(k)$, and so the statement says that *if* $T_k(k)$ *halts then* $T_k(k)$ *doesn't halt!* Therefore it must be that $P_k(k)$ in fact does not stop for if it did then it could not, a blatant contradiction. It follows from this that $P(k, k)$, which is identical to $T_k(k)$, also cannot stop and so the program P is *not able to decide* whether the particular computation $T_k(k)$ halts *even though you know that it does not.*

4.2.

In the *Technical Notes* to Chapter 1, we let Γ_s denote the set of all infinite strings that begin with s for a binary string s of length $l(s)$. Now s itself corresponds to a real number x that is a terminating fraction, and since the first $l(s)$ positions of the infinite string have already been specified, the remaining digits sum to a number whose length is at most $1/2^{l(s)}$, as we saw. Therefore, the event Γ_s corresponds to the interval between x and x plus $1/2^{l(s)}$, and this interval has length $1/2^{l(s)}$. It follows that the set Γ_s of all real numbers that begin with x has probability $1/2^{l(s)}$. Specifically, $\Gamma_s = \left[x, x + 1/2^{l(s)}\right)$.

A binary string s' is a *prefix* to some longer string s if there is a string z such that $z = s's$. If we are given a collection of binary strings of finite length, none of them prefixes of any other string, then any concatenation of them can be unambiguously decoded into the individual strings of the collection. For example, if 10, 110, 010, and 0110 are four strings, then 10 is not a prefix of any of the others, whereas 010 is a prefix of 0101.

In the last section of Chapter 4, we are given a binary string whose complexity is determined by the shortest binary computer program, namely, a string s of length $l(s)$, that outputs the given string when started with no input. The *algorithmic probability* of any given finite length string is defined to be $2^{-l(s)}$. Thus, among all $2^{l(s)}$ computer programs of length $l(s)$, the a priori probability that some specific program is chosen to generate the given string is $2^{-l(s)}$. Since probabilities of disjoint events must sum to something not exceeding one, we need to impose a technical

condition that the shortest length programs which generate any given finite length string must constitute a prefix-free set. It is possible to ensure this by employing an argument known as Kraft's inequality.

To establish this inequality note that if s' is a prefix to s, then the interval Γ_s is contained within the interval $\Gamma_{s'}$. This follows because if x' is the real number corresponding to the binary s', then evidently $x' < x$ where, as before, x is the real number matching s, and moreover, since the length of s exceeds that of s', $2^{-l(s)}$ is less than $2^{l(s')}$ and so Γ_s resides within the interval $\left[x', x' + 1/2^{l(s')}\right)$.

Now suppose that s' and s are two arbitrary strings with $s' < s$, meaning that corresponding real numbers x' and x satisfy $x' < x$ and assume that s' is *not* a prefix of s. If $\Gamma_{s'}$ and Γ_s overlap, this means that s belongs to $\Gamma_{s'}$ and so s' is a prefix of s, a contradiction. It follows that in any collection of strings, none of which are prefixes of each other, the interval Γ_s that is identified with a given string is disjoint from the intervals associated with other strings. Since the Γ intervals all lie within the unit interval $[0, 1]$ and are all disjoint, the sum of their lengths must be less than or equal to one. This is the requisite inequality.

The shortest length programs which regurgitate a binary string starting on an empty tape are prefix-free if they each terminate with the instruction "stop" that does not appear elsewhere within the program; these halting programs evidently cannot be prefixes of each other, and therefore the intervals Γ and the probabilities they define will indeed sum to one, as they should.

4.3.

It was mentioned that log$n$ bits suffice to code an integer n in binary form. This is shown in Appendix C. Also, since 2^n increases much faster than does n, $\log(2^n/n)$ will eventually exceed $\log 2^m$ for any fixed integer m, provided that n is taken large enough. This means that $n - \log n$ is greater than m for all sufficiently large n, a fact that is needed in the proof of Chaitin's theorem.

4.4.

