Proofs

1 What is a Proof?

A proof is a method of ascertaining truth. There are many ways to do this:

Jury Trial Truth is ascertained by twelve people selected at random.

Word of God Truth is ascertained by communication with God, perhaps via a third party.

Word of Boss Truth is ascertained from someone with whom it is unwise to disagree.

Experimental Science The truth is guessed and the hypothesis is confirmed or refuted by experiments.

Sampling The truth is obtained by statistical analysis of many bits of evidence. For example, public opinion is obtained by polling only a representative sample.

Inner Conviction/Mysticism "*My* program is perfect. I know this to be true."

"I don't see why not..." Claim something is true and then shift the burden of proof to anyone who disagrees with you.

"Cogito ergo sum" Proof by reasoning about undefined terms.

This Latin quote translates as "I think, therefore I am." It comes from the beginning of a famous essay by the 17th century Mathematician/Philospher, René Descartes. It may be one of the most famous quotes in the world: do a web search on the phrase and you will be flooded with hits.

Deducing your existence from the fact that you're thinking about your existence sounds like a pretty cool starting axiom. But it ain't Math. In fact, Descartes goes on shortly to conclude that there is an infinitely beneficent God.

Mathematics also has a specific notion of "proof" or way of ascertaining truth.

Definition. A *formal proof* of a *proposition* is a chain of *logical deductions* leading to the proposition from a base set of *axioms*.

The three key ideas in this definition are highlighted: proposition, logical deduction, and axiom. Each of these terms is discussed in a section below.

2 Propositions

Definition. A *proposition* is a statement that is either true or false.

This definition sounds very general, but it does exclude sentences such as, "Wherefore art thou Romeo?" and "Give me an A!".

Proposition 2.1. *2 + 3 = 5.*

This proposition is true.

Proposition 2.2. *Let* $p(n) ::= n^2 + n + 41$.

$$\forall n \in \mathbb{N}\ p(n) \textit{ is a prime number}.$$

The symbol $\forall$ is read "for all". The symbol $\mathbb{N}$ stands for the set of *natural numbers*, which are 0, 1, 2, 3, . . . ; (ask your TA for the complete list). A *prime* is a natural number greater than one that is not divisible by any other natural number other than 1 and itself, for example, 2, 3, 5, 7, 11, ...

Let's try some numerical experimentation to check this proposition: $p(0) = 41$ which is prime. $p(1) = 43$ which is prime. $p(2) = 47$ which is prime. $p(3) = 53$ which is prime. . . . $p(20) = 461$ which is prime. Hmmm, starts to look like a plausible claim. In fact we can keep checking through $n = 39$ and confirm that $p(39) = 1601$ is prime.

But if $n = 40$, then $p(n) = 40^2 + 40 + 41 = 41 \cdot 41$, which is not prime. Since the expression is not prime *for all* n, the proposition is false! In fact, it's not hard to show that *no* nonconstant polynomial can map all natural numbers into prime numbers. The point is in general you can't check a claim about an infinite set by checking a finite set of its elements, no matter how large the finite set. Here are two even more extreme examples:

Proposition 2.3. $a^4 + b^4 + c^4 = d^4$ *has no solution when* a, b, c, d *are positive integers. In logical notation, letting* $\mathbb{Z}^+$ *denote the positive integers, we have*

$$\forall a \in \mathbb{Z}^+ \forall b \in \mathbb{Z}^+ \forall c \in \mathbb{Z}^+ \forall d \in \mathbb{Z}^+\ a^4 + b^4 + c^4 \neq d^4.$$

Strings of $\forall$'s like this are usually abbreviated for easier reading:

$$\forall a, b, c, d \in \mathbb{Z}^+\ a^4 + b^4 + c^4 \neq d^4.$$

Euler (pronounced "oiler") conjectured this 1769. But the proposition was proven false 218 years later by Noam Elkies at the liberal arts school up Mass Ave. He found the solution $a = 95800, b = 217519, c = 414560, d = 422481$.

Proposition 2.4. $313(x^3 + y^3) = z^3$ *has no solution when* $x, y, z \in \mathbb{N}$.

This proposition is also false, but the smallest counterexample has more than 1000 digits!

Proposition 2.5. *Every map can be colored with 4 colors so that adjacent*[1] *regions have different colors.*

[1] Two regions are adjacent only when they share a boundary segment of positive length. They are not considered to be adjacent if their boundaries meet only at a few points.

This proposition is true and is known as the "four-color theorem". However, there have been many incorrect proofs, including one that stood for 10 years in the late 19th century before the mistake was found. An extremely laborious proof was finally found about 15 years ago by a Mathematician named Haaken who used a complex computer program to categorize maps as four-colorable; the program left a couple of thousand maps uncategorized, and these were checked by hand by Haaken and his assistants—including his 15-year-old daughter. There was a lot of debate about whether this was a legitimate proof: the argument was too big to be checked without a computer, and no one could guarantee that the computer calculated correctly, nor did anyone have the energy to recheck the four-colorings of thousands of maps that was done by hand. Finally, about five years ago, a humanly intelligible proof of the four color theorem was found (see http://www.math.gatech.edu/ thomas/FC/fourcolor.html).

Proposition 2.6. *The original Pentium chip divided properly.*

Intel's "proofs" by authority and by sampling turned out to be invalid. The proposition is false.

Proposition 2.7 (Goldbach). *Every even integer greater than 2 is the sum of two primes.*

No one knows whether this proposition is true or false. This is the "Goldbach Conjecture," which dates back to 1742.

3 Axioms

Definition. An *axiom* is a proposition that is assumed to be true.

There is no proof that an axiom is true; you just assume it is true because you believe it is reasonable. Here are some examples:

Axiom 3.1. If $a = b$ and $b = c$, then $a = c$.

This seems very reasonable! But sometimes the right choice of axiom is not clear.

Axiom 3.2 (Euclidean geometry). Given a line l and a point p not on l, there is exactly one line through p parallel to l.

Axiom 3.3 (Spherical geometry). Given a line l and a point p not on l, there is *no* line through p parallel to l.

Axiom 3.4 (Hyperbolic geometry). Given a line l and a point p not on l, there are *infinitely many* lines through p parallel to l.

No one of the three preceding axioms is better than the others; all yield equally good proofs. Of course, a different choice of axioms makes different propositions true. Still, a set of axioms should not be chosen arbitrarily. In particular, there are two basic properties that one would want in any set of axioms; it should be consistent and complete.

Definition. A set of axioms is *consistent* if no proposition can be proven to be both true and false.

This is an absolute must. One would not want to spend years proving a proposition true only to have it proven false the next day! Proofs would become meaningless if axioms were inconsistent.

Definition. A set of axioms is *complete* if it can be used to prove or disprove every proposition.

Completeness is an attractive property; we would like to believe that any proposition could be proven or disproven with sufficient work and insight.

Surprisingly, making a complete, consistent set of axioms is not easy. Bertrand Russell and Alfred Whitehead tried during their entire careers to find such axioms for basic arithmetic and failed. Then Kurt Gödel proved that no set of axioms can be both consistent and complete! This means that any set of consistent axioms (an absolute must) can not be complete; there will be true statements that can not be proven. For example, it might be that Goldbach's conjecture is true, but there is no proof!

In 6.042 we will not worry about the precise set of axioms underpinning our proofs. The requirements are only that you be upfront about what you are assuming, that the background knowledge of Math that you assume is self-consistent, and that you do not try to avoid homework and exam problems by declaring everything an axiom!

4 Logical Deductions

Logical deductions or *inference rules* are used to combine axioms and true propositions to construct more true propositions.

A fundamental inference rule is *modus ponens*. This rule says that if p is true and $p \longrightarrow q$ is true, then q is true. The expression $p \longrightarrow q$ is read "p implies q" or "if p, then q." A truth table for $\longrightarrow$ is shown below:

p	q	$p \to q$
T	T	T
T	F	F
F	T	T
F	F	T

Inference rules are sometimes written in a funny notation. For example, *modus ponens* is written:

Rule.

$$\frac{p, \quad p \longrightarrow q}{q}$$

When the statements above the line, called the *antecedents*, are true, then we can infer that the statement below the line, called the *conclusion* or the *consequent*, is also true. There are many other natural inference rules, for example:

Rule.

$$\frac{p \longrightarrow q, \quad q \longrightarrow r}{p \longrightarrow r}$$

Rule.

$$\frac{p \longrightarrow q, \quad \neg q}{\neg p}$$

Rosen describes additional standardized inference rules useful in proofs. As with axioms, we will not be too formal about the set of legal inference rules. Each step in a proof should be clear and "logical"; in particular, you should state what previously proved facts are used to derive each new conclusion.

5 Good Proofs and Bad Proofs

An estimated 1/3 of all mathematical papers contain errors. Even some of the world's most famous mathematicians have botched proofs. Here are some famous examples.

- Andrew Wiles recently announced a proof of Fermat's Last Theorem. It was several hundred pages long. It took mathematicians months of hard work to discover it had a fatal flaw (so Wiles produced another proof of several hundred pages; this one seems to have convinced people).
- Gauss's 1799 Ph.D. thesis is usually referred to as being the first rigorous proof of the Fundamental Theorem of Algebra (every polynomial has a zero over the complex numbers). But it contains quotes like

  > "If a branch of an algebraic curve enters a bounded region, it must necessarily leave it again. ... Nobody, to my knowledge, has ever doubted [this fact]. But if anybody desires it, then on another occasion I intend to give a demonstration which will leave no doubt."

  Fields Medalist Steve Smale writes about this, calling it an "immense gap" in the proof that was not filled in until 1920, more than a hundred years later.
- In 1900 Poincare carelessly claimed a certain very simple topological characterization of the 3-dimensional sphere. Later realizing it was not so obvious, he demoted the claim to the status of a "conjecture" in 1904. The Poincare Conjecture is now one of the biggest open questions in mathematics (two Fields Medals have been given out for partial progress on it).

Here are some of the characteristics of a good proof:

- It is clear and *correct*!
- It has a nice structure, like a good program. It is broken up into separate parts that define and prove key intermediate properties. This makes it easy to understand the reason the whole thing works. It also makes it more likely that pieces can be reused.
- The pieces are general and abstract. This avoids the clutter of unnecessary hypotheses, useless restrictions, etc. Again, the analogy to programming holds; a subroutine should be as generally applicable as possible.

- Important conclusions are not "justified" by being "left to the reader," nor by intimidating phrases like "it is obvious that . . . " or "any moron can see that . . . ." These phrases save the writer's time, but consume the reader's time. Mistakes in proofs are typically found in these parts "left to the reader."

- Like a scientific experiment, someone else must be able to "replicate" (i.e. understand) your proof.

Proofs are important. They permit you to convince yourself and others that your reasoning is correct. The insights gained can help you understand why something is true and whether it will stay true when other things change. Proofs are particularly important in computer science and electrical engineering. Bugs have proven costly for Intel, AT&T, and Airbus. A good proof is strong evidence that no bugs exist.

Massachusetts Institute of Technology
6.042J/18.062J, Fall '02: Mathematics for Computer Science
Professor Albert Meyer and Dr. Radhika Nagpal

Induction

1 Proof by Induction

1.1 The Induction Axiom

Induction is by far the most powerful and commonly-used proof technique in Discrete Mathematics and Computer Science. In fact, one could say that applicabillity of induction is the defining characteristic of *discrete*, as opposed to *continuous*, Mathematics.

The standard formulation of induction involves proving properties of the natural numbers, $\mathbb{N} ::= 0, 1, 2, \ldots$. But since most objects of interest in Computer Science—computer programs, task schedules, game outcomes, steps in a computation—can be numbered[1], induction applies widely.

Induction captures a style of reasoning which is so obvious and familiar that its use often goes unnoticed. For example, suppose we had some recipe for assigning a unique "color" to every natural number. One recipe might, for example, assign red to even numbers and blue to odd numbers. Another recipe would be to color even numbers red, odd prime numbers blue, and all other numbers white. Now suppose someone formulates a recipe for natural number coloring, but doesn't tell you exactly what the recipe is. But they do tell you that zero is colored red, and that the coloring has the property that, whenever some number is red, then the next number is red. Can there be any doubt about what the unknown coloring is? Of course not: *every* number is colored red!

The Axiom of Induction essentially just this: if zero is red, and the next number after a red number is also red, then all numbers are red. So the Induction Axiom is both simple and obvious. What's not so obvious is how much mileage we get by using it. For example, let's prove by induction that

$$1 + 2 + \cdots + n + (n + 1) = \frac{(n + 2)(n + 1)}{2}, \tag{1}$$

for all $n \in \mathbb{N}$. The trick for applying Induction is to use this equation for assigning colors to numbers: color the number n red when equation (1) holds, otherwise color it white. To verify that equation (1) holds for all $n \in \mathbb{N}$, we must show that every number is red. Induction allows us to prove this using simple arithmetic.

To begin with, we have to show that zero is red. In other words, we have to show that zero satisfies equation (1). Now when $n = 0$, the lefthand side of the equation is simply 1 and the righthand side is $(0 + 2)(0 + 1)/2$, which equals 1. So zero is red.

[1] A variant of induction, called *structural induction*, is specially tailored for proof about recursively defined data structures and processes; structural induction will be discussed in later notes.

Next, we suppose we have arrived at some natural number, m, which is colored red. We only have to show that the next number, $m+1$, must also be red. Then by Induction all natural numbers are red. That is, equation (1) holds for all $n \in \mathbb{N}$.

Now in this case, saying that m is red means

$$1+2+\cdots+m+(m+1)=\frac{(m+2)(m+1)}{2}. \tag{2}$$

This is called the *induction hypothesis*.

How do we show the next number, $m+1$, is red? We have to show:

$$1+2+\cdots+(m+1)+((m+1)+1)=\frac{((m+1)+2)((m+1)+1)}{2}. \tag{3}$$

But that's easy using the redness of m and rules of arithmetic:

$$\begin{aligned}
1+2+\cdots+(m+1)+((m+1)+1) =&[1+2+\cdots+(m+1)]+(m+2) \\
=&[1+2+\cdots+(m+1)]+(m+2) \quad &\text{(associativity of +)} \\
=&\frac{(m+2)(m+1)}{2}+(m+2) \quad &\text{(by (2))} \\
=&(\frac{m+1}{2}+1)(m+2) \\
=&(\frac{m+1}{2}+\frac{2}{2})(m+2) \\
=&\frac{(m+1)+2}{2}(m+2) \\
=&\frac{((m+1)+2)((m+1)+1)}{2}.
\end{aligned}$$

Here associativity for sums tells us it's ok to parenthesize the sum in any convenient way, and the unlabelled equalities each follow by simple arithmetic. So we have finished the proof by induction that (2)

The Induction Axiom is usually stated formally using logical formulas. To begin with, let's consider some fixed coloring, and interpret the predicate $P(n)$ to mean that "n is colored red." Then we translate our informal language in logical formulas as follows: "we have a coloring that makes zero red" simply translates into $P(0)$. The clause "whenever some number is red, then the next number is red," translates first into "whenever some number, call it, m, satisfies $P(m)$, then $P(m+1)$." We can translate the "whenever some number m" phrase into a universal quantifier and the "if ... then" into $\longrightarrow$, so the whole phrase translates into

$$\forall m \in \mathbb{N}\; P(m) \longrightarrow P(m+1).$$

The conclusion that "every number is colored red" translates into $\forall n\; P(n)$. So now we can formally state the

Axiom (Induction). Suppose that $P(0)$ is true and

$$\forall m \in \mathbb{N}\; P(m) \longrightarrow P(m+1).$$

Then $\forall n \in \mathbb{N}\; P(n)$ is true.

In fact, we can get rid of the English altogether and formulate

Rule 1.1 (Induction).

$$\frac{P(0), \quad \forall m \in \mathbb{N}\, P(m) \longrightarrow P(m+1)}{\forall n \in \mathbb{N}\, P(n).}$$

We saved this last formulation to last because, until you're experienced translating logical formulas into intelligible language, the formula can hide how obvious and simple the Induction Axiom really is. Actually, you'll often see the Induction Axiom and Rule stated with n in place of m, which can make them even harder to decipher. But since m is a bound variable in the second hypothesis of the rule, it doesn't matter if we rename it to be n.

1.2 Ellipses

Incidentally, the argument above could be criticized because notation such as $1 + 2 + 3 + \cdots + n$ may seem imprecise. Alway watch out for notation with "$\cdots$" or " $\ldots$ " in it (the dots are called an "ellipsis"). This notation is common because it is convenient. The idea is to show enough of a sequence that anyone can figure out the pattern needed to fill in the ellipsis. We could have been more precise by using summation notation instead, namely, $1 + 2 + 3 + \cdots + n$ could be written either as

$$\sum_{i=1}^{n} i$$

or as

$$\sum_{1 \le i \le n} i.$$

In this notation, the pattern of terms in the summation is made explicit. In two important special cases, the definition of the summation $1 + 2 + 3 + \cdots + n$ requires some care. We already observed that if $n = 1$, then $1 + 2 + 3 + \cdots + n = \sum_{1 \le i \le 1} i = 1$. That is, There is only one term in the summation; the appearance of 2 and 3 to indicate the pattern is misleading in this case, because they don't appear.

What about when $n = 0$? Then $\sum_{1 \le i \le 0} i$ is a sum over an *empty set* of i's. That is, there are no terms at all in the summation. In this case, the sum is *defined* to be zero by convention. This convention is useful, because, for example, we can say that for any function $f : \mathbb{N} \to \mathbb{R}$,

$$\sum_{1 \le i \le n+1} f(i) = \left(\sum_{1 \le i \le n} f(i) \right) + f(n+1)$$

for all $n \in \mathbb{N}$, even for $n = 0$.

1.3 Proof Format

The text of a proof by induction should consist of four parts. We've aleady seen each of these parts in the proof of equation (1).

1. **State that the proof is by induction.** This immediately conveys the general structure of the argument.

2. **Specify the induction hypothesis**: $P(n)$. Sometimes, the choice of $P(n)$ will come directly from the theorem statement. In the proof above, $P(n)$ was the equation (1) to be proved. Other times, the choice of $P(n)$ is not obvious at all; we will see an example of this soon.

3. **The basis step**: prove $P(0)$. The "basis step" or "base case" is a proof of the predicate $P(0)$.

4. **The inductive step**: prove that $\forall m \in \mathbb{N}\ P(m) \longrightarrow P(m+1)$. Begin the inductive step by writing, "For $m \geq 0$, assume $P(m)$ in order to prove $P(m+1)$." (You can substitute in the statements of the predicates $P(m)$ and $P(m+1)$ if the reminder seems helpful.) Then verify that $P(m)$ indeed implies $P(m+1)$ for every $m \in \mathbb{N}$.

In the case of equation (1), we used induction purely as a proof technique; it gave little insight into why the theorem is true.

Furthermore, while induction was essential in proving the summation equal to $n(n+1)/2$, it did not help us find this formula in the first place. We'll turn to the problem of finding sums of series in a couple weeks.

1.4 Induction Examples

This section contains several examples of induction proofs. We begin with an example about Fibonacci numbers, followed by an example from elementary plane geometry, and finally an application of induction to a design problem vital to the future of Computer Science at MIT. Then we illustrate some typical mistakes in using induction by proving (incorrectly!) that all horses are the same color and that camels can carry an unlimited amount of straw.

1.4.1 A Fibonacci Identity

Fibonacci was a thirteenth century mathematician who invented *Fibonacci numbers* to model population growth (or rabbits, see Rosen, pp. 205, 310). The first two Fibonacci numbers are 0 and 1, and each subsequent Fibonacci number is the sum of the two previous ones. The n Fibonacci numbers is denoted F_n. In other words, the Fibonacci numbers are defined defined recursively by the rules

$$\begin{aligned} F_0 &::= 0, \\ F_1 &::= 1, \\ F_i &::= F_{i-1} + F_{i-2}, \text{ for } i \geq 2. \end{aligned}$$

Here, we're using the notation "::=" to indicate that an equality holds *by definition*. The first few Fibonacci numbers are

$$0, 1, 1, 2, 3, 5, 8, 13, 21, \ldots$$

Fibonacci numbers come up in several different settings, but they have captivated a continued mathematical following out of proportion to their importance in applications because they have a rich and surprising collection of properties, such as the one expressed in the following theorem. The theorem is a good thing to forget if you run low on brain space, its proof just provides a nice illustration of induction.

Theorem 1.2. $\forall n \geq 1, F_1^2 + F_2^2 + \cdots + F_n^2 = F_n F_{n+1}$

For example, for $n = 4$ we have $1^2 + 1^2 + 2^2 + 3^2 = 15 = 3 \cdot 5$.

Let's look for a proof by induction. First, the theorem statement suggests that the induction hypothesis $P(n)$ be

$$P(n) ::= \left[\sum_{i=1}^{n} F_i^2 = F_n F_{n+1}\right].$$

.

Second, we want to identify the gap between $P(m)$ and $P(m+1)$. The predicate $P(m+1)$ states that $\sum_{i=1}^{m+1} F_i^2 = F_{m+1}F_{m+2}$. Now the plan is to use $P(m)$ to reduce this statement to a simpler assertion. An easy way is to subtract the equation in predicate $P(m)$. Taking the $P(m+1)$ equation "minus" $P(m)$ equation gives:

$$F_{m+1}^2 = F_{m+1}F_{m+2} - F_m F_{m+1}.$$

This is the Fibonacci recurrence in disguise; dividing by F_{m+1} and moving a term gives $F_m + F_{m+1} = F_{m+2}$. This is the extra fact need to bridge the gap between $P(m)$ and $P(m+1)$ in the inductive step. The full proof is written below.

Proof. The proof is by induction. Let $P(n)$ be the proposition that $\sum_{i=1}^{n} F_i^2 = F_n F_{n+1}$. In the base case, $P(0)$ is true because $0 = F_0 F_1 = 0 \cdot 1 = 0$. For $m \geq 0$, assume $\sum_{i=1}^{m} F_i^2 = F_m F_{m+1}$ to prove $\sum_{i=1}^{m+1} F_i^2 = F_{m+1}F_{m+2}$.

For all $m \geq 0$, the equation $F_m + F_{m+1} = F_{m+2}$ holds by the definition of the Fibonacci numbers. Multiplying both sides by F_{m+1} and rearranging terms gives $F_{m+1}^2 = F_{m+1}F_{m+2} - F_m F_{m+1}$. Adding this identity to the equation in the proposition $P(m)$ gives:

$$\begin{aligned} F_{m+1}^2 + \sum_{i=1}^{m} F_i^2 &= (F_{m+1}F_{m+2} - F_m F_{m+1}) + F_m F_{m+1} \\ \sum_{i=1}^{m+1} F_i^2 &= F_{m+1}F_{m+2} \end{aligned}$$

This proves that for all $m \in \mathbb{N}$, $P(m) \longrightarrow P(m+1)$ and completes the proof. □

1.4.2 Geometry

Definition 1.3. A convex polygon is a polygon such that any straight line between any two vertices doesn't leave the polygon.

Theorem. *The sum of the interior angles in any n-sided convex polygon is exactly $(n-2) \cdot 180$ degrees, for all $n \geq 3$.*

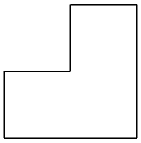


Figure 1: One of the L-shaped tiles that will be used in the courtyard of the new computer science building.

Proof. The proof is by induction. The induction hypothesis is $P(n)$::= The sum of the interior angles in any n-sided convex polygon is exactly $(n-2)\cdot 180$ degrees.

Base case $n = 3$: An 3-sided polygon is a triangle, whose interior angles were shown always to sum to 180 degrees by Euclid.

Inductive step: Assume that $P(m)$ holds for some $m \geq 3$. We must show that $P(m+1)$ holds.

So let X be any $(m+1)$-vertex convex polygon, say with successive vertices $x_1, x_2, \ldots, x_{m+1}$. Let Y be the polygon with vertices $x_1, x_2, \ldots, x_m$. That is, Y is obtained by cutting out one vertex from X. Now Y is also a convex polygon (proof left to the reader!), so by induction hypothesis $P(m)$, the sum of the interior angles of Y is $(m-2)180$. Now let T be the triangle with vertices x_m, x_{m+1}, x_1. The sum of the interior angles in X is the sum of those in Y plus the sum of those in T (proof again left to the reader: draw a picture [2]). So the sum of the interior angles in X is $(m-2)180+180 = ((m+1)-2)180$. Since X was arbitrary, we conclude that the sum of the interior angles of any $(m+1)$-sided convex polygon is $((m-2)+1)180$. That is, $P(m+1)$ holds. □

Note that this induction argument started with base case $n = 3$ rather than 0. The induction step proved that $P(m) \longrightarrow P(m+1)$ for all $m \geq 3$. The final conclusion was that $\forall n \geq 3\ P(n)$. This is a valid variant of induction.

1.4.3 The Fate of Computer Science at MIT

In the preceding examples, induction has served purely as a proof technique. However, it can be useful more generally in problem solving.

MIT is constructing a new Stata Center on the site of the old Building 20. Designed by the world famous architect Frank Gehry, the current cost of the project is budgeted at around $200 million. The Center includes two Computer Science Buildings, one of which is already named after Bill Gates in recognition of his $20 million donation toward construction. But the budget has grown enormously—it was originally supposed to be $100 million. Despite the dramatic recent declines in the stock market, Bill can still afford to make another contribution to cover the shortfall[3], but it will take some special enticement.

Gehry has designed an atrium with a spacious central plaza to be tiled in L-shaped tiles, and MIT is thinking about offering to place a statue of Bill in the courtyard.

The planned courtyard consists of $2^n \times 2^n$ squares. Most of these will be covered by L-shaped tiles, each covering three squares as shown in Figure 1. However, one square will be covered by

[2] see Velleman, example 6.2.3

[3] Up to this point, the story is all true.

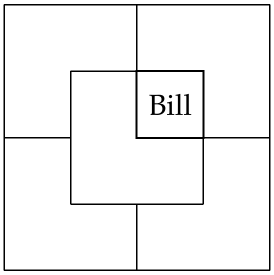


Figure 2: Example with $n = 2$: a legal tiling of a 4x4 courtyard.

the statue of Bill; in fact, this should be one of the central squares. The problem is to find a suitable tiling. An example solution for the case of $n = 2$ is shown.

(The phrase "central squares" is a little ambiguous. If $n = 0$, then the courtyard is a single square, and Bill takes it. If $n > 0$, then there are four central squares, and Bill will take any of them.)

Let's try to prove by induction that such a tiling exists. As usual, we first try to lift the inductive hypothesis directly from the theorem statement.

Theorem 1.4. *For all $n \geq 0$ there exists a tiling of a $2^n \times 2^n$ courtyard with Bill in a central square.*

Proof. (doomed attempt) The proof is by induction. Let $P(n)$ be the proposition that there exists a tiling of a $2^n \times 2^n$ courtyard with Bill in the center. In the base case, $P(0)$ is true because Bill fills the whole courtyard. For $n \geq 0$, assume that there is a tiling of a $2^n \times 2^n$ courtyard with Bill in the center to prove that there is is a legal tiling of a $2^{n+1} \times 2^{n+1}$ courtyard with Bill in the center... □

Now we're in trouble! The ability to tile a smaller courtyard with Bill in the center is of no obvious help in tiling a larger courtyard with Bill in the center. The usual recipe for finding an inductive proof will not work!

Sometimes, making the induction hypothesis *stronger* makes a proof *easier*. For example, we could make $P(n)$ the proposition that for every position of Bill in a $2^n \times 2^n$ courtyard, there exists a tiling of the remainder. This hypothesis is "stronger" in the sense that the earlier claim was just a special case. However, when we have to prove $P(n) \longrightarrow P(n+1)$, we will be in better shape because we can *assume* $P(n)$, which is now a more general, more useful statement.

Method 1. If you can not show that $P(n) \longrightarrow P(n+1)$ in a proof by induction, change the induction hypothesis; in particular, strengthening the hypothesis may make the proof easier.

Even with this new hypothesis, finding the right way to prove that $P(n) \longrightarrow P(n+1)$ requires some work.

Proof. (successful attempt) The proof is by induction. Let $P(n)$ be the proposition that if any one square of a $2^n \times 2^n$ courtyard must be left blank, then there exists a tiling of the remainder. In the base case, $P(0)$ is true because if the one and only square is left blank, then there exists a tiling of the remainder (which is nothing). For $n \geq 0$, assume that if any one square of a $2^n \times 2^n$ courtyard must be left blank, then there exists a tiling of the remainder. We will use this to prove that if any one square of a $2^{n+1} \times 2^{n+1}$ courtyard must be left blank, then there exists a tiling of the remainder.

Divide the $2^{n+1} \times 2^{n+1}$ courtyard into four quadrants, each $2^n \times 2^n$. One will contain the square that must be left blank and can be tiled by induction. Now place a tile in the center of the courtyard so that it covers one square in each remaining quadrant. All that remains is to tile each of these three quadrants, excluding the one square in each that is already covered. But this can also be done by induction. This proves that $\forall n \geq 1\ P(n) \longrightarrow P(n+1)$. The theorem follows as a special case in which a central square is left blank during tiling and is later covered by a statue of Bill. □

This proof has two nice properties. First, we have a stronger result; if Bill wants his statue on the edge of the courtyard, away from the pigeons, we can accommodate him. Second, not only does the proof guarantee that a tiling exists, it actually gave a *recursive procedure* for producing one. For example: To tile a $2^3 \times 2^3$ square leaving the upper right corner empty, divide it into 4, put one tile in the center, and recursively tile the 4 pieces, each with one square missing. (See Figure 3)

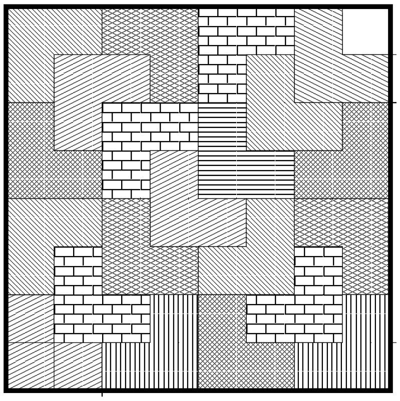


Figure 3: A valid tiling for an 8x8 square leaving the upper right corner empty

1.4.4 A False Proof

False Theorem 1.5. *All horses are the same color.*

Proof. The proof is by induction. Let $P(n)$ be the proposition that in any set of n horses, all the horses are the same color. This is true in the base case $n = 1$, since there is only one horse in the set. For $n \geq 1$, assume that in every set of n horses, all are the same color in order to prove that in every set of $n+1$ horses, all are the same color. Consider a set of $n+1$ horses $h_1, h_2, \ldots, h_{n+1}$. By induction, $h_1, h_2, \ldots, h_n$ all are the same color. Likewise, $h_2, \ldots, h_{n+1}$ all are the same color. Therefore, $h_1, h_2, \ldots, h_{n+1}$ must all share the same color, namely the color of h_2. This proves that $P(n) \longrightarrow P(n+1)$ for any n, and so completes the proof. □

Where is the bug?—it's in the sentence beginning "Therefore." The " ... " notation helps create confusion about an implicit assumption that the sets $\{h_1, h_2, \ldots, h_n\}$ and $\{h_2, \ldots, h_{n+1}\}$ overlap at h_2, and therefore are colored the same. But if $n = 1$, then the first set is just $\{h_1\}$ and the second is $\{h_2\}$, and they do not overlap at all.

Because of this bug, we have really only proven $P(1)$, and $P(n) \longrightarrow P(n+1)$ for $n \geq 2$. But we haven't proved that $P(1) \longrightarrow P(2)$, which of course does not hold.

1.4.5 Another False Proof

False Theorem 1.6. *A camel can always carry all the straw in a barn.*

Proof. The proof is by induction. Let $P(n)$ be the predicate, "The camel can carry n pieces of straw." The base case $P(1)$ is true because a camel can certainly carry one piece of straw. In the inductive step, assume that the camel can carry n pieces of straw to prove that it can carry $n + 1$ pieces. But if it can carry n pieces of straw, then surely it can carry $n + 1$: one little piece of straw won't make any difference. Therefore $P(n) \longrightarrow P(n + 1)$, completing the proof. □

The flaw here is in the bogus assertion that the camel can carry $n + 1$ straws if it can carry n. Just because it is hard to say exactly for which n this is false, we have no doubt that there is an n that finally exceeds the camel's carrying ability. There will always be "a straw that broke the camel's back."

2 Strong Induction

"Strong" induction[4] is a variation of the induction proof method. Strong induction is quite similar to ordinary induction, but is sometimes easier to use when solving problems.