It has been pointed out several times that a requirement of randomness is that a binary sequence define a *normal number*. This may be too restrictive a requirement for finite sequences, however, since it implies that zeros and ones should be equally distributed. Assuming that n is even, the number of strings containing exactly $n/2$ ones is approximately proportional to the reciprocal of the square root of n (see the paper by Pincus and Singer [64]). This means that randomness in the strict sense of normality is very unlikely for lengthy but finite strings. The problem is that one is trying to apply a property of infinite sequences to truncated segments of such sequences. This shows that randomness defined strictly in terms of maximum entropy or maximum complexity, each of which imply normality, may be too limiting. The reader may have noticed the more pragmatic approach adopted in the present chapter, in which randomness as maximum complexity is re-defined in terms of d-randomness for d small relative to n. With this proviso, most long strings are now very nearly random, since the complexity of a string s is stipulated to be comparable (but not strictly equal) to its length. The same ruse applies in

general to any lengthy but finite block of digits from Bernoulli $\frac{1}{2}$-trials since the Law of Large Numbers ensures that zeros and ones are very nearly equally distributed with a probability that increases with the length of the block. So we can now say that most long sequences from a Bernoulli $\frac{1}{2}$-process are random with the proviso of "very nearly." In Chapter 2, the notion of approximate entropy, which measures degrees of randomness, implicitly defines a binary block to be random if its entropy is very nearly maximum. As before, most long strings from Bernoulli $\frac{1}{2}$-trials are now random in this sense.

4.5.

Champernowne's normal number, which will be designated by the letter *c*, has already been mentioned several times in this book, but I cannot resist pointing out another of its surprising properties. The binary representation of *c* is, as we know, the sequence in which 01 is followed by the four doublets 00 01 10 11, then followed by the eight triplets 000 001…, and so forth. Every block of length *k*, for any positive integer *k*, will eventually appear within the sequence. If the inverse of the Janus algorithm, namely, the mod 1 algorithm, is applied to *c*, then, after m iterates, its *m* leftmost digits will have been deleted. This means that if *x* is any number in the unit interval, the successive iterates of the mod 1 algorithm applied to *c* will eventually match the first *k* digits of the binary representation of *x*; one simply has to choose *m* large enough for this to occur. Thus *c* will approximate *x* as closely as one desires simply by picking *k* sufficiently large that the remaining digits, from *k* + 1 onward, represent a negligible error. It follows that the

iterative scheme will bring Champernowne's number in arbitrary proximity to any x between zero and one: the successive iterates of c appear to jump pell-mell all over the unit interval. Though the iterates of the inverse Janus algorithm follow a precise rule of succession, the actual values seem random to the uninitiated eye as it attempts to follow what seems to be a sequence of aimless moves.

Chapter 5: The Edge of Randomness

5.1.

I want to establish that the power law relation between magnitude and frequency is scale invariant. To see this, multiply the frequency x by some constant c greater than one, and denote the magnitude by $s(x) = 1/x^b$ for b not less than one. Then $s(cx) = c^{-b}s(x)$, a constant multiple of $s(x)$. Therefore, $s(cx)$ has the same shape as $s(x)$, which is self-similarity. Moreover, the integral of $s(x)$ between two frequencies x_1 and x_2 represents the total variability in a data series within this range (this is first but not the last time in this book that a calculus concept is mentioned!) and from this we obtain, after a change of variables $x = cu$:

$$\int_{x_1}^{x_2} f(x)\,dx = \int_{x_1/c}^{x_2/c} f(cu)\,du = c^{1-b}\int_{x_1/c}^{x_2/c} f(x)\,dx$$

Therefore the total variability within the interval (x_1, x_2) is a multiple of the total variability within the wider frequency band (cx_1, cx_2). When $b = 1$, the sum of the magnitudes within each band are equal.

5.2.

Using logarithms (Appendix C), it is readily seen that the logarithm of $1/x^b$ is approximately equal to some constant minus the logarithm of x, which is the equation of a straight line with negative slope. This is the characteristic shape of a power law when plotted on a logarithmic scale.

Chapter 6: Fooled by Chance

6.1.

To complete the proof of Berresford's result note that the probability of i is $P(n-1,k)$ whereas ii has the probability $(1-P(n-k-1,k)) \bullet p^k(1-p)$. The quantity in parenthesis is multiplied by the probability of success p a total of k times since we have k independent Bernoulli trials for k heads, and this is followed by the probability $1-p$ of a failure (namely, a tail) which is also multiplied because, once again, of independence. Because i and ii are disjoint, the separate probabilities are added:

$$P(n,k) = P(n-1,k) + \left(1 - P\left(n-(k+1)\right)\right) p^k (1-p)$$

To begin the recursion one has a couple of immediate identities: $P(n,k) = 0$ for all $n < k$, and $P(k,k) = p^k$ since a k-clump occurs in only one way in k tosses.

I provide here the MATLAB code clumps.m for computing the probability of runs of length at least k using Berresford's result. Entering clumps in the MATLAB workspace returns a prompt for the number n of tosses, the minimum run size k, and the probability p of getting a head on each toss of a coin:

```
n=input('number of coin tosses      ');
k=input('minimum size of a run      ');
p=input('probability of a head      ');
r=p^k;
for j=1:k-1
a(j)=0;
end
a(k)=r;a(k+1)=r*(2-p);
for s=k+2:n
a(s)=a(s-1)+(1-a(s-k-1))*r*(1-p);
end
fprintf('prob of runs of heads of size no less than
k   :%g.\n',a(n))
```

6.2.

Full disclosure requires that the expression for the Poisson distribution P_k be given. It is $P_k = \lambda^k e^{-\lambda}/k!$ where $e^{-\lambda}$ is the exponential function that some readers may have encountered in a first calculus course (this is the second and last time calculus is mentioned in this book) and $k!$ is the factorial expression $k\cdot(k-1)\cdot(k-2)\cdot\ldots\cdot 1$. If one sums these terms for k = 0, 1, 2, …, they add to one, as befits a probability.

There is a very useful relation between P_k and the probability B_k of obtaining k successes in a succession of n Bernoulli trials with constant probability p for a success on each trial (this probability is usually called the *Binomial* distribution, valid for each k up to n. For n sufficiently large P_k is a close approximation to B_k with $\lambda = np$, a fact that is proven in virtually all texts on probability theory. An application appears below).

6.3.

To compute the probability that in a room of N randomly chosen people, at least two have the same birthday assuming 365 days in a year and an equal probability $1/365$ that a birthday occurs on any chosen day. We are in effect asking for the probability 1 – probability no two people share a birthday. There are $n = N(N-1)/2$ ways of choosing two distinct people out of N. The reason is that the first person can be picked in N ways, and for each of these possibilities, there are now $N-1$ ways of picking the second person. Since the pair can be chosen in two different ways, one divides by 2 to avoid double counting. So now we apply the Poisson approximation discussed in 6.2 with $\lambda = np = N(N-1)/2 \cdot 1/365$ to obtain the probability of no shared birthdays (namely, $k = 0$ in the binomial distribution) equal to $e^{-N(N-1)/2(365)}$. The probability we seek is one minus this quantity, and the table below gives the result for different values of N:

N	Probability
10	.116
20	.406
30	.696
40	.882

With 30 individuals, the probability of shared birthdays exceeds $1/2$, and in fact this is already true when $N = 23$.

On the other hand, if we ask for coincident birthdays between *myself* and someone else, the probability for this is quite different. There are 364 opportunities for someone else to have a different birthday for myself and the probability of that is 364/365. With n randomly chosen individuals, the probability that none of them have my birthdate is, by

independence, $(364/365)^n$ and so the probability that least one individual matches my birthday is one minus this quantity. In order that the likelihood of this event exceed $\frac{1}{2}$ one requires n to be 253. It takes that many people for a coincidence with *me*.

Appendix A: Geometric Sums

The notation S_m is used to denote the finite sum $1 + a + a^2 + a^3 + \ldots + a^m$, where a is any number, many positive integer, and a_m is the mth power of a. This is called a geometric sum, and its value, computed in many introductory mathematics courses, is given by

$$S_m = \left(1 - a^{m+1}\right) / \left(1 - a\right).$$

It is straightforward to see, for example, that if a equals 2, the finite sum S_m is $2^{m+1} - 1$.