The difference between ordinary induction and strong induction is subtle. Both proofs can be written with nearly the same structure. The only difference is that in an ordinary induction proof we assume only $P(n)$ in order to prove $P(n+1)$. In a strong induction proof, we get to assume all of $P(0), P(1), \ldots, P(n)$ in order to prove $P(n + 1)$. This can be a big help. When we try to prove $P(n+1)$ in the inductive step, we do not have just one fact in hand, but rather a whole list of facts!

2.1 The Strong Induction Axiom

Like ordinary induction, strong induction can be expressed as an axiom:

Axiom (Strong Induction). If $P(0)$ is true and $\forall n \geq 0\ (P(0) \wedge P(1) \wedge \cdots \wedge P(n)) \longrightarrow P(n + 1)$, then $P(n)$ is true for all $n \geq 0$.

The expression $(P(0) \wedge P(1) \wedge \cdots \wedge P(n)) \longrightarrow P(n + 1)$ might be a little hard to decrypt. It just means that $P(n + 1)$ logically follows if we accept all the statements $P(0), P(1), \ldots, P(n)$. Writing this as a rule with logical formulas makes this explicit

Rule 2.1. *[Strong Induction]*

$$\frac{P(0), \quad \forall n \in \mathbb{N}\, \forall m \leq n\ P(m) \longrightarrow P(m+1)}{\forall n \in \mathbb{N}\ P(n)}$$

[4] Strong Induction is the same as what Rosen calls the *Second Principle of Induction*

Strong induction is as obvious a principle as ordinary induction, so we could confidently take it as another axiom. Actually, we don't have to make it an axiom, because we could prove the correctness of strong induction by very elementary reasoning starting from the induction axiom.

There's also another interesting way to justify strong induction without using ordinary induction at all. The proof is by contradiction: suppose that some statement in the list $P(0), P(1), \dots, P(n), \dots$ was actually false. Since there's some false statement in the list, there must be a *first* one, say $P(k)$ for some $k > 0$, that is false. (The number k has to be > 0 because we know $P(0)$ is true.) Now we know that $P(0), P(1), \dots, P(k-1)$ are true, since $P(k)$ is the first false statement. But since $P(k)$ logically followed from the preceding statements $P(0), P(1), \dots, P(k-1)$, it must be true, contradicting our assumption that it was false. So there can't be any false statement in the list, that is, $P(n)$ is true for all $n \in \mathbb{N}$.

Of course, we do not prove axioms; we just accept them as facts. But in this case we didn't need to *assume* a strong induction axiom, because we were able to prove the correctness of strong induction, and we did it without even using induction! How come? Well, if you look back at the previous argument, you can see we made a key assumption: that there exists a *first* false statement, $P(k)$. This assumption is an instance of another axiom called the *Least Number Principle* which says that in any set of one or more natural numbers, there must be a *least* (smallest) number. So we have proved the soundness of strong induction, and could similarly prove the soundness of ordinary induction too, by elementary reasoning from the Least Number Principle. This may help you think more clearly about why induction works.

2.2 Postage Stamp Example

Now we're ready to solve a problem using strong induction.

Problem: Given an unlimited supply of 3 cent and 5 cent stamps, what postages are possible?

Solution: Let's first try to guess the answer and then try to prove it. A table that shows the values of all possible combinations of 3 and 5 cent stamps will help. The column heading is the number of 5 cent stamps and the row heading is the number of 3 cent stamps.

	0	1	2	3	4	5	...
0	0	5	10	15	20	25	...
1	3	8	13	18	23	...	
2	6	11	16	21	...		
3	9	14	19	24	...		
4	12	17	22	...			
5	15	20	...				
...	...	...					

Looking at the table, a reasonable guess is that the possible postages are 0, 3, 5, and 6 cents and every value of 8 or more cents. Let's try to prove this last part using strong induction.

Claim 2.2. *For all $n \geq 8$, it is possible to produce n cents of postage from 3¢ and 5¢ stamps.*

Now let's preview the proof. The induction hypothesis will be

$$P(n)::=\text{if } n \geq 8,\ \text{then } n\text{¢ postage can be produced using 3¢ and 5¢ stamps} \tag{4}$$

A proof by strong induction will have the same four-part structure as an ordinary induction proof. The base case, $P(0)$, won't be interesting because $P(n)$ is *vacuously* true for all $n < 8$.

In the inductive step we have to show how to produce $n+1$ cents of postage, assuming the strong induction hypothesis that we know how to produce k¢ of postage for all values of k between 8 and n. A simple way to do this is to let $k = n - 2$ and produce k¢ of postage; then add a 3¢ stamp to get $n + 1$ cents.

But we have to be careful; there is a pitfall in this method. If $n + 1$ is 8, 9 or 10, then we can not use the trick of creating $n + 1$ cents of postage from $n - 2$ cents and a 3 cent stamp. In these cases, $n - 2$ is less than 8. None of the strong induction assumptions help us make less than 8¢ postage. Fortunately, making $n + 1$ cents of postage in these three cases can be easily be done directly.

Proof. The proof is by strong induction. The induction hypothesis, $P(n)$, is given by (4).

Base case ($n = 0$): $P(0)$ is true vacuously.

In the inductive step, we assume that it is possible to produce postage worth $8, 9, \ldots, n$ cents in order to prove that it is possible to produce postage worth $n + 1$ cents.

There are four cases:

1. $n + 1 < 8$: So $P(n + 1)$ holds vacuously.
2. $n + 1 = 8$: $P(n + 1)$ holds because we produce 8¢postage using one 3¢and one 5¢stamp.
3. $n + 1 = 9$: $P(n + 1)$ holds by using three 3¢stamps.
4. $n + 1 = 10$: $P(n + 1)$ holds by using two 5¢stamps.
5. $n + 1 > 10$: We have $n \geq 10$, so $n - 2 \geq 8$ and by strong induction we may assume we can produce exactly $n - 2$ cents of postage.

So in every case, $P(0) \wedge P(1) \wedge \ldots P(n) \longrightarrow P(n + 1)$. By strong induction, we have conclude that $P(n)$ is true for all $n \in \mathbb{N}$. □

2.2.1 Induction with nonzero base cases

To conform to the standard format, we organized the proof of Claim 2.2 with a base case of 0. But since we only were interested in 8 or more cents postage, it would have made more sense to start the induction at 8 instead of 0, to treat 8, 9 and 10 as *three* base cases, and to consider the induction step only for $n + 1 > 10$. From now on, we will allow induction proofs formatted with several base cases in this way.

At the other extreme, we can formulate strong induction with no base case at all—just an induction step. Namely, we could replace the strong induction Rule 2.1 with another logical rule:

Rule 2.3. *[Strong Induction without base case]*

$$\frac{\forall n \in \mathbb{N}\ (\forall m < n\ P(m)) \longrightarrow P(n))}{\forall n \in \mathbb{N}\ P(n)}$$

Notice that the base case antecedent, $P(0)$, is missing from Rule 2.3. That's because it's hidden in the single, "induction-step" antecedent of the rule. Namely, when $n = 0$ the antecedent requires that $P(0)$ holds as long as $P(m)$ holds for all natural numbers $m < 0$. But we can say that $P(m)$ *does* hold for *all* such natural numbers $m < 0$ since there aren't any! In practice, using this form of strong induction means that even though the proof has no base case, doing the induction step requires handling $n = 0$ as a separate case.

2.3 Strong Induction False Proof

In the preceding proof, we were careful not to accidently assume more than is permitted by the strong induction axiom. Now let's be sloppy and see what fun facts we can prove!

False Theorem 2.4. *All Fibonacci numbers are even.*

Remember that the Fibonacci numbers are denoted by $F_0, F_1, F_2, \ldots$ where $F_0 = 0$, $F_1 = 1$, and $F_i = F_{i-1} + F_{i-2}$ for $i \geq 2$. The first few Fibonacci numbers are $0, 1, 1, 2, 3, 5, 8, 13, \ldots$.

Proof. The proof is by strong induction. Let $P(n)$ be the predicate that F_n is even. In the base case, $P(0)$ is true because $F_0 = 0$, which is even. In the inductive step, for $n \geq 0$ assume that $F_0, F_1, \ldots, F_n$ are all even in order to prove that F_{n+1} is even. By definition, $F_{n+1} = F_n + F_{n-1}$. Since both F_n and F_{n-1} are even, F_{n+1} is even. □

Where is the bug? If $n = 0$, then the statement "By definition, $F_{n+1} = F_n + F_{n-1}$" is false. In this case, $F_{n+1} = F_1$, which equals 1 by definition. We forgot a special case!

We really only proved $P(0)$ and $P(0) \wedge P(1) \longrightarrow P(2), P(1) \wedge P(2) \longrightarrow P(3), \ldots$. We forgot to check one little thing, $P(1)$, and reached an infinite number of false conclusions!

2.4 Winning the Game of Nim

The game of Nim is defined as follows: Some positive number of sticks are placed on the ground. Two players take turns removing one, two, or three sticks. The player to remove the last stick loses.

Theorem 2.5. *The first player has a winning strategy iff the number of sticks, n, is not $4k + 1$ for any $k \in \mathbb{N}$.*

A strategy is a rule for how many sticks to remove when there are n left. We show that if $n = 4k+1$, then player 2 has a strategy that will force a win for him, otherwise, player 1 has a strategy that will force a win for him.

Proof. The induction hypothesis is: for all $k \in \mathbb{N}$, if $n = 4k + 1$, then the first player loses, and if $n = 4k$, $4k + 2$, or $4k + 3$, the first player wins. This exhausts all possible cases for n.

We proceed by strong induction, using starting from 1.

Base case: $n = 1$. The first player has no choice but to remove 1 stick and lose, which is what the theorem says for this case.

Strong inductive step: Suppose the theorem is true for numbers 1 through n and show that it is true for $n + 1$. For the inductive step, there are four cases:

- $n + 1 = 4k + 1$: show that the first player loses. We've already handled the base case (1) so we can assume $n + 1 \geq 5$. Consider what the first player might do to win: he can choose to remove 1, 2 or 3 sticks. If he removes one stick, the remaining number of sticks is $n = 4k$. By strong induction, the player who plays at this point has a winning strategy. So the player who played first will lose.

  Similarly, if the first player removes two sticks, the remaining number is $4(k-1)+3$. Again, he loses, by the same reasoning. Similarly, by removing 3 sticks, he loses. So, however the first player moves, he loses.

- $n + 1 = 4k$: show that the first player can win.

  Have the first player remove 3 sticks: the second player then sees $4(k - 1) + 1$ sticks, and loses, by the strong inductive hypothesis.

- $n + 1 = 4k + 2$: show that the first player can win.

  Have the first player remove 1 stick: the second player then sees $4k + 1$ sticks, and loses as in the previous case.

- $n + 1 = 4k + 3$: show that the first player can win.

  Have the first player remove 2 sticks: again, the second player sees $4k + 1$ sticks and loses.

□

3 Induction, Strong Induction, and Least Number Principle

We argued above that strong induction is better than ordinary induction, but it's worth observing now that it's only "better" from the point of view of writing up a proof, not because it can be used to prove *more* theorems. It is always possible to convert a proof using one form of induction into a proof using the other.

Of course the conversion from induction to strong induction is trivial because an ordinary induction proof already *is* a strong induction proof—think about that! It's conversion the other way that's interesting.

3.1 Converting Strong Induction to Ordinary Induction [Optional]

Here is a recipe for converting, piece-by-piece, a strong induction proof that some proposition, $P(n)$, holds for all n, into an ordinary induction proof.

- For the new, ordinary induction proof, use the hypothesis $Q(n)$ where

  $$Q(n) ::= \forall m \leq n \ P(m).$$

- In the base case, the strong induction proof establishes $P(0)$. In the new proof, we can use exactly the same argument to establish $Q(0)$, since $Q(0)$ is equivalent to $P(0)$.
- In the inductive step, the strong induction proof shows that $\forall m \leq n \ P(m) \longrightarrow P(n+1)$. In other words, the old induction step proof concludes that

  $$Q(n) \longrightarrow P(n+1).$$

  But since $Q(n)$ implies itself, we can add an additional conclusion to the proof, namely,

  $$Q(n) \longrightarrow (Q(n) \land P(n+1)).$$

- But $(Q(n) \land P(n+1))$ is equivalent to $Q(n+1)$, so we can add as a final conclusion that

  $$Q(n) \longrightarrow Q(n+1).$$

So by adding the previous two conclusions at the end of the induction case of the strong induction proof, we wind up with an ordinary induction proof of $\forall n \ Q(n)$.

3.2 Least Number Principle

Another proof method closely related to induction depends on the

Axiom (Least Number Principle). Every nonempty subset, $S \subseteq \mathbb{N}$, has a smallest element.

The Least Number Principle (LNP) looks nothing like the induction axiom, and it may seem obvious but useless.

But as for obvious, note that this axiom would be false if the set of non-negative integers, $\mathbb{N}$, were replaced by, say, the set, $\mathbb{Z}$, of *all* integers, or the set, $\mathbb{Q}^+$, of positive rational numbers. Neither of these sets has a least element. So the LNP is capturing something special about the natural numbers.

As for useless, recall that at the end of Section 2 we used the LNP to "prove" the strong induction axiom. If you look back at this proof, you can read it as a recipe for converting any strong induction proof into an LNP proof—similar to the recipe we gave for converting a strong induction into ordinary induction. So LNP is at least as useful as strong induction!

Conversely, we can use strong induction to prove the LNF. This allows us to convert any LNF proof into a strong induction proof, if we choose. In short, a proof using induction, strong induction, or the LNF to prove some proposition can always be converted in a proof using any the other methods. Mathematicians like LNP, because it is often "prettier" (fewer symbols) than an induction proof. On the other hand, as it often involves proof by contradiction, using the LNP is not always the best approach. The choice of method is really a matter of style—but style does matter.

[Optional] To prove the LNP by strong induction, let $P(n)$ be the predicate that every set of natural numbers containing the number n also contains a smallest element. So if we prove $\forall n\ P(n)$, then we have proved the LNP, since a nonempty set has to contain *some* element n.

Proof. We prove $\forall n\ P(n)$ by strong induction. The induction hypothesis is $P(n)$.

Base case $P(0)$: If a set contains 0, then 0 is its smallest element.

strong induction step: Assume $\forall m \leq n\ P(m)$, and prove $P(n+1)$.

Consider any set, S, containing the integer, $n+1$. If $n+1$ is actually the smallest element of S, then we are done. Otherwise, S must contain a smaller element $m < n+1$. But then $m \leq n$, and the strong induction hypothesis implies that S contains a smallest element, and we are done in this case too. □

Massachusetts Institute of Technology
6.042J/18.062J, Fall '02: Mathematics for Computer Science
Professor Albert Meyer and Dr. Radhika Nagpal

Course Notes 3

Relations

A "relation" is a fundamental mathematical notion expressing a relationship between elements of sets.

Definition 0.1. A *binary relation from a set A to a set B* is a subset $R \subseteq A \times B$.

So, R is a set of ordered pairs. We often write $a \sim_R b$ or aRb to mean that $(a, b) \in R$.

Functions, for example, are a type of relation. The abstract notion of a relation is useful both in mathematics and in practice for modeling many different sorts of relationships. It's the basis of the *relational database* model, the standard data model for practical data processing systems.

Many times we will talk about a "relation on the set A", which means that the relation is a subset of $A \times A$. We can also define a *ternary* relation on A as a subset $R \subseteq A^3$ or, in general, an n-ary relation as a subset $R \subseteq A^n$, or $R \subseteq A_1 \times A_2 \times \cdots \times A_n$ if the sets A_i are different. In this class, we will focus only on binary relations. Here are some examples:

1. The relation "is taking class" as a subset of $\{$students at MIT$\} \times \{$classes at MIT$\}$. A relation from students to classes.
2. The relation "has lecture in" as a subset of $\{$classes at MIT$\} \times \{$rooms at MIT$\}$. A relation from classes to rooms.
3. The relation "is living in the same room" as a subset of $\{$students at MIT$\} \times \{$students at MIT$\}$. A relation on students.
4. The relation "can drive from first to second city". (Not necessarily directly—just some way, on some roads.)
5. Relation on computers, "are connected (directly) by a wire"
6. "meet one another on a given day"
7. "likes"
8. Let $A = \mathbb{N}$ and define $a \sim_R b$ iff $a \leq b$.
9. Let $A = \mathcal{P}(\mathbb{N})$ and define $a \sim_R b$ iff $a \cap b$ is finite.
10. Let $A = \mathbb{R}^2$ and define $a \sim_R b$ iff $d(a, b) = 1$.
11. Let $A = \mathcal{P}(\{1, \ldots, n\})$ and define $a \sim_R b$ iff $a \subseteq b$.