When a is some number greater than zero and less than one, the powers a^{m+1} get smaller and smaller as m increases (e.g., if $a = 1/3$, then $a^2 = 1/9$, $a^3 = 1/27$, and so forth) and so the expression for S_m tends to a limiting value $1/(1 - a)$ as m increases without bound. In effect, the infinite sum $1 + a + a^2 + a^3 + \ldots$ has the value $1/(1 - a)$.

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2

In a number of places throughout the book, the infinite sum $a + a^2 + a^3 + \ldots$will be required, which starts with a rather than 1, and this is 1 less than the value $1/(1 - a)$, namely, $a/(1 - a)$.

Of particular interest is the case in which a equals $\frac{1}{2}$, and it is readily seen that $\frac{1}{2} + \frac{1}{2}^2 + \frac{1}{2}^3 + \ldots$ equals $\frac{1}{2}$ divided by $\frac{1}{2}$, namely, 1.

There is also a need for the infinite sum that begins with a^{m+1}: $a^{m+1} + a^{m+2} + a^{m+3} + \ldots$. This is evidently the difference between the infinite sum of terms that begins with a, namely, $1/(1 - a)$, and the finite sum that terminates with a_m, namely, S_m; the difference is $a^{m+1}/(1 - a)$.

The formulas derived so far are summarized below in a more compact form by using the notation Σ_m to indicate the infinite sum that begins with a^{m+1} for any $m = 0, 1, 2, \ldots$. This shorthand is not used in the book proper and is introduced here solely as a convenient mnemonic device:

$$\sum_m = a^{m+1} / (1 - a);$$

$$\sum_0 = a / (1 - a).$$

Here is one example that actually occurs in the book:

Find the finite sum of powers of 2 from 1 to $n - c - 1$ for some integer c smaller than n, namely, the sum $2 + 4 + 8 + \ldots + 2^{n-c-1}$; this is 1 less than S_{n-c-1}, with the number a set equal to 2. A simple computation establishes that the term S_{n-c-1} reduces to $2^{n-c} - 1$ and therefore the required sum is $2^{n-c} - 2$. This is used in Chapter 4.

Appendix B: Binary Notation

Throughout the book, there are frequent references to the binary representation of whole numbers, as well as to numbers in general within the interval from 0 to 1. Let us begin with the whole numbers $n = 1, 2, \ldots$.

For each integer n, find the largest possible power of 2 smaller than n and call it $2k_1$. For instance, if $n = 19$, then $k_1 = 4$, since 2^4 is less than 19, which is less than 2^5.

The difference between n and $2k_1$ may be denoted by r_1. Now find the largest power of 2, call it $2k_2$, that is smaller than r_1, and let r_2 be the difference between r_1 and $2k_2$. Repeat this procedure until you get a difference of zero, when n is even, or one, when n is odd. It is evident from this finite sequence of steps that n can be written as a sum of powers of 2 (plus 1, if n is odd).

For example, $20 = 16 + 4 = 2^4 + 2^2$. Another instance is $77 = 64 + 8 + 4 + 1 = 2^6 + 2^3 + 2^2 + 2^0$ (by definition, $a^0 = 1$ for any positive number a). In general, any positive integer n may be written as a finite sum

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2

$$n = a_{m-1}2^{m-1} + a_{m-2}2^{m-2} + \ldots + a_1 2 + a_0, (\star)$$

where the m coefficients a_{m-1}, …, a_0 are all either 0 or 1 and the leading coefficient a_{m-1} is always 1.

The binary representation of n is the string of digits $a_{m-1}a_{m-2}\ldots a_0$ in the sum (★), and the string itself is often referred to as a *binary string*.

In the case of 20, $m = 4$, $a_4 = a_2 = 1$ and $a_3 = a_1 = a_0 = 0$. The binary representation of 20 is therefore 10100.

For $n = 77$, m is 6 and $a_6 = a_3 = a_2 = a_0 = 1$,while $a_5 = a_4 = a_1 = 0$. The binary representation of 77 is correspondingly 1001101.

There are 2^m *binary strings of length m*, since each digit can be chosen in one of two ways, and for each of these, the next digit can also be chosen in two ways, and as this selection process ripples through the entire string, the number 2 is multiplied by itself m times.

Each individual string $a_{m-1}a_{m-2}\ldots a_0$ *corresponds to a whole number between* 0 *and* $2^m - 1$. The reason why this is so is to be found in the expression (★) above: The integer n is never less than zero (simply choose all the coefficients to be zero), and it can never exceed the value obtained by setting all coefficients to one. In the latter case, however, you get a sum $S_{m-1} = 1 + 2 + 2^2 + \ldots + 2^{m-1}$, and from Appendix A, this sum equals $2^m - 1$.Since each of the 2^m possible strings represents an integer, these must necessarily be the 2^mwhole numbers ranging from 0 to 2^{m-1}.

As an illustration of the preceding paragraph, consider all $2^3 = 8$ strings of length 3. It is evident from (★) that the following correspondence holds between the eight triplet strings and the integers from 0 or 7:

000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Tum now to the *unit interval*, designated by [0, 1], which is the collection of all numbers x that lie between 0 and 1. The number x is an infinite sum of powers of $\frac{1}{2}$:

$$x = b_1 / 2^1 + b_2 / 2^2 + b_3 / 2^3 + \ldots + b_k / 2^k + \ldots . (\star\star)$$

To see how (★★) comes about one engages in a game similar to "twenty questions," in which the unit interval is successively divided into halves, and a positive response to the question "is x in the right half?" result in a power of $\frac{1}{2}$ being added in; a negative reply accrues nothing. More specifically, the first question asks whether x is in the interval $\left(\frac{1}{2}, 1\right]$, the set of all numbers between $\frac{1}{2}$ and 1 not including $\frac{1}{2}$. If so, add $\frac{1}{2}$ to the sum, namely, choose b_1 to be 1. Otherwise, if x lies within the interval $\left[0, \frac{1}{2}\right]$, the set of numbers between 0 and $\frac{1}{2}$ inclusively, put b_1 equal to 0. Now divide each of these subintervals in half again, and repeat the questioning: If b_1 is 1, ask whether the number is within $\left(\frac{1}{2}, \frac{3}{4}\right]$ or $\left(\frac{3}{4}, 1\right]$. In the first instance let b_2 equal 0, and set b_2 to 1 in the second. The same reasoning applies to

each half of $\left[0, \frac{1}{2}\right]$ whenever b_1 is 0. Continuing in this manner, it becomes clear that b_k is 0 or 1 depending on whether x is within the left or right half of a subinterval of length $\frac{1}{2^k}$. The subintervals progressively diminish in width as k increases, and the location of x is captured with more and more precision. The unending series of terms in (★★) represents successive halvings carried out ad infinitum.

The binary expansion of x is defined as the infinite binary string $b_1b_2b_3\ldots$.

As an example consider $= \frac{1}{3}$, which is the sum $\frac{1}{4} + \frac{1}{16} + \frac{1}{64} + \ldots$. This sum can also be written as $\frac{1}{4} + \frac{1}{4^2} + \frac{1}{4^3} + \ldots$, and from Appendix A, we see that the latter sum is represented by Σ_0 for $a = \frac{1}{4}$, namely, $\frac{1}{4}$ divided by $\frac{3}{4}$ or $\frac{1}{3}$. The binary expansion of $\frac{1}{3}$ is therefore the repeated infinite pattern 01010101…. Another instance is the number $11/16 = \frac{1}{2} + \frac{1}{8} + \frac{1}{16}$, and the binary expansion is then simply 1011000… with an unending trail of zeros. For numbers that are not fractions, such as $\pi/4$, the corresponding binary string is less obvious, but the previous argument shows that there is, nevertheless, some sequence of zeros and or ones that represents the number.