1 Properties of Relations

Once we have modeled something abstractly as a relation, we can talk about its properties without referring to the original problem domain. For a relation on a set A there are several standard properties of relations that occur commonly. Later on we will use these properties to classify different types of relations.

Definition 1.1. A binary relation R on A is:

1. *reflexive* if for every $a \in A$, $a \sim_R a$.
2. *symmetric* if for every $a, b \in A$, $a \sim_R b$ implies $b \sim_R a$.
3. *antisymmetric* if for every $a, b \in A$, $a \sim_R b$ and $b \sim_R a$ implies $a = b$.
4. *asymmetric* if for every $a, b \in A$, $a \sim_R b$ implies $\neg(b \sim_R a)$.
5. *transitive* if for every $a, b, c \in A$, $a \sim_R b$ and $b \sim_R c$ implies $a \sim_R c$.

The difference between antisymmetric and asymmetric relations is that antisymmetric relations may contain pairs (a, a), i.e., elements can be in relations with themselves, while in an asymmetric relation this is not allowed. Clearly, any asymmetric relation is also antisymmetric, but not vice versa.

Among our relations from Example :

- Relation 3 is reflexive, symmetric, transitive.
- Relation 4 is reflexive, transitive. Not necessarily symmetric, since roads could be one-way (consider Boston), but in actuality . . . . But definitely not antisymmetric.
- Relation 5 is symmetric but not transitive. Whether it is reflexive is open to interpretation.
- Relation 6 likewise.
- Relation 7 is (unfortunately) not symmetric. Not antisymmetric. Not transitive. Not even reflexive!
- Relation 8 is reflexive, antisymmetric, transitive.
- Relation 9 is not reflexive. It is symmetric. It is not transitive. $\{$even naturals$\}\cap\{$odd naturals$\}$ is finite (empty), but not $\{$even naturals$\} \cap \{$even naturals$\}$.
- Relation 10 is only symmetric.
- Relation 11 is reflexive, antisymmetric and transitive.

2 Representation

There are many different ways of representing relations. One way is to describe them by properties, as we did above. For infinite sets, that's about all we can do. But for finite sets, we usually use some method that explicitly enumerates all the elements of the relation. Some alternatives are lists, matrices and graphs. Why do we have so many different ways to represent relations? Different representations may be more efficient for encoding different problems and also tend to highlight different properties of the relation.

2.0.1 Lists

A finite relation from set A to set B can be represented by a list of all the pairs.

Example 2.1. The relation from $\{0, 1, 2, 3\}$ to $\{a, b, c\}$ defined by the list:

$\{(0, a), (0, c), (1, c), (2, b), (1, a)\}$.

Example 2.2. The divisibility relation on natural numbers $\{1, \ldots, 12\}$ is represented by the list:

$$\begin{gathered}\{(1,1),(1,2),(1,3),(1,4),(1,5),(1,6),(1,7),(1,8),(1,9),(1,10),(1,11),(1,12),(2,2),(2,4),\\ (2,6),(2,8),(2,10),(2,12),(3,3),(3,6),(3,9),(3,12),(4,4),(4,8),(4,12),(5,5),(6,6),\\ (6,12),(7,7),(8,8),(9,9),(10,10),(11,11),(12,12)\}.\end{gathered}$$

We can recognize certain properties by examining this representation:

Reflexivity: Contains all pairs (a, a).

Symmetry: Contains (a, b) then contains (b, a).

Transitivity: Contains (a, b) and (b, c) then contains (a, c).

2.0.2 Boolean Matrices

Boolean matrices are a convenient representation for representing relations in computer programs. The rows are for elements of A, columns for B, and for every entry there is a 1 if the pair is in the relation, 0 otherwise.

Example 2.3. The relation from Example 2.1 is represented by the matrix

$$\begin{array}{c|ccc} & a & b & c \\ \hline 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 2 & 0 & 1 & 0 \\ 3 & 0 & 0 & 0 \end{array}$$

Example 2.4. The divisibility relation over $\{1, 2, \ldots, 12\}$ is represented by the enormous matrix

$$\begin{array}{c|cccccccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\
\hline
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
2 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\
3 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
5 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
6 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
7 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}$$

Again, properties can by recognized by examining the representation:

Reflexivity the major diagonal is all 1.

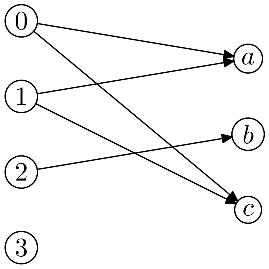
Symmetry: the matrix is clearly not symmetric across the major diagonal.

Transitivity: not so obvious . . .

2.0.3 Digraphs

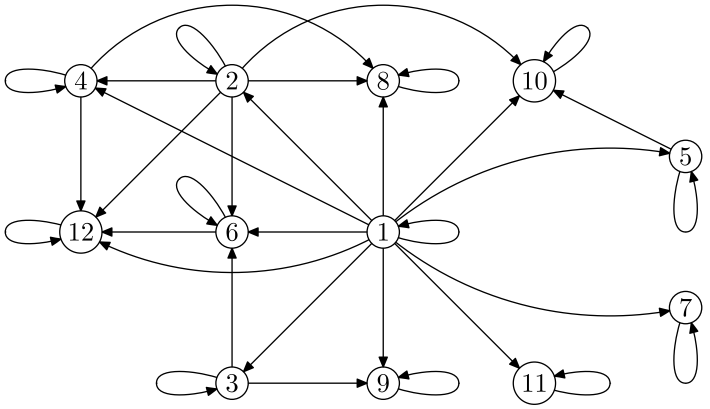
We can draw a picture of a relation $R \subseteq A \times B$ by drawing a dot for every element of A, a dot for every element of B, and an arrow from $a \in A$ to $b \in B$ iff aRb. Such a picture is called a *directed graph*, or *digraph* for short.

Example 2.5. The relation from Example 2.1 is represented by the digraph



Digraphs are mainly used for relations where $A = B$, i.e., for relations on a finite set A. To represent such a relation as a digraph we draw a dot (vertex) for each element of A, and draw an arrow from first element to second element of each pair in the relation. The digraph may contain self-loops, i.e. arrows from a dot to itself, associated to the elements a such that (a, a) is in the relation.

Example 2.6. The divisibility relation over $\{1, 2, \ldots, 12\}$ is represented by the digraph



Reflexivity: All nodes have self-loops.

Symmetry: all edges are bidirectional.

Transitivity: Short-circuits—for any sequence of consecutive arrows, there is a single arrow from the first to the last node.

3 Operations on Relations

3.1 Inverse

If R is a relation on $A \times B$, then R^{-1} is a relation on $B \times A$ given by $R^{-1} = \{(b, a) \mid (a, b) \in R\}$. It's just the relation "turned backwards."

Example 3.1. Inverse of "is taking class" (Relation 1 in Example) is the relation "has as a student" on the set $\{\text{classes at MIT}\} \times \{\text{students at MIT}\}$; a relation from classes to students.

We can translate the inverse operation on relation to operations on the various representations of a relation. Given the matrix for R, we can get the matrix for R^{-1} by transposing the matrix for R (note that the inverse of a relation is not the same thing as the inverse of the matrix representation). Given a digraph for R, we get the graph for R^{-1} by reversing every edge in the original digraph.

3.2 Composition

The *composition* of relations $R_1 \subseteq A \times B$ and $R_2 \subseteq B \times C$ is the relation

$$R_2 \circ R_1 = \{(a, c) \mid (\exists b)((a, b) \in R_1) \wedge ((b, c) \in R_2)\}.$$

In words, the pair (a, c) is in $R_2 \circ R_1$ if there exists an element b such that the pair (a, b) is in R_1 and the pair (b, c) is in R_2. Another way of thinking about this is that a "path" exists from element a to c via some element in the set B[1].

[1] Notice that $R_2 \circ R_1$ and $R_1 \circ R_2$ are different. The symbol $\circ$ is a source of eternal confusion in mathematics—if you read a book you should always check how the authors define $\circ$—some of them define composition the other way around.

Example 3.2. The composition of the relation "is taking class" (Relation 1 in Example) with the relation "has lecture in" (Relation 2 in Example) is the relation "should go to lecture in", a relation from {students at MIT} to {rooms at MIT}.

Example 3.3. Composition of the parent-of relation with itself gives grandparent-of. Composition of the child-of relation with the parent-of relation gives the sibling-of relation. (Here we relax the meaning of sibling to include that a person is the sibling of him/herself.) Does composition of parent-of with child-of give married-to/domestic partners? No, because that misses childless couples.

Example 3.4. Let B be the set of boys, G be the set of girls, $R_1 \subseteq B \times G$ consist of all pairs (b, g) such that b is madly in love with g, and $R_2 \subseteq G \times B$ consist of all pairs (g, b) such that g is madly in love with b. What are the relations $R_2 \circ R_1$ and $R_1 \circ R_2$, respectively?

3.2.1 Computing Composition and Path Lengths

If we represent the relations as matrices, then we can compute the composition by a form of "boolean" matrix multiplication, where $+$ is replaced by $\vee$ (Boolean OR) and $\times$ is replaced by $\wedge$ (Boolean AND).

Example 3.5. Let R_1 be the relation from Example 2.1:

	a	b	c
0	1	0	1
1	1	0	1
2	0	1	0
3	0	0	0

Let R_2 be the relation from $\{a, b, c\}$ to $\{d, e, f\}$ given by:

	d	e	f
a	1	1	1
b	0	1	0
c	0	0	1

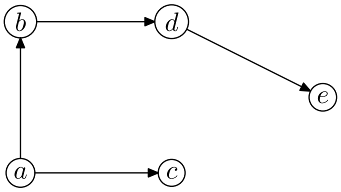
Then $R_2 \circ R_1 = \{(0, d), (0, e), (0, f), (1, d), (1, e), (1, f), (2, e)\}$, that is,

	d	e	f
0	1	1	1
1	1	1	1
2	0	1	0
3	0	0	0

A relation on a set A can be composed with itself. The composition $R \circ R$ of R with itself is written R^2. Similarly R^n denotes R composed with itself n times. R^n can be recursively defined: $R^1 = R$, $R^n = R \circ R^{n-1}$.

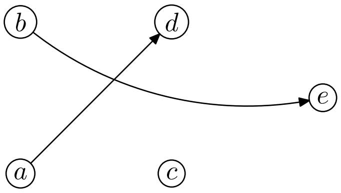
Example 3.6. Consider the relation $R = \{(a,b),(a,c),(b,d),(d,e)\}$ or:

	a	b	c	d	e
a	0	1	1	0	0
b	0	0	0	1	0
c	0	0	0	0	0
d	0	0	0	0	1
e	0	0	0	0	0



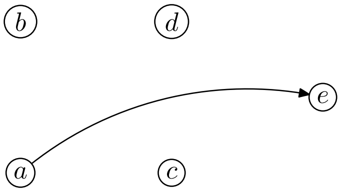
R^2 will be:

	a	b	c	d	e
a	0	0	0	1	0
b	0	0	0	0	1
c	0	0	0	0	0
d	0	0	0	0	0
e	0	0	0	0	0



R^3 will be:

	a	b	c	d	e
a	0	0	0	0	1
b	0	0	0	0	0
c	0	0	0	0	0
d	0	0	0	0	0
e	0	0	0	0	0



Definition 3.7. A *path* in a relation R is a sequence $a_0, \ldots, a_k$ with $k \geq 0$ such that $(a_i, a_{i+1}) \in R$ for every $i < k$. We call k the *length* of the path.

In the digraph model, a path is something you can trace out by following arrows from vertex to vertex, without lifting your pen. Note that a singleton vertex is a length 0 path (this is just for convenience). A *simple path* is a path with no repeated vertices.

Lemma 3.8. *$R^n = \{(a,b) \mid$ there is a length n path from a to b in $R\}$*

Proof. By induction. Let

$$P(n) ::= R^n = \{(a,b) \mid \text{there is a length } n \text{ path from } a \text{ to } b \text{ in } R\}.$$

The base case is clear. There is exactly one edge from a to b for every $(a,b) \in R$. And there is exactly one pair $(a,b) \in R$ for every edge from a to b, $P(1)$ is true. Note that since the induction hypothesis is an equality we have to prove both sides.

For the inductive step, suppose $P(n)$ is true.

First consider a path $a_0, \ldots, a_{n+1}$ in R. This is a path $a_0, \ldots, a_n$ in R followed by a pair (a_n, a_{n+1}) of R. By the inductive hypothesis, we can assume that $(a_0, a_n) \in R^n$. And we have already mentioned that $(a_n, a_{n+1}) \in R$. Therefore $(a_0, a_{n+1}) \in R^{n+1}$ by the definition of composition. Thus, every path of length $n+1$ corresponds to a relation in R^{n+1}.

Now consider a pair $(a, b) \in R^{n+1}$. By the definition of composition, there exists a c such that $(a, c) \in R^n$ and $(c, b) \in R$. By the inductive hypothesis, we can assume that (a, c) corresponds to a length n path from a to c, and since $(c, b) \in R$ there is an edge from c to b. Thus, there is a length $n+1$ path from a to b. To conclude, $P(n) \longrightarrow P(n+1)$. □

3.3 Closure

A closure "extends" a relation to satisfy some property. But extends it as little as possible.

Definition 3.9. The *closure* of relation R with respect to property P is the relation S that

(i) contains R,

(ii) has property P, and

(iii) is contained in *any* relation satisfying (i) and (ii).

That is, S is the "smallest" relation satisfying (i) and (ii).

As a general principle, there are two ways to construct a closure of R with respect to property P: We can either start with R and add as few pairs as possible until the new relation has property P; or we can start with the largest possible relation (which is $A \times A$ for a relation on A) and then remove as many not-in-R pairs as possible while preserving the property P.

3.3.1 The Reflexive Closure

Lemma 3.10. *Let R be a relation on the set A. The reflexive closure of R is $S = R \cup \{(a, a), \forall a \in A\}$.*

Proof. It contains R and is reflexive by design. Furthermore (by definition) any relation satisfying (i) must contain R, and any satisfying (ii) must contain the pairs (a, a), so any relation satisfying both (i) and (ii) must contain S. □

Example 3.11. Let $R = \{(a, b)(a, c)(b, d)(d, e)\}$, then the reflexive closure of R is

$$\{(a, b)(a, c)(b, d)(d, e)(a, a)(b, b)(c, c)(d, d)(e, e)\}.$$

3.3.2 The Symmetric Closure

Lemma 3.12. *Let R be a relation on the set A. The symmetric closure of R is $S = R \cup R^{-1}$.*

Proof. This relation is symmetric and contains R. It is also the smallest such. For suppose we have some symmetric relation T with $R \subseteq T$. Consider $(a, b) \in R$. Then $(a, b) \in T$ so by symmetry $(b, a) \in T$. It follows that $R^{-1} \subseteq T$. So $S = R \cup R^{-1} \subseteq T$. □

Example 3.13. Let $R = \{(a, b)(a, c)(b, d)(d, e)\}$, then the symmetric closure of R is

$$\{(a, b)(a, c)(b, d)(d, e)(b, a)(c, a)(d, b)(e, d)\}$$

3.3.3 The transitive closure

The transitive closure is a bit more complicated than the closures above.

Lemma 3.14. *Let R be a relation on the set A. The transitive closure of a relation R is the set*

$$S = \{(a, b) \in A^2 \textit{given there is a path from } a \textit{ to } b \textit{ in } R\}.$$

Proof. Obviously, $R \subseteq S$. Next, we show that S is transitive. Suppose $(a, b) \in S$ and $(b, c) \in S$. This means that there is an (a, b) path and a (b, c) path in R. If we "concatenate" them (attach the end of the (a, b) path to the start of the (b, c) path, we get an (a, c) path. So $(a, c) \in S$. So S is transitive.

Finally, we need to show that S is the smallest transitive relation containing R. So consider any transitive relation T containing R. We have to show that $S \subseteq T$. Assume for contradiction that $S \not\subseteq T$. This means that some pair $(a, b) \in S$ but $(a, b) \notin T$. In other words, there is a path $a_0, \ldots, a_k = b$ in R where $k \geq 1$ and $(a_0, a_k) \notin T$. Call this a *missing path.* Now let M be the set of missing paths.