A number like $11/16$ can also be given, in addition to the expansion provided above, the binary expansion 1010111… with 1 repeated indefinitely. This is because the sum of this infinite stretch of ones is, according to Appendix A, equal to $\frac{1}{16}$, and this is added to 1010. Whenever two such expansions exist, the one with an infinity of zeros is the expansion of choice.

With this proviso every number is the unit interval has a well-defined binary expansion, and conversely, every such

expansion corresponds to some specific number in the interval; *there is therefore a unique identification between every one of the numbers in the unit interval and all possible infinite binary strings.* In technical note 1.2, it is further established that *all infinite binary strings are uniquely identified with all possible outcomes of a Bernoulli* $\frac{1}{2}$ *-process.*

Appendix C: Logarithms

The properties of logarithms used in Chapters 2 and 4 to discuss randomness in the context of information and complexity are reviewed here.

The *logarithm*, base b, of a positive number x is the quantity y that is written as $\log x$ and defined by the property that $b^y = x$. *The only base of interest in this book is* $b = 2$, and in this case, $2^{\log x} = x$.

By convention, a^0 is defined to be 1 for any number a. It follows that $\log 1 = 0$. Also, it is readily apparent that $\log 2 = 1$.

Since 2^{-y} means $1/2^y$, then $-\log x = 1/x$ whenever $y = \log x$. Taking the logarithm of a power a of x gives $\log x^n = a \log x$. This is because $x = 2^y$, and so $x^a = (2^y) = 2^{ay} = 2^{a \log x}$. For example, $\log 3^4 = 4 \log 3$.

Whenever u is less than v, for two positive numbers u, v, the definition of logarithm shows that $\log u$ is less than $\log v$.

It is always true that $\log x$ is less than x for positive x. Here are two examples to help convince you of this fact:

$$\text{Log}x = 1 \text{ when } x = 2.$$

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2

When $x = 13$, then 13 is less than 16, which is the same as 2^4, and as you saw above, this equals 4 log 2, namely, 4. Therefore, log13 is less than 13.

Let u and v be two positive numbers with logarithms log$u$ and log$v$. Then the product uv equals to $2^{\log u} 2^{\log v}$, which is the same as $2^{(\log u + \log v)}$. It now follows that $\log uv = \log u + \log v$. A similar argument establishes that the logarithm of a product of k numbers is the sum of their respective logarithms, for any positive integer k.

Here is a simple application of logarithms that appears in Chapter 4:

From Appendix B, it is known that an integer n may be written as a finite sum $n = a_k 2^k + a_{k-1} 2^{k-1} + \ldots + a_1 2^1 + a_0$,for some integer k, where the coefficients a_1 are either 0 or 1, $i = 0, 1, \ldots, k - 1$, and $a_k = 1$.This sum is always less than or equal to the sum obtained by setting all coefficients a_i to 1, and by Appendix A, that equals $2^{k+1} - 1$, which, in turn, is evidently less than 2^k. On the other hand, n is never less than the single term 2^{k+1}. Therefore, n is greater than or equal to 2^k and less than 2^{k+1}. Taking logarithms of 2^k and 2^{k+1} shows that log$n$ is no less than k but less than $k + 1$. It follows that the number of bits needed to describe the integer n in binary form is, roughly, log$n$.

References

Note: the items marked with * are especially recommended to the general reader

1. Abraham, M., Moser, W.: More birthday surprises. Am. Math. Mon. **77**, 856–858 (1970)
2. Ascioti, A., Beltrami, E., Carroll, O., Wireck, C.: Is there chaos in plankton dynamics? J. Plankton Res. **15**, 603–617 (1993)
3. Attneave, F.: Applications of Information Theory to Psychology. Holt, Rinehart and Winston (1959)
4. Ayer, A.: Chance. Sci. Am. **213**, 44–54 (1965).*
5. Bak, P.: How Nature Works. Springer-Verlag (1996).*
6. Bak, P., Chen, K., Wiesenfeld, K.: Self-organized criticality. Phys. Rev. A. **38**, 364–374 (1988)
7. Bar-Hillel, M., Wagenaar, W.: The perception of randomness. Adv. Appl. Math. **12**, 428–454 (1991)
8. Bartlett, M.: Chance or Chaos? J. R. Stat. Soc. Ser. A. **153**, 321–347 (1990)
9. Beltrami, E., Mendelsohn, J.: More thoughts regarding Di Maggio's 1941 streak. Baseb. Res. J. **39**, 31–34 (2010)