Now we use well-ordering. We have just claimed that the set M of missing paths is nonempty. So consider a shortest missing path $s_0, \ldots, s_m$. We will derive a contradiction to this being the shortest missing path.

Case 1: $m = 1$. Then s_0, s_1 is a path in R, so $(s_0, s_1) \in R$. But we know T contains R, so $(s_0, s_1) \in T$, a contradiction.

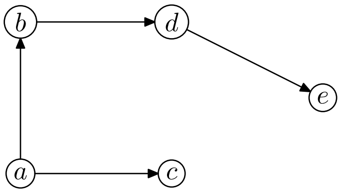
Case 2: $m > 1$. Then $s_1, \ldots, s_{m-1}$ is a path in R ($m - 1 > 0$). But it is shorter than our original shortest missing path, so cannot be missing. Thus $(s_1, s_{m-1}) \in T$. Also we have $(s_{m-1}, s_m) \in T$ since $R \subseteq T$. Thus by transitivity of T, $(s_1, s_m) \in T$, a contradiction.

We get a contradiction either way, so our assumption (that $S \not\subseteq T$) is false. This completes the proof. □

Wait a minute. Well-ordering is applied to sets of *numbers;* we applied it to a set of paths! How? Well, look at the set of "lengths of missing paths". It is nonempty, so has a smallest element. There is path that has this length—so it is a shortest path.

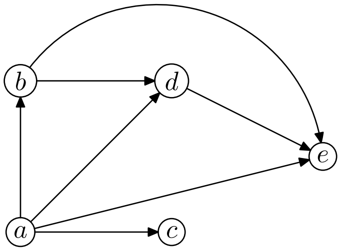
Example 3.15. The transitive closure of the relation

	a	b	c	d	e
a	0	1	1	0	0
b	0	0	0	1	0
c	0	0	0	0	0
d	0	0	0	0	1
e	0	0	0	0	0



is

	a	b	c	d	e
a	0	1	1	1	1
b	0	0	0	1	1
c	0	0	0	0	0
d	0	0	0	0	1
e	0	0	0	0	0



3.3.4 Computing the Transitive Closure

With reflexive and symmetric closure, it is pretty clear how to actually build them. But for transitive closure, how do we actually find all the paths we need?

Let's start by finding paths of a given length. Recall the definition of R^n of R composed with itself n times. We proved that $R^n = \{(a,b) given$there is a length n path from a to b in $R\}$. This means we can write the transitive closure of R as $R \cup R^2 \cup R^3 \cup \cdots$. Better—since we know how to do composition—but still a problem: there are infinitely many terms!

Lemma 3.16. *Suppose A has n elements and that R is a relation on A. Let a and b be elements of A and suppose that there is a path from a to b in R. Then, there is a path of length at most n from a to b in R.*

Proof. We'll use well-ordering (again). Consider the shortest path $a = a_0, a_1, \ldots, a_k$ from a to b (we know one exists, so by well-ordering there is a shortest one). Suppose $k > n$. Then some element of A appears twice in the list (with more than n list entries, one must be a repeat). This means the path is at some point circling back to where it was before. We can cut out this cycle from the path and get a shorter path. This contradicts that we had a shortest path. So we cannot have $k > n$. □

So we don't need infinitely many terms. It is enough to take paths of length at most n, namely $R^1 \cup R^2 \cup \cdots \cup R^n$.

4 Equivalence Relations and Partitions

We can use properties of relations to classify them into different types. We will be considering two important types of relations, *equivalence relations* and *partial orders*.

Definition 4.1. An *equivalence relation* is a relation that is reflexive, symmetric and transitive.

For example, the "roommates" relation is an equivalence relation. So is "same size as", and "on same Ethernet hub". A trivial example is the $=$ relation on natural numbers. The hallmark of equivalence relation is the word *same*. It provides a way to hide unimportant differences. Using an equivalence relation we can actually partition the universe into subsets of things that are the "same", in a natural way.

Definition 4.2. A *partition* of a set A is a collection of subsets $\{A_1, \ldots, A_k\}$ such that any two of them are disjoint (for any $i \neq j$, $A_i \cap A_j = \emptyset$) and such that their union is A.

Let R be an equivalence relation on the set A. For an element $a \in A$, let $[a]$ denote the set $\{b \in A \text{ given } a \sim_R b\}$. We call this set the *equivalence class of a under R*. We call a a *representative* of $[a]$.

Lemma 4.3. *The sets $[a]$ for $a \in A$ constitute a partition of A. That is, for every $a, b \in A$, either $[a] = [b]$ or $[a] \cap [b] = \emptyset$.*

Proof. Consider some arbitrary $a, b \in A$. If either $[a] = [b]$ or $[a] \cap [b] = \emptyset$ then we are done, so suppose not. Let c be any element that is in one of the sets but not the other. Without loss of generality we can assume that $c \in [b] - [a]$. (We know that either $c \in [b] - [a]$ or $c \in [a] - [b]$. In the latter case we can simply swap a and b and reduce to the first case.) Let d be any element in $d \in [a] \cap [b]$. We will get a contradiction by showing that $a \sim_R c$ and therefore that $c \in [a]$. First, $a \sim_R d$ because $d \in [a]$ (note that $d = a$ is a possibility but this is ok because R is reflexive). Second, $d \sim_R b$ and $b \sim_R c$ because both $c, d \in [b]$ and R is symmetric. This implies, by transitivity, that $d \sim_R c$. Finally, by transitivity, $a \sim_R c$ because $a \sim_R d$ and $d \sim_R c$. □

Note that all three properties of equivalence relations were used in this proof. Checking that the proof uses all available assumptions if usually a good sanity check when writing proofs—if one of the properties you assumed were not needed, you have either made a mistake or proven a much more strong theorem than you thought—*e.g.*, if you didn't use transitivity anywhere in the proof of Lemma 4.3, you would be proving that any reflexive symmetric relation produces a partition, which is false.

Lemma 4.4. *Any partition $\{A_1, \ldots, A_k\}$ of A defines an equivalence relation by letting $a \sim_R b$ iff a and b are in the same A_i.*

Proof. Reflexivity: for all a we know $a \in A_i$ for some i, by definition of partition. Clearly a and a are in the same A_i, *i.e.*, $a \sim_R a$.

Symmetry: Assume $a \sim_R b$, that is a and b are in the same A_i. Also b and a are in the same A_i and therefore $b \sim_R a$.

Transitivity: Assume $a \sim_R b$ and $b \sim_R c$. By definition of $\sim_R$, a and b are in the same A_i for some i, and b and c are in the same A_j for some j. But by definition of partition b cannot be in two different A_i's. So, it must be $A_i = A_j$ and a and c are in the same A_i, proving $a \sim_R c$. □

Therefore, we can look at partitions and at equivalence relations as the same thing.

4.1 Integers modulo m

A familiar and important equivalence relation on integers (positive, negative and 0) is:

Definition 4.5. If a and b are integers, then we say that $a \equiv b \pmod{m}$ if $m \mid (a - b)$.

$a \equiv b \pmod{m}$ is pronounced "a is equivalent to b modulo m". An equivalent formulation says that $a = b + km$ for some integer k.

Theorem 4.6. *Equality modulo m is an equivalence relation.*

Proof. We need to show that the relation is reflexive, symmetric, and transitive.

Reflexive: Clearly $m \mid (a - a) = 0$, so $a \equiv a \pmod{m}$.

Symmetric: If $a = b + km$ then $b = a + (-k)m$.

Transitive: Suppose $a \equiv b \pmod{m}$ and $b \equiv c \pmod{m}$. Then $m \mid (a - b)$ and $m \mid (b - c)$. So $(a - b) = k_1 m$ and $(b - c) = k_2 m$. So $(a - c) = (a - b) + (b - c) = (k_1 + k_2)m$. Therefore $m \mid (a - c)$. □

The equivalence class of a is the set $[a] = \{b \in \mathbb{Z} \mid a \equiv b \pmod{m}\}$, or $\{km + a \mid k \in \mathbb{Z}\}$.

It turns out that we can extend a lot of standard arithmetic to work modulo m. In fact, we can define notions of sum and product for the equivalence classes mod m. For example, we define $[a] + [b]$, given two equivalence classes mod m, to be the equivalence class $[a + b]$. This is not as obvious as it seems: notice that the result is given in terms of a and b, two selected representatives from the equivalence classes, but that it is supposed to apply to the equivalence classes themselves. To prove that this works, we have to show that it doesn't matter which representatives of the equivalence classes we choose:

Lemma 4.7. *If $a \equiv x \pmod{m}$ and $b \equiv y \pmod{m}$ then $(a + b) \equiv (x + y) \pmod{m}$.*

Proof. $m \mid (a - x)$ and $m \mid (b - y)$ so $m \mid ((a - x) + (b - y)) = (a + b) - (x + y)$, □

It follows that if we are interested only the result of the addition modulo m—which is the case, for example, in the RSA cryptosystem—then we can at any time replace a given number with a different number equivalent to it, without changing the value (equivalence class) of the final answer. The same fact can be proven for multiplication.

5 Partial Orders

Partial orders are another type of binary relation that is very important in computer science. They have applications to task scheduling, database concurrency control, and logical time in distributed computing,

Definition 5.1. A binary relation $R \subseteq A \times A$ is a *partial order* if it is reflexive, transitive, and antisymmetric.

Recall that antisymmetric mean $aRb \wedge bRa \Rightarrow a = b$, or $\forall a \neq b, aRb \Rightarrow \neg bRa$. In other words this relation is *never* symmetric! This single property is what distinguishes it from an equivalence relation. The reflexivity, antisymmetry and transitivity properties are abstract properties that generally describe "ordering" relationships.

For a partial order relation we often write an ordering-style symbol like $\preceq$, instead of just a letter like R, for a partial order relation. This lets us use notation similar to $\leq$. For example, we write $a \prec b$ if $a \preceq b$ and $a \neq b$. Similarly, we write $b \succeq a$ as equivalent to $a \preceq b$. But this could be misleading, note that $\geq$ is a partial order on natural numbers, as well as $\leq$. If we use the $\preceq$ symbol for $\geq$, things look really funny. In cases like this it is better to use R.

A partial order is always defined on some set A. The set together with the partial order is called a "poset":

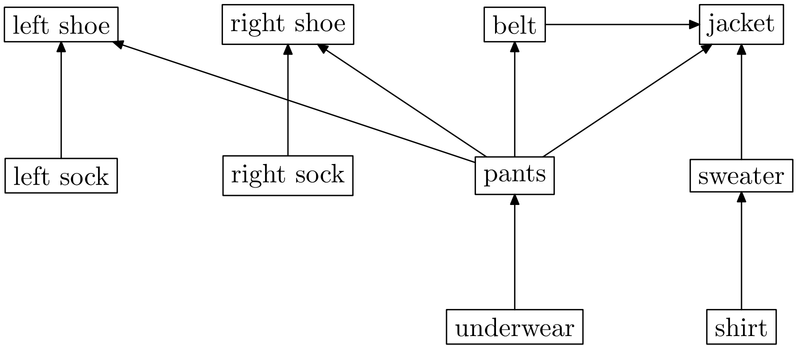
Definition 5.2. A set A together with a partial order $\preceq$ is called a *poset* $(A, \preceq)$.

Example 5.3. Consider the following relations:

- $A = \mathbb{N}, R =\leq$, easy to check reflexive, transitive, antisymmetric
- $A = \mathbb{N}, R =\geq$, same.
- $A = \mathbb{N}, R =<$, *not* because not reflexive
- $A = \mathbb{N}, R = \mid$ (divides), easy to check reflexive, transitive, antisymmetric
- $A = \mathcal{P}(\mathbb{N}), R =\subseteq$, check reflexive: $S \subseteq S$, transitive: $S \subseteq S' \wedge S' \subseteq S'' \Rightarrow S \subseteq S''$, antisymmetric: $S \subseteq S' \wedge S' \subseteq S \Rightarrow S = S'$.
- $A =$ "set of all computers in the world", $R =$ "is (directly or indirectly) connected to". *not* a partial order because it is not true that $aRb \wedge bRa \Rightarrow a = b$. In fact, it is symmetric and transitive. Equivalence relation.
- $A =$ "set of all propositions", $R =\Longrightarrow$, **not** because it's not antisymmetric. Not symmetric either, so not equivalence relation.

5.1 Directed Acyclic Graphs

A common source of partial orders in computer science is in "task graphs". You have a set of tasks A, and a relation R in which aRb means "b cannot be done until a is finished". Implicitly, "if all the things that point at b are done, I can do b." This can be nicely drawn as a graph. We draw an arrow from a to b if aRb. For example, below is a graphs that describes the order in which one would put on clothes. The set is of clothes, and the edges say what should be put on before what.



The "depends on" graph imposes an ordering on tasks. But what if I add a relation edge from belt to underwear? In that case my dependency graph stops making sense: there is no way to get dressed! What goes wrong? A cyclic dependency.

Definition 5.4. A *cycle* is a path that ends where it started (*i.e.*, the last vertex equals the first).

Definition 5.5. A *directed acyclic graph (DAG)* is a directed graph with no cycles.

Lemma 5.6. *Any partial order is a DAG.*

Proof. Suppose the graph representation of a partial order $\preceq$ has a cycle $a_1, \ldots, a_k, a_1$. Then by transitivity of $\preceq$ (with an induction hiding inside) we have $a_1 \preceq a_k$. We also have $a_k \preceq a_1$. This violates the antisymmetry of $\preceq$, a contradiction. □

But is it a partial order? No, because it isn't reflexive or transitive. But there is a natural extension: the reflexive transitive closure *is* a partial order. It gives the relation "must be done before."

Lemma 5.7. *The transitive reflexive closure of a DAG is a partial order.*

Proof. Let the DAG be R and its transitive reflexive closure S. S is transitive and reflexive; we just need to prove that it is antisymmetric. We do so by contradiction. Suppose that there exists some a, b such that $a \neq b$, $a \sim_S b$, and $b \sim_S a$. In other words, there is a path from a to b and a path from b to a in R. If we attach these two paths, we get a path from a to a in R, *i.e.*, R contains a cycle. This contradicts the assumption that R is acyclic. □

5.2 Partial vs. Total Orders

A partial order is called *partial* because it is not necessary that an ordering exists between every pair of elements in the set.

Example 5.8. Lshoe and Rshoe have no prescribed ordering between them.

Example 5.9. For two sets, it's not necessary that either be a subset of the other.

When there is no prescribed order between two elements we say that they are "incomparable".

Definition 5.10. We say that a and b are *incomparable* if neither $a \preceq b$ nor $b \preceq a$, and that they are *comparable* if $a \preceq b$ or $b \preceq a$.

Example 5.11. For subsets of $\mathbb{N}$, $\{1, 2, 3\}$ and $\{2, 3, 4\}$ are incomparable.

However, a partial order need not have incomparable elements. As a special case, we can have a partial order in which there is a specified order between every pair of elements.

Definition 5.12. A poset $(S, \preceq)$ is *totally ordered* if $(\forall a, b \in S)[a \preceq b \vee b \preceq a]$.

The DAG for a total order looks like a line.

5.3 Topological Sorting

Sometimes when we have a partial order, *e.g.*, of tasks to be performed, we want to obtain a consistent total order. That is, an order in which to perform all the tasks, one at a time, so as not to conflict with the precedence requirements.

The task of finding an ordering that is consistent with a partial order is known as *topological sorting*—probably because the sort is based only on the shape, *i.e.*, topology, of the poset, and not on the actual values.

Definition 5.13. A *topological sort* of a finite poset $(A, \preceq)$ is a total ordering of all the elements of A, $a_1, a_2, \cdots, a_n$ in such a way that for all $i < j$, either $a_i \preceq a_j$ or a_i and a_j are incomparable.

For example, underwear, shirt, Lsock, Rsock, pants, sweater, Lshoe, Rshoe, belt, jacket is a topological sort of how to dress. One of the nice facts about posets is that such an ordering always exists and is even easy to find:

Theorem 5.14. *Every finite poset has a topological sort.*

The basic idea to prove this theorem is to pick off a "first" element and then proceed inductively.

Definition 5.15. A *minimal* element a in a poset $(A, \preceq)$ is one for which $(\forall b \in A)[a \preceq b \vee a$ and b are incomparable$]$. Equivalently, it is an element a for which $(\nexists b \in A)[b \prec a]$.

Lemma 5.16. *Every finite poset* $(A, \preceq)$ *has a minimal element.*

Proof. For every element $a \in A$, let $p_a = \{b \in A \text{ given } b \prec a\}$, *i.e.*, p_a is the set of predecessors of a according to the partial order. It is enough to show that there exists an $a \in A$ such that $p_a = \emptyset$; to accomplish this, we use well-ordering.