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2

10. Bennett, D.: Randomness. Harvard University Press (1998).∗
11. Bennett, C.: Demons, engines, and the second law. Sci. Am. **255**, 108–116 (1987).∗
12. Bennett, C.: Logical depth and physical complexity. In: Henken, R. (ed.) The Universal Turing Machine: A Half-Century Survey. Springer-Verlag (1995)
13. Berresford, G.: Runs in coin tossing: randomness revealed. Coll. Math. J. **33**, 391–394 (2002)
14. Bloom, D.: Singles in a sequence of coin tosses. Coll. Math. J. **29**, 120–127 (1998)
15. Borges, J.: Labyrinths. New Directions (1964)
16. Browne, M.: Many small events may add up to one mass extinction. New York Times (Sept 2, 1997).∗
17. Calude, C.: Who is afraid of randomness? Technical report CDMTCS-143, University of Auckland (2000)
18. Casti, J.: Truly, madly, randomly. New Sci. **155**, 32–35 (1997).∗
19. Cawelti, J.: Adventure, Mystery, and Romance. University of Chicago Press (1976)
20. Chaitin, G.: Randomness and mathematical proof. Sci. Am. **232**, 47–52 (1975).∗
21. Chaitin, G.: Information-theoretic computational complexity. IEEE Trans. Inf. Theory. **IT-20**, 10–15 (1974)
22. Chandrasekhar, S.: Beauty and the quest for beauty in science. Phys. Today. **32**, 25–30 (1979)
23. Collins, J., Chow, C.: It's a small world. Nature. **393**, 409–410 (1998).∗
24. Daub, E.: Maxwell's demon. Stud. Hist. Phil. Sci. **1**, 213–227 (1970).∗
25. David, F.: Games, Gods, and Gambling. Hafner Publishing Co. (1962).∗
26. Davis, M.: What is a computation? In: Steen, L. (ed.) Mathematics Today. Springer-Verlag (1978).∗
27. Diaconis, P., Skyrms, B.: Ten Great Ideas About Chance. Princeton University Press (2018)
28. Ehrenberg, W.: Maxwell's demon. Sci. Am. **217**, 103–110 (1967).∗

29. Ekeland, I.: The Broken Dice. University of Chicago Press (1993).∗
30. Falk, R.: Judgment of coincidences, mine versus yours. Am. J. Psychol. **102**, 477–493 (1989)
31. Falk, R., Konold, C.: Making sense of randomness: implicit encoding as a basis for judgment. Psychol. Rev. **104**, 301–318 (1997)
32. Feller, W.: An Introduction to Probability Theory and its Applications, Volume 1, 2nd edn. Wiley (1957)
33. Ford, J.: How random is a coin toss? Phys. Today. **36**, 40–47 (1983).∗
34. Gawande, A.: The cancer cluster myth. The New Yorker (Feb 8, 1999)
35. Gell-Mann, M.: The Quark and the Jaguar. W.H. Freeman (1994).∗
36. Gould, S.: The evolution of life on the earth. Sci. Am. **271**, 85–91 (1994).∗
37. Gould, S.: Eight Little Piggies. Norton (1993).∗
38. Hacking, I.: The Taming of Chance. Cambridge University Press (1990).∗
39. Haddon, M.: The Curious Incident of the Dog in the Night-Time. Vintage Books (2004)
40. Johnson, G.: Fire in the Mind. Vintage Books (1996).∗
41. Johnson, G.: Of mice and elephants. New York Times (Jan 12, 1999).∗
42. Kac, M.: What is random? Am. Sci. **71**, 405–406 (1983).∗
43. Kac, M.: Probability. Sci. Am. **211**, 92–108 (1964)
44. Kahn, D.: The Code-Breakers. Macmillan (1967)
45. Kahneman, D., Tversky, A.: Subjective probability: a judgment of representativeness. Cogn. Psychol. **3**, 430–454 (1972)
46. Kahneman, D.: Thinking, Fast and Slow. Farrar, Straus and Giroux (2011)
47. Kauffman, S.: At Home in the Universe. Oxford University Press (1995).∗
48. Knuth, D.: Seminumerical Algorithms, Volume 2. Addison-Wesley (1969)