Let $P = \{p_a \text{ given } a \in A\}$. Now let a' be an element corresponding to a set $p_{a'}$ of minimum cardinality. We now that such an a' exists by applying the well-ordering principle to the subset $\{|p| : p \in P\}$ of $\mathbb{N}$. There can be several sets in P of minimum cardinality, but that doesn't matter—we pick $p_{a'}$ to be one of the sets. We now prove by contradiction that $p_{a'} = \emptyset$.

Suppose that $|p_{a'}| > 0$, *i.e.*, that $p_{a'} \neq \emptyset$. Then there exists some $b' \in A$ such that $b' \prec a'$. Consider the set $p_{b'}$. We now claim that $p_{b'} \subset p_{a'}$. Since $c \in p_{b'}$ implies that $c \prec b'$, we obtain, by transitivity, that $c \in p_{b'}$ implies that $c \prec a'$, *i.e.*, every element in $p_{b'}$ is also an element of $p_{a'}$, or, equivalently, $p_{b'} \subseteq p_{a'}$. Furthermore, $b' \in p_{a'}$ since $b' \prec a'$, but $b' \notin p_{b'}$ since $b' \not\prec b'$. Thus $p_{b'} \subset p_{a'}$. But this contradicts our assumption that $p_{a'}$ is a set of minimum cardinality. Thus, $p_{a'} = \emptyset$. □

Example 5.17. Consider the dressing example. Construct an ordering by picking one item at a time. At each step, look at the poset formed by the remaining elements. Lsock, shirt, sweater, Rsock, underwear, pants, Lshoe, belt, jacket, Rshoe

Example 5.18. Subsets of $\{1, 2, 3, 4\}$:

$\emptyset$ is the unique minimal element, then we have choices, *e.g.*, do: $\{1\}$, $\{2\}$, $\{1, 2\}$, $\{3\}$, $\{1, 3\}$, $\{2, 3\}$, $\{1, 2, 3\}$, $\{4\}$, $\{1, 4\}$, $\{2, 4\}$, $\{3, 4\}$, $\{1, 2, 4\}$, $\{1, 3, 4\}$, $\{2, 3, 4\}$, $\{1, 2, 3, 4\}$

5.4 Parallel Task Scheduling

When elements of a poset are tasks that need to be done and the partial order is precedence constraints, topological sorting provides us with a legal way to execute tasks sequentially, *i.e.*, without violating any precedence constraints. But what if we have the ability to execute more than one task at the same time? For example, say tasks are programs, partial order indicates data dependence, and we have a parallel machine with lots of processors instead of a sequential machine with only one. How should we schedule the tasks? Our goal should be to minimize the total *time* to complete all the tasks. For simplicity, let's say all the tasks take the same amount of time and all the processors are identical.

So, given a finite poset of tasks, how long does it take to do them all, in an optimal parallel schedule? We can use partial order concepts to analyze this problem.

On the clothes example, we could do all the minimal elements first (Lsock, Rsock, underwear, shirt), remove them and repeat. We'd need lots of hands, or maybe dressing servants. We can do pants and sweater next, and then Lshoe, Rshoe, and belt, and finally jacket.

We can't do any better, because the sequence underwear, pants, belt, jacket must be done in that order. A sequence like this is called a chain.

Definition 5.19. A *chain* in a poset is a sequence of elements of the domain, each of which is smaller than the next in the partial order ($\preceq$ and $\neq$). The *length* of a chain is the number of elements in the chain.

Note that a chain is just a path in the corresponding graph.

Clearly, the parallel time is at least length of any chain. For if we used less time, then two tasks in the chain would have to be done at the same time. (This is "obvious," but is formalized as an application of the "pigeonhole principle" we will study shortly.) But by definition of chains this violates precedence constraints. A longest chain is also known as a *critical path*. So we need at least t steps, where t is the length of the longest chain. Fortunately, it is always possible to use only t:

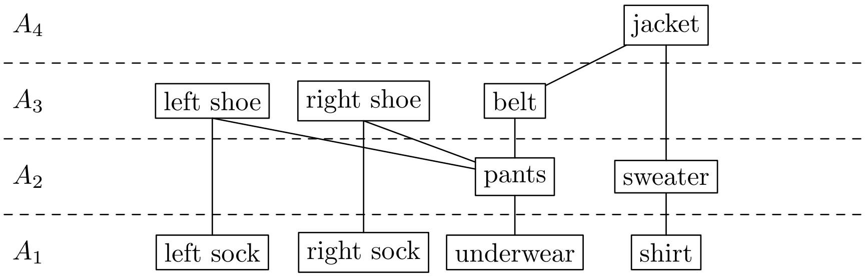
Theorem 5.20. *Given any finite poset $(A, \preceq)$ for which the longest chain has length t, it is possible to partition A into t subsets, $A_1, A_2, \cdots, A_t$ such that*

$$(\forall i \in \{1, 2, \ldots, t\})(\forall a \in A_i)(\forall b \prec a)\big[b \in A_1 \cup A_2 \cup \cdots \cup A_{i-1}\big].$$

That is, we can divide up the tasks into t groups so that for each group A_i, all tasks that have to precede tasks in A_i are in smaller-numbered groups.

Corollary 5.21. *For $(A, \preceq)$ and t as above, it is possible to schedule all tasks in t steps.*

Proof. For all i, schedule all elements of A_i at time i. This satisfies the precedence requirements, because all tasks that must precede a task are scheduled at preceding times. □



Corollary 5.22. *parallel time = length of longest chain*

So it remains to prove the partition theorem:

Proof of Theorem 5.20. Construct the sets A_i as follows:

$$A_i = \{a \in A \text{ given longest chain ending in } a \text{ has length } i\}.$$

This gives just t sets, because the longest chain has length t. Also, each $a \in A$ belongs to exactly one A_i. To complete the proof, we also need to show

$$(\forall i \in \{1, 2, \ldots, t\})(\forall a \in A_i)(\forall b \prec a)\big[b \in A_1 \cup A_2 \cup \cdots \cup A_{i-1}\big].$$

The proof is by contradiction. Assume that $a \in A_i$ and that there exists a $b \notin A_1 \cup A_2 \cup \cdots \cup A_{i-1}$ such that $b \prec a$. Then there is a chain of length exceeding $i - 1$ ending in b. This means, since $b \prec a$, that there is a chain of length $> i$ ending in a, which means that $a \notin A_i$. □

So with an unlimited number of processors, the time to complete all the tasks is the length of the longest chain. It turns out that this theorem is good for more than parallel scheduling. It is usually stated as follows.

Definition 5.23. An *antichain* is a set of incomparable elements.

Corollary 5.24. *If t is the length of the longest chain in a poset $(A, \preceq)$ then A can be partitioned into t antichains.*

Proof. Let the antichains be the sets A_i defined as in the proof of Theorem 5.20. We now claim that the elements in those sets are incomparable. Suppose that there exists $a, b \in A_i$ such that $a \neq b$ and a and b are comparable. Then either $a \prec b$ or $b \prec a$, which—again by the proof of Theorem 5.20—contradicts the assumption that a and b are in the same A_i. □

5.4.1 Dilworth's Theorem

We can use the above corollary to prove a famous result about posets:

Theorem 5.25 (Dilworth). *For all t, every poset with n elements must have either a chain of size greater than t or an antichain of size at least n/t.*

Proof. Assume there is no chain of length greater than t. So, longest chain has length at most t. Then by Corollary 5.24, the n elements can be partitioned into at most t antichains. Let ℓ be the size of the largest antichain. Since there are at most t antichains, every antichain contains at most ℓ elements, and an element belongs to exactly one antichain, $\ell t \geq n$. So there is an antichain with at least n/t elements. □

Corollary 5.26. *Every poset with n elements has a chain of length greater than $\sqrt{n}$ or an antichain of size at least $\sqrt{n}$.*

Proof. Set $t = \sqrt{n}$ in Theorem 5.25. □

Example 5.27. In the dressing poset, $n = 10$. Try $t = 3$. Has a chain of length 4. Try $t = 4$. Has no chain of length 5, but has an antichain of size $4 \geq 10/4$.

Posets arise in all sorts of contexts, and when they do, Dilworth's theorem can have interesting implications.

5.4.2 Increasing and Decreasing Sequences

Theorem 5.28. *In any sequence of n different numbers, there is either an increasing subsequence of length greater than $\sqrt{n}$ or a decreasing subsequence of length at least $\sqrt{n}$.*

Example 5.29. $\langle 6, 4, 7, 9, 1, 2, 5, 3, 8 \rangle$ has the decreasing sequence $\langle 6, 4, 1 \rangle$ and the increasing sequence $\langle 1, 2, 3, 8 \rangle$.

We can prove this using Dilworth's theorem; the trick is to define the appropriate poset. The domain is the set of values in the sequence. For the ordering, define $a \preceq b$ if either $a = b$, or else ($a < b$ and a comes before b in the sequence). You should check that this is reflexive (stated explicitly), transitive, and antisymmetric. A chain corresponds to a sequence that increases in value and moves to the right, that is, an increasing sequence. But what does an antichain correspond to?

Lemma 5.30. *If a and b are incomparable (under the partial order) and $a > b$ (as numbers) then a precedes b in the sequence.*

Proof. By contradiction. Assume b precedes a in the sequence. Then $b < a$ and b precedes a in the sequence, so $b \preceq a$ in the partial order. This contradicts our assumption that a and b are incomparable. □

We extend this lemma to more than 2 elements:

Lemma 5.31. *If $a_1, a_2, \cdots, a_t$ are incomparable and $a_1 > a_2 > \cdots > a_t$ then $a_1, a_2, \cdots a_t$ form a decreasing subsequence.*

Proof. For all i, the fact that $a_i > a_{i+1}$ implies that a_i precedes a_{i+1} in the sequence, by the previous lemma. □

So given an antichain, arrange the elements so that they are in decreasing order and the "left of" relation follows, giving a decreasing subsequence. Dilworth's theorem implies that there is either a chain of size greater than $\sqrt{n}$ or an antichain of size at most $\sqrt{n}$. By the analysis above, this yields either an increasing sequence of length greater than $\sqrt{n}$ or a decreasing subsequence of length at most $\sqrt{n}$.

Massachusetts Institute of Technology
6.042J/18.062J, Fall '02: Mathematics for Computer Science
Professor Albert Meyer and Dr. Radhika Nagpal

Course Notes 4

Graphs

1 Graph Definitions

Graphs are mathematical objects used heavily in Computer Science. Often one can reduce a real-world problem to a purely mathematical statement about graphs. If the graph problem can be solved, then — in principle at least — the real-world problem is solved. Already we've seen examples of using graphs to represent relations (which in turn model real world problems). Graphs give a picture of the relation, which is often much more revealing than a list of tuples.

A nuisance in first learning graph theory is that there are so many definitions. They all correspond to intuitive ideas, but can take a while to absorb. Worse, the same thing often has several names and can even have several equivalent definitions!

1.1 Simple Graphs

A *simple graph* is a pair of sets (V, E). Elements of V are called *vertices*. An element of E is called an *edge*. The basic property of an edge in a simple graph is that it adjoins two vertices. Formally, we identify an edge with the two vertices it adjoins. That is, the elements of E are specified to be subsets of V of size two.

Graphs are also sometimes called *networks*. Vertices are also sometimes called *nodes*. Edges are sometimes called *arcs*. If $u \neq v$ are vertices of a simple graph and the set $\{u, v\}$ is an edge of the graph, this edge is said to be *incident* to u and v. Equivalently, u and v are said to be *adjacent* or *neighbors*. Phrases like, "an edge joins u and v" and "the edge between u and v" are common. Notice that a simple graph in fact represents a *symmetric* relation, any two vertices, u and v, that are connected by an edge are related to each other in both directions (uRv and vRu). Simple graphs are also sometimes called *undirected* graphs.

Graphs can be nicely represented with a diagram of dots for vertices and lines for edges as shown in Figure 1.

1.2 Not Simple Graphs

Simple graphs are just one kind of graph; there are other kinds, often not as simple.

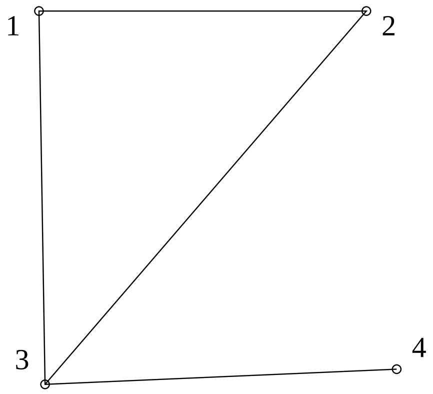


Figure 1: *This is a picture of a graph $G = (V, E)$. There are 4 vertices and 4 edges. The set of vertices V is $\{1, 2, 3, 4\}$. The set, E, of edges is $\{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{3, 4\}\}$. Vertex 1 is adjacent to vertex 2, but is not adjacent to vertex 4.*

Multigraphs

In a simple graph, there are either zero or one edges joining a pair of vertices. In a *multigraph*, multiple edges are permitted between the same pair of vertices.[1] There may also be edges called *self-loops* that connect a vertex to itself. Figure 2 depicts a multigraph with self-loops.

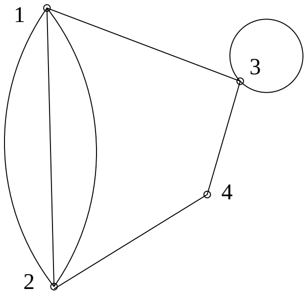


Figure 2: *This is a picture of a multigraph with a self-loop. In particular, there are three edges connecting vertices 1 and 2 and there is a self-loop on vertex 3.*

Directed Graphs

Like a simple graph, a *directed graph* or *digraph* is a pair of sets (V, E) where V is a set whose elements are called vertices. Now the edges are regarded not as lines, but as arrows going from a *start* (or *tail*) vertex to an *end* (or *head*) vertex. Formally, an edge in E is specified to be an ordered pair of vertices. In other words, $E \subseteq V \times V$, where $V \times V$, the Cartesian product of V with itself, is the set of all ordered pairs of elements of V. Notice that the definition of E is the same as that of a relation on the set V, so in fact a directed graph can be used to model any relation on a set. Figure 3 depicts a directed graph.

[1] This requires that an edge be represented as something slightly more than just two endpoints, for example as an ordered pair whose first element is a number and whose second element is the two endpoints.

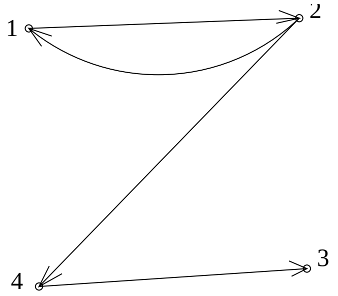


Figure 3: *This is a picture of a directed graph or digraph.*

Weighted Graphs

Sometimes it is useful to associate a number, often called its *weight*, with each edge in a graph. Such graphs are called *edge-weighted* or simply *weighted* graphs; they may be simple, directed, multi, etc. The weight of an edge (u, v) of a digraph is often denoted $w(u, v)$. More generally, edges (or nodes) may be labelled with elements from some designated set of labels — these are called edge (or node) *labelled* digraphs, simple graphs, multigraphs, etc.

2 Graphs in the Real World

There are many real-world phenomena that can be described nicely by graphs. Here are some examples:

Computer network The set of vertices V represents the set of computers in the network. There is an edge (u, v) iff there is a direct communication link between the computers corresponding to u and v.

Airline Connections Here the vertices are airports and edges are flight paths. We could indicate the direction that planes fly along each flight path by using a directed graph. We could use weights to convey even more information. For example, $w(i, j)$ might be the distance between airports i and j, or the flying time between them, or even the air fare. The edges might also be labelled with the set of call signs (`TW`, `AA`, . . .) of airlines using that flight path.

Precedence Constraints Suppose you have a set of jobs to complete, but some must be completed before others are begun. (For example, Atilla advises you always pillage *before* you burn.) Here the vertices are jobs to be done. Directed edges indicate constraints; there is a directed edge from job u to job v if job u must be done before job v is begun.

Program Flowchart Each vertex represents a step of computation. Directed edges between vertices indicate control flow.

Two-Player Game Tree All of the possibilities in a board game like chess can be represented in a graph. Each vertex stands for one possible board position. (For chess, this is a very big graph!)