49. Li, M., Vitanyi, P.: An Introduction to Kolmogorov Complexity and its Applications, 4th edn. Springer (2019)
50. Mandelbrot, B.: The Fractal Geometry of Nature. W.H. Freeman (1977)
51. Marsaglia, G.: Random numbers fall mainly in the planes. Proc. Natl. Acad. Sci. **61**, 25–28 (1968)
52. May, R.: Simple mathematical models with very complicated dynamics. Nature. **261**, 459–467 (1976)
53. Miller, J., Sanjurjo, A.: Surprised by the hot hand fallacy? A truth in the law of small numbers. Econometrica. **86**, 2019–2047 (2018)
54. Miller, J., Sanjurjo, A.: A bridge from Monty Hall to the hot hand: the principle of restricted choice. J. Econ. Perspect. **33**, 144–162 (2019)
55. Monod, J.: Chance and Necessity. Knopf (1971)
56. Montroll, E., Schleslinger, M.: On 1/f noises and other distributions with long tails. Proc. Natl. Acad. Sci. **79**, 3380–3383 (1982)
57. Newman, J.: The World of Mathematics, 4 Volumes. Simon and Schuster (1956).∗
58. Newman, M.J.: Power laws, Pareto distributions, and Zipf's law. Contemp. Phys. **46**, 323–351 (2005)
59. Ore, O.: Cardano: The Gambling Scholar. Princeton University Press (1953).∗
60. Patch, H.: The Goddess Fortuna in Mediaeval Literature. Harvard University Press (1927).∗
61. Penrose, R.: Shadows of the Mind. Oxford University Press (1994)
62. Pierce, J.: An Introduction to Information Theory, 2nd edn. Dover Publications (1980)
63. Pincus, S., Singer, B.: Randomness and degrees of regularity. Proc. Natl. Acad. Sci. **93**, 2083–2088 (1996)
64. Pincus, S., Kalman, R.: Not all (possibly) "random" sequences are created equal. Proc. Natl. Acad. Sci. **94**, 3513–3518 (1997)
65. Prigogine, I., Stengers, I.: Order Out of Chaos. Bantam (1984).∗

66. Rosenhouse, J.: The Monty Hall Problem. Oxford University Press (2009)
67. Sayer, C.D.: Strong Poison (1930) HarperCollins Publishers
68. Shannon, C., Weaver, W.: The Mathematical Theory of Communication. University of Illinois Press (1949)
69. Stigler, S.: The History of Statistics. Harvard University Press (1986)
70. Strogatz, S.: Sync. Hyperion Books (2003)
71. Susskind, L., Friedman, A.: Quantum Mechanics. Basic Books (2014)
72. Tierney, J.: Behind Monty Hall's doors: puzzle, debate, and answer? New York Times (July 21, 1991)
73. Walker, W., Chaiken, J., Ignall, E.: Fire Department Deployment Analysis. North Holland (1979)
74. Watts, D., Strogatz, S.: Collective dynamics of "small world" networks. Nature. **393**, 440–442 (1998)
75. West, B., Shlesinger, M.: The noise in natural phenomena. Am. Sci. **78**, 40–45 (1990).∗
76. West, B., Goldberger, A.: Physiology in fractal dimensions. Am. Sci. **75**, 354–365 (1987).∗

Index

E. Beltrami, *What Is Random?*,
https://doi.org/10.1007/978-1-0716-0799-2