Some graphs appear so frequently that they have names. The most important examples are shown in Figure 4.

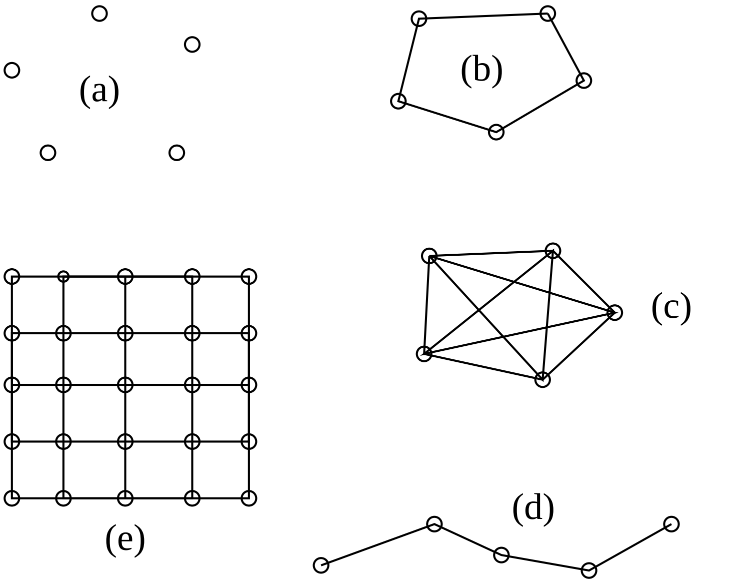


Figure 4: *The types of graph shown here are so common that they have names. (a) The* empty graph *or* Anticlique *on five vertices, A_5. An empty graph has no edges at all. (b) The* cycle *on five vertices, C_5. (c) The* complete graph *on five vertices, K_5. A complete graph has an edge between every pair of vertices. (d) A five-vertex* line graph. *(e) A* 5×5 2-dimensional mesh.

3 Graph Isomorphism

Graphs are intended to be abstract data types. This means what matters about a graph is only its connectedness: which vertices are incident to which edges, but not what the vertices actually are. For example the simple graph whose vertices are the integers $1, \ldots, 2n$ with an edge between two vertices iff they have the same parity (that is, both are even or both are odd), has the same connectedness as the graph whose vertices are $-1, -2, \ldots, -2n$ with an edge between two vertices iff either both vertices are $< -n$ or both are $\geq -n$, since in each case, the graph has two disjoint sets of n vertices, with all possible edges between vertices in the same set and none between a vertex and any of the vertices in the opposite set. Technically we say the graphs are *isomorphic* when their vertices can be put in exact correspondence so that an edge between two vertices in one graph corresponds exactly to an edge between corresponding vertices in the other graph.

It helps to say this in precise mathematical style. Namely, digraphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are isomorphic iff there is a bijection $f : V_1 \to V_2$ such that for all $u, v \in V_1$

$$(u, v) \in E_1 \longleftrightarrow (f(u), f(v)) \in E_2.$$

The bijection, f, is called an *isomorphism* between the graphs. For example, the function $f : \{-1, \ldots, -2n\} \to \{1, \ldots, 2n\}$, where $f(k) = -2k$ if $-n \leq k \leq -1$ and $f(k) = 2(2n + k) + 1$ if $-2n \leq k < -n$, is an isomorphism between the two graphs described above.

4 Properties of Graphs

When one is confronted by a new graph — say in a dark alley — there is often a need to study properties at a higher-level than, "Is vertex u connected to vertex v?" Some questions that might arise are, "How many edges are incident to each vertex?", "Is the graph all in one piece or in several pieces?", "How can we color the vertices?" (The last may seem odd, but comes up surprisingly often; we'll see why in a minute.) This section discusses some of these higher-level properties of graphs.

4.1 Vertex Degree

The *degree* of a vertex is the number of edges incident to it. The degree of vertex v is often denoted $d(v)$. In a digraph, we might also examine the *in-degree* (resp. *out-degree*) of a vertex, namely the number of edges into (resp. out of) a vertex.

For example, referring to Figure 4, every vertex in an empty graph has degree 0, but every vertex in a cycle has degree 2. A simple example of what we mean by a "higher-level" property of simple graphs is:

Theorem 4.1. *The sum of the degrees of the vertices in a simple graph equals twice the number of edges.*

Proof. Every edge contributes two to the sum of the degrees, one for each of its endpoints. □

Problem. (a) The previous proof is not as precise as we desire at the beginning of 6.042. Rewrite the proof more carefully as an induction on the number of edges in a simple graph.

(b) Extend Theorem 4.1 to multigraphs.

(c) Extend Theorem 4.1 to digraphs.

Here is a puzzle that can be addressed with graphs. There is a party. Some people shake hands an even number of times and some shake an odd number of times. Show that an even number of people shake hands an odd number of times.

We can represent the party by a graph. (Yeah, right.) Each person is represented by a vertex. If two people shake hands, then there is an edge between the corresponding vertices. This reduces the problem to the following theorem:

Theorem 4.2. *In every graph, there are an even number of vertices of odd degree.*

Proof. Partitioning the vertices into those of even degree and those of odd degree, we know

$$\sum_{v \in V} d(v) = \sum_{d(v) \text{ is odd}} d(v) + \sum_{d(v) \text{ is even}} d(v)$$

The value of the lefthand side of this equation is even, and the second summand on the righthand side is even since it is entirely a sum of even values. So the first summand on the righthand side must also be even. But since it is entirely a sum of odd values, it must must contain an even number of terms. That is, there must be an even number of vertices with odd degree. □

Two graphs with the same shape will of course have the same pattern of vertex degrees – that's one of things that "same shape" should imply. More precisely, if f is an isomorphism between two graphs, then it is easy to show that v and $f(v)$ have the same degree for any vertex v. We say that the degree of a vertex is *invariant* under graph isomorphism. Since f is a bijection, it follows that isomorphic graphs must have exactly the same numbers of vertices of any degree d. That is, the number of vertices of degree d in a graph is an invariant under isomorphism. Isomorphic graphs must also have the same number of pairs of vertices of degree d_1 and degree d_2 which are adjacent – another invariant under isomorphism. If two graphs are found to have different values of some invariant, then they are cannot be isomorphic. Finding such invariants can be a simple means to prove that two perhaps roughly similar looking graphs, are not actually isomorphic (see Rosen, 7.3, Examples 9, 10).

4.2 Chromatic Number

Time to discuss final exams. The MIT Schedules Office needs to assign a time slot for each final. This is not easy, because some students are taking several classes with finals, and a student can take only one test during a particular time slot. The Schedules Office wants to avoid all conflicts, but wants to make the exam period as short as possible.

This scheduling problem can be represented by a graph. Let each vertex represent a course. Put an edge between two vertices if there is some student taking both courses. Identify each possible time slot with a color. For example, Monday 9–12 is red, Monday 1–4 is blue, Tuesday 9–12 is green, etc.

If there is an edge between two vertices with the same color, then a conflict exam will have to be scheduled because there is a student who has to take exams for the courses represented by the vertices, but the exams are scheduled at the same time. Everyone wants to avoid conflict exams, so the registrar would like to color each vertex of the graph so that no adjacent vertices have the same color; to keep exam period as short as possible, the registrar would like to use the minimum possible number of colors. An example is shown in Figure 5.

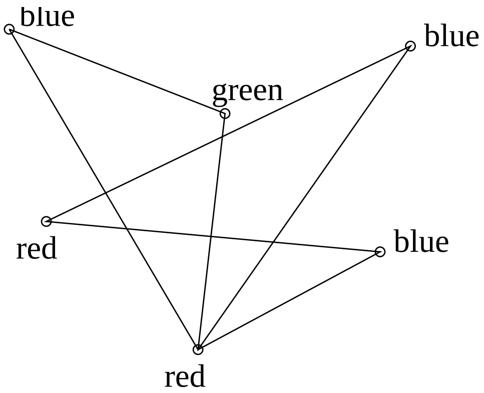


Figure 5: *This graph represents the exam scheduling problem. Each vertex stands for a course. An edge between two vertices indicates that a student is taking both courses and therefore the exams cannot be scheduled at the same time. Each exam time slot is associated with a color. A schedule that creates no conflicts for any student corresponds to a coloring of the vertices such that no adjacent vertices receive the same color.*

In general, the minimum number of colors needed to color the vertices of a graph G so that no two adjacent vertices are the same is called the *chromatic number* and is written $\chi(G)$. For example, if G is the 3-cycle or triangle graph, then $\chi(G) = 3$.[2]

The only completely general way known for finding the chromatic number of a graph is to exhaustively try all possible colorings; the exhaustive approach really is exhausting, and it becomes prohibitive even for graphs with only a few dozen vertices. Sometimes we can take advantage of the structure of special classes of graphs to get a better grip on their colorings. Also, we can at least put some general upper bounds on the chromatic number. For example, the chromatic number of an n-vertex graph is certainly at most n, since every vertex could be assigned a different color.

Suppose that we tried to do better by coloring vertices in an arbitrary order, but using a new color only when forced to do so. This strategy lies behind the following recursive algorithm. The input is a graph, and the output is a graph with appropriately colored vertices.

- Pick a vertex v. Remove v and all incident edges from the graph.
- If the graph is not empty, color the remainder recursively.
- Add v back to the graph. Color v differently from all neighbors, using a new color only if necessary.

Theorem 4.3. *The preceding algorithm colors a graph with at most $p + 1$ colors, where p is the maximum degree of any vertex.*

This theorem implies, for example, that a graph with thousands of vertices, each of degree 3, requires at most 4 colors. The proof is surprisingly easy:

Proof. The proof is by induction. Let $P(n)$ be the predicate that the preceding algorithm colors every n-vertex graph in which every vertex has degree at most p using at most $p + 1$ colors.

In the base case, $P(1)$, there is a single vertex with degree zero. In this case, $p = 0$ and the algorithm requires $p + 1 = 1$ colors.

In the inductive step, assume $P(n)$ to prove $P(n + 1)$. Let G' be the graph obtained from G by removing vertex v and incident edges. No vertex in G' has degree greater than p, since removing a vertex and incident edges can not increase the degree of any other vertex. (This is what we had to check to avoid "buildup error" — see below.) By induction, the algorithm (applied recursively) colors G' with at most $p + 1$ colors. Now we add back vertex v. Since v has at most p neighbors and there are $p + 1$ colors available, there is always one color left over for vertex v. Therefore, the algorithm colors G with at most $p + 1$ colors. □

4.3 Paths in Graphs

Is a graph all in one piece or composed of several pieces? To walk from one vertex to another, how many edges must one cross? Are there many different routes or just one? These are questions about *connectivity*.

[2] χ is the Greek letter "chi" — pronounced "he" by the Greeks and "kye" by (non-Greek) mathematicians and college fraternities.

A *path* in a digraph from a vertex u to a vertex v is a sequence of $k \geq 1$ edges

$$(v_0, v_1), (v_1, v_2), \ldots , (v_{k-1}, v_k)$$

such that $v_0 = u$ and $v_k = v$. A path may contain the same edge multiple times. The *length* of the path is k. The sequence of *sequence of vertices on the path* is the sequence $v_0, v_1, v_2, \ldots, v_k$. Vertex u is said to be *connected* to vertex v iff $u = v$ or there is a path from u to v. The definitions for simple graphs are the same except that the ordered pair (v_i, v_{i+1}) is replaced by the unordered set $\{v_i, v_{i+1}\}$. A path is *simple* when no vertex occurs more than once in the sequence of vertices on the path.

Lemma 4.4. *If vertex u is connected to vertex $v \neq u$ in a graph, then there is a simple path from u to v.*

The proof provides a nice example illustrating the Least Number Principle.

Proof. Since $u \neq v$, there is a path from u to v. By the Least Number Principle, there must be a minimum length path

$$(v_0, v_1), (v_1, v_2), \ldots , (v_{k-1}, v_k)$$

from u to v. We claim this path must be simple. This is clearly the case if $k = 1$, since then the path consists of a single edge from u to v. Given that $k > 1$, we prove the claim by contradiction.

Namely, assume that some vertex on the path occurs twice. More precisely, there are $i, j \in \mathbb{N}$ such that $i < j \leq k$ with $v_i = v_j$. Then removing the sequence of $j - i > 0$ edges

$$(v_i, v_{i+1}), \ldots, (v_{j-1}, v_j)$$

from the path yields a path from u to v of length $< k$, contradicting the fact that k is minimal. □

A *cycle* (also called a *circuit*) in a graph is a path from a vertex to itself, i.e. a sequence of edges $(u, v_1), \ldots, (v_{k-1}, u)$. A *simple cycle* is a cycle in which only the first and last vertex occurring on the path are the same and no edge occurs twice[3]. Another way to say this is that a simple cycle is either a self-loop or is a cycle which becomes a simple path when its first or last edge is deleted.

4.4 Adjacency Matrices

We defined a graph in terms of a set of vertices and a set of edges. A graph can also be described by a matrix.

Let $v_1, \ldots, v_n$ be list of the vertices of a digraph, G. The *adjacency matrix* of G is an $n \times n$ matrix of zeroes and ones. The entry in row i and column j of the matrix is one iff there is an edge from v_i to vertex v_j.

Weighted or labelled digraphs can also be described well by matrices: let the label of edge (v_i, v_j) be the matrix entry in row i, column j.

[3] The condition that no edge occurs twice is only needed for undirected graphs, where we don't want going back and forth on the same edge to count as a simple cycle.

Observe that in a digraph with n vertices, the paths from a vertex, u, to a vertex, v, can be partitioned (divided into nonoverlapping groups) according to their next-to-last vertex. That is, the paths from u to v can be partitioned into at most n sets, one for each vertex adjacent to v. The kth set consists of those paths, if any, which begin with a path from u to v_k followed by an edge from v_k to v. Matrix multiplication can now be given a neat graphical interpretation.

Theorem 4.5. *Let A be the $n \times n$ adjacency matrix of a digraph G, and let A^m be its mth power for $m > 0$. Then A^m_{ij}, the ijth entry of A^m, is exactly equal to the number of distinct paths of length m from vertex i to vertex j.*

Proof. By induction on m.

Base case $m = 1$ holds by definition.

To prove the induction case, note that from the partitioning of paths by next-to-last vertices described above, the number of paths from v_i to v_j of length $m > 1$ is the sum of the number of paths of length $m - 1$ from v_i to v_k with the sum taken over all k such that there is an edge from v_k to v_j. This is the same as

$$\sum_{k=1}^{n} |\{\text{length } m-1 \text{ paths from } v_i \text{ to } v_k\}| \, A_{kj}$$

But by induction, $|\{\text{length } m-1 \text{ paths from } v_i \text{ to } v_k\}| = A^{m-1}_{ik}$, and $\sum_{k=1}^{n} A^{m-1}_{ik} A_k j = A^m_{ij}$, so indeed A^m_{ij} is the number of paths from v_i to v_j of length m. □

4.5 Connectedness

Clearly if u is connected to v, and v is connected to w, then u is connected to w. That is, the “connected” relation is *transitive*. It is also *reflexive* since every vertex is by definition connected to itself. If the graph is simple, then obviously u is connected to v iff v is connected to u, so the relation is *symmetric*. That is, “connected” is actually an *equivalence relation* (*cf.* Rosen, section 6.5) on the set of vertices of a simple graph.

A graph is *connected* if there is a path between every pair of distinct vertices. For example, referring back to Figure 4, the empty graph is disconnected, but all others shown are connected.

A subset of the vertices of an undirected graph is a *connected component* of the graph if it consists of precisely all the vertices connected to some single vertex. That is, the connected components of a graph are the *equivalence classes* of the connectedness relation. In particular, every vertex of a simple graph belongs to exactly one connected component, and for any two vertices v_1, v_2, either the component of v_1 is the same as the component of v_2 (when v_1 is connected to v_2), or the two components have no elements in common (when v_1 is not connected to v_2).

Another way to say that a subset of vertices is a connected component is to say that every pair of vertices in the subset is connected, and the subset is *maximal* for this property, that is, no new vertex can be added to the subset which keeps the subset connected.

So a simple graph is connected iff it has exactly one connected component. The empty graph on n vertices has n connected components. A graph with three connected components is shown in Figure 6. We let $\gamma(G)$ be the number of connected components in a simple graph, G.

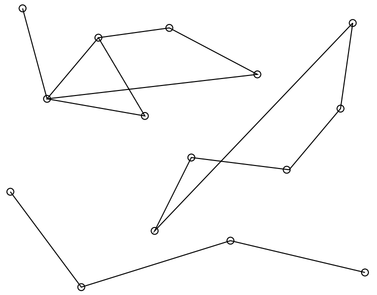


Figure 6: *This is a picture of a graph with 3 connected components.*

A False Theorem about Connectivity

If a graph is connected, then every vertex must be adjacent to some other vertex. Is the converse of this statement true? If every vertex is adjacent to some other vertex, then is the graph connected? The answer is no. In fact, the graph with three connected components shown in in Figure 6 is a counterexample. So what is wrong with the following proof?

False Theorem 4.6. *If every vertex in a graph is adjacent to another vertex, then the graph is connected.*

Nothing helps a false proof like a good picture; see Figure 7.

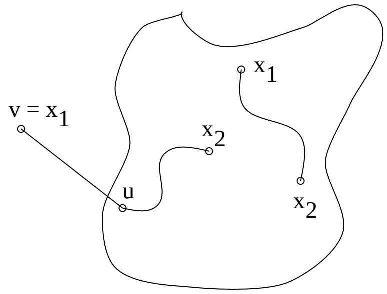


Figure 7: *This picture accompanies the false proof. Two situations are depicted. In one, vertices x_1 and x_2 both are among the vertices of G, and so there is a connecting path by induction. In the second, $v = x_1$ and x_2 is a vertex of G. In this case there is a connecting path because there is an edge from v to u and a path in G from u to x_2 by induction.*

Proof. The proof is by induction. Let $P(n)$ be the predicate that if every vertex in an n-vertex graph is adjacent to another vertex, then the graph is connected. In the base case, $P(1)$ is trivially true because there is only one vertex.

In the inductive step, we assume $P(n)$ to prove $P(n+1)$. Start with an $n+1$-vertex graph, G', in which every vertex is adjacent to another vertex. Now take some vertex v away from the graph and let the G be the remaining graph. By assumption v is adjacent in G' to one of the n vertices of G; call that one u.

Now we must show that for every pair of distinct vertices x_1 and x_2 in G', there is a path between them. If both x_1 and x_2 are vertices of G, then since G has n vertices, we may assume by induction it is connected. So there is a path between x_1 and x_2. Otherwise, one of the vertices is v (say x_1)

and the other, x_2 is in G. But x_2 is connected to u by induction, so there is a path from x_1 to u to x_2 as shown in the figure. □

The error is in the statement *"since G has n vertices, we may assume by induction it is connected."* The induction hypothesis does not say that every n-vertex graph is connected, but only, *"if every vertex in an n-vertex graph is adjacent to another vertex,* then the graph is connected". For example, if G' is the graph with vertices 1, 2, 3, 4 and edges $\{1, 2\}$ and $\{3, 4\}$, then removing vertex 1 to form G leaves vertex 2 without an adjacent vertex in G, and we can't conclude by induction that G is connected (which of course it isn't).

This is a variant of "buildup error." We're proving something about graphs with the property that every vertex is adjacent to another vertex. The argument implicitly assumes that any size $n + 1$ graph with the property can be built up from a size n graph with the same property, but not every such size $n + 1$ graph can be built this way.

A True Theorem about Connectivity

If a graph has too few edges, then there cannot be a path between every pair of vertices. More generally, a graph with a very small number of edges ought to have many connected components. (Remember, "many connected components" does not mean "very connected"; rather, it means "broken into many pieces"!) The following theorem generalizes these observations.

Theorem 4.7. *Adding a single edge to a simple graph reduces the number of connected components by at most one. More precisely, let G be a simple graph and G' be G with the addition of a single edge between two vertices of G. Then $\gamma(G') \geq \gamma(G) - 1$.*

Proof. If the new edge is between two vertices which are already connected, then the connectedness relation between vertices remains the same with or without the edge, so G and G' have exactly the same connected components and $\gamma(G') = \gamma(G)$. If the new edge is between vertices v_1 and v_2 which are not connected in G, then the component of v_1 and the component of v_2 are distinct in G and become one component of G'. The components of G which contain neither v_1 nor v_2 remain as connected components of G'. The effect is that the number of components in G' is one less than in G, *viz.*, $\gamma(G') = \gamma(G) - 1$. □

Corollary 4.8. *For any simple graph $G = (V, E)$,*

$$\gamma(G) \geq |V| - |E| .$$

For example, a graph with 0 edges and n vertices has $n - 0 = n$ connected components. A graph with 100 vertices and 35 edges must have at least 100 - 35 = 65 connected components.

Proof. By induction on $|E|$. (This may seem odd, because for $|E| > |V|$, the theorem says that the number of connected components is greater than some negative number! This is useless, but certainly true, so there is no harm.)

The induction hypothesis will be $P(k)$: any graph $G = (V, E)$ with k edges has at least $|V| - k$ connected components.

In the base case, $P(0)$ holds because in a graph with 0 edges, each vertex is a connected component, so $\gamma(G) = |V| \geq |V| - 0$.

In the inductive step, let $G' = (V', E')$ be a graph with $k+1$ edges. We want to prove that $\gamma(G') \geq |V'| - (k+1)$.

Pick some edge of G' and let G be G' with the chosen edge removed. By the lemma above,

$$\gamma(G') \geq \gamma(G) - 1.$$

By induction we may assume that $P(k)$ is true, and since G has k edges, we have

$$\gamma(G) \geq |V| - k.$$

But $|V| = |V'|$, so

$$\gamma(G') \geq \gamma(G) - 1 \geq (\left|V'\right| - k) - 1 = \left|V'\right| - (k+1).$$

□

Since a graph G is connected iff $\gamma(G) = 1$, we conclude immediately:

Corollary 4.9. *If a simple graph $G = (V, E)$ is connected, then*

$$|E| \geq |V| - 1.$$

5 Trees

Tree are an important special type of graph for CS applications. We have just seen that $n-1$ edges are required for connectivity. They are sometime sufficient, for example if we connect all the nodes in a line. Let's explore the smallest connected graphs. What is a good notion of smallest? No wasted edges.

Definition 5.1. A *tree* is a connected graph with no cycles.

The vertices in a tree can be classified into two categories. Vertices of degree at most one are called *leaves*, and vertices of degree greater than one are called *internal nodes*.

Trees are often drawn as in Figure 8 with the leaves on the bottom and a single node (called the *root* at the top). Keep this convention in mind; otherwise, phrases like "all the vertices *below*" will be confusing. [4]

Trees arise in many problems. Family trees are an example—each node is a person, and there is an edge between any parent and child. Similarly, the file structure in a computer system can often be represented by a tree. In this case, each internal node corresponds to a directory, and each leaf corresponds to a file. If one directory contains another, there there is an edge between the associated internal nodes. If a directory contains a file, then there is an edge between the internal node and a leaf. In both family trees and directories, there can be exceptions that make the graph not a tree—relatives can marry and have children, while directories sometimes use soft links to create multiple directory entries for a given file.

There are in fact many different equivalent ways of defining trees formally.

[4] (The English mathematician Littlewood once remarked that he found such directional terms particularly bothersome, since he habitually read mathematics reclined on his back!)

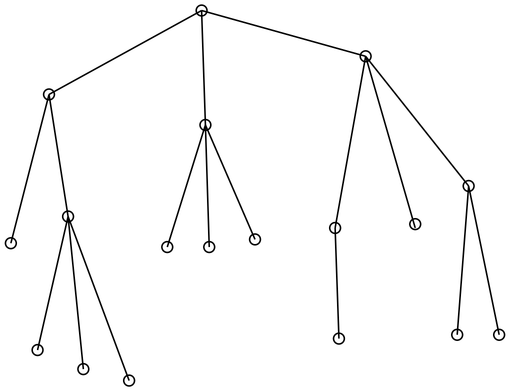


Figure 8: This tree has 11 *leaves*, which are defined as vertices of degree at most one. The remaining 7 vertices are called *internal nodes*.

Theorem 5.2. *For any simple graph $G = (V, E)$, the following are equivalent:*

1. *G is connected and $|E| = |V| - 1$.*
2. *G is connected, but removing any edge from G leaves a disconnected graph.*
3. *G is connected and acyclic.*
4. *There is a unique simple path between any two distinct vertices of G.*

Of course it would be a waste of effort to prove that each statement implied all the others – twelve different implications. Instead, we prove that each implies the next, and the fourth implies the first – only four implications. Some of the implications among these properties are harder to prove directly than others, so the order in which they are listed will affect the simplicity of the four direct implications to be proved. You might wonder how to pick an order which leads to simple proofs. There was no trick used here, just trial and error (over several hours) to find an order which allowed the simplest proofs – and there may well be a still simpler one we overlooked.

Proof.

- (Property 1. implies Property 2.) This follows immediately from Corollary 4.9.
- (2. implies 3.) By contradiction.

  Suppose Property 2. holds, but G has a simple cycle. Then removing the first edge in the cycle leaves a simple path connecting all the vertices in the cycle. This implies that the connected component of the vertices in the cycle is the same after the edge is removed as it was in G. Since there was only one connected component to begin with, there is still only one after the edge is removed. That is, the graph is still connected, contradicting Property 2.
- (3. implies 4.) By contradiction.

  Suppose Property 3. holds, but there are distinct simple paths between two vertices of G. That is, there are vertices $u \neq v$ and distinct simple paths P_1 with vertices $u, w_1, \ldots, w_n, v$ and P_2 with vertices $u, w'_1, \ldots, w'_{n'}, v$. Among all such u, v, P_1, P_2, we can, by the Least Number Principle, choose some for which P_1 is shortest.

Suppose P_1 a single edge, that is $P_1 = \{u, v\}$. If P_2 started with this same edge, then P_2 without its first edge would be a simple cycle from v to v, contradicting proposition 3. On the other hand, if P_2 did not start with $\{u, v\}$, then since $\{u, v\} = \{v, u\}$, the path starting with this edge followed by the edges in P_2 form a simple cycle from v to v, again contradicting proposition 3.

Hence, P_1 is of length greater than one, and $1 \leq n \leq n'$. Moreover, the first edge of P_1 must differ from the first edge of P_2, since otherwise $w_1 = w'_1$ and P_1 and P_2 without their first edges would be distinct paths between w_1 and v, contradicting the minimality of the length of P_1.

Now we claim that no vertex w_i internal to P_1 is the same as any vertex w'_j internal to P_2. This follows because if $w_i = w'_j$, then the two paths $u, w_1, \ldots, w_i$ and $u, w'_1, \ldots, w'_j$ between u and w_i would be distinct because their first edges differ, and the first path would be shorter than P_1, again contradicting the minimality of P_1. So no w_i can equal any w'_j. Hence the concatenation of all but the last vertex of P_1 with the reversal of P_2, namely,

$$u, w_1, \ldots, w_n, v, w'_{n'}, \ldots, w'_1, u$$

is a simple cycle, contradicting proposition 3.

- (4. implies 1.)

  By strong induction on $|V|$. The induction hypothesis is that 4 implies 1 for all graphs with n vertices.

  (Base case: $|V| = 1$) Property 1. is immediate if G has one vertex.

  (Induction) Suppose 4. implies 1. for all graphs with $1 \leq k \leq n$ vertices. Let G be a graph with $n + 1$ vertices satisfying proposition4. We must show that 1. holds for G.

  Since G has $n + 1 \geq 2$ vertices, and any two vertices are connected by a simple path, G certainly has at least one edge. Let G' be the graph which is left after removing some edge, $\{u, v\}$, from G.

  If there was still a path between u and v in G', then by Lemma 4.4 there would a simple path between u and v in G'. But then in G, this path and the edge $\{u, v\}$ would be distinct simple paths between u and v, contradicting 4. So G' cannot be connected. Let $G_i = (V_i, E_i)$ for $1 \leq i \leq n \geq 2$ be the set of connected components of G'.

  Now each connected component G_i still has unique simple paths and has fewer vertices than G, so by strong induction $|E_i| = |V_i| - 1$. Now we have

$$\begin{array}{rclclcl} |V| & = & \sum_{i=1}^{n} |V_i| & = & \sum_{i=1}^{n} (|E_i| + 1) & & \\ & = & (\sum_{i=1}^{n} |E_i|) + n & = & |E'| + n & & \\ & \geq & |E'| + 2 & = & (|E| - 1) + 2 & = & |E| + 1. \end{array}$$

  That is, $|E| \leq |V| - 1$. But since G is connected, we have from Corollary 4.9 that $|E| \geq |V| - 1$. So $|E| = |V| - 1$.

□

6 Euler Tours

The Koenigsberg bridge problem: can we cross all seven bridges in the town of Koenigsberg exactly once, without repeats, returning to our starting point?

```
          |    |           |     /
      ----|----|---   -----|----/
     /    |    |   \ /     |
----<             --X--
     \    |    |   / \     |
      ----|----|--/   \----|----\
          |    |           |     \
```

This problem has a graph representation as shown in Figure 9

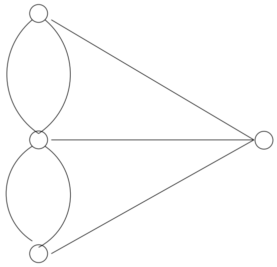


Figure 9: Graph representation for the Koenigsberg bridge problem.

Stated as a general graph theory problem, the problem is to construct a circuit of the graph that traverses every edge exactly once. (Each edge will be traversed in exactly one direction.) (Actually, note that the Koenigsberg graph is technically a multigraph, not a simple graph, because it has multiple edges between two of the pairs of vertices. We could change this into a simple graph by adding some extra vertices in the middle of the repeated edges.) Euler, in 1736, proved that it can't be done. So we call such tours "Euler Tours."

Definition 6.1. An Euler tour of an undirected graph G is a circuit that traverses every edge of G exactly once.

Theorem 6.2. *If undirected graph G has Euler tour, then G is connected and every vertex has even degree.*

Proof. Direct every edge according to tour. We enter a vertex as many times as we leave it. So every vertex must have indegree equal to outdegree. So total even. □

It follows that there is no tour of Koenigsberg, because there is an odd-degree vertex.

100 years later, Hierholzer proved the converse (that's the trouble with coming second—no one remembers your name!)

Theorem 6.3. *If G is connected and every node has even degree, then it has an Euler tour.*

Proof. If G has only one node then it has a trivial (1-node) Euler tour. So assume G has at least two nodes. Then G must have at least one edge (to be connected). Starting from this edge, construct a circuit with no repeated edges: trace from node to node until we cannot go any further on an unused edge. Because every node has even degree, this can only happen when we return to the node we started at (every other node we reach will have another edge by which we may leave).

Let C be the longest circuit without any repeated edges. If C includes all the edges of G, we are done. So assume it doesn't. Let H be the subgraph of G consisting of all the edges of G that aren't in C, and all the nodes of G that are incident on these edges. We claim that some node u in H is also in C. Why? Because G is connected.

Now choose any edge of H incident upon u. Starting with that edge, construct a cycle in H, following the same procedure as above. Eventually, it has to get back to its starting point u. Now we have two cycles with disjoint edge sets and a common vertex u. Splice them together to get a larger cycle than C. The contradicts the choice of C as maximum. □

7 Hamiltonian Circuits

A slightly different question: is there a simple circuit that traverses every *vertex* exactly once? Such a circuit is called a *Hamiltonian circuit* (Hamiltonian cycle). Similarly, a simple path that traverses every vertex exactly once is a *Hamiltonian path.*

The Rosen text has some simple conditions for determining the existence or nonexistence of Hamiltonian cycles in many cases. But although there's only a small change in switching the question from Euler circuits to Hamiltonian circuits, these two kinds of circuits have dramatically different properties. Specifically, while determining the existence of a Euler circuit in a graph is easy, determining Hamiltonian circuite is very complicated.

In fact, no simple criterion is known for determining whether or not a graph has a Hamiltonian circuit. And it's not merely that we're ignorant of a criterion: there are powerful theoretical arguments supporting the belief that there *is no* simple criterion for Hamiltonian circuits. The Hamiltonian circuit problem is an example of an *NP-complete* problem. Not only don't we expect to find a simple criterion for solving any *NP*-complete problem, but we don't even expect there to be a not-so-simple criterion which could still be checked quickly by a computer program. The theory of *NP*-completeness is a basic topic in Algorithms and Computability courses; we shall not describe it further in these Notes.

[Optional] Here is a slightly harder result.

Lemma 7.1. *Any graph with n vertices, $n \geq 3$, in which the minimum degree of each node is at least $n/2$, has a Hamiltonian circuit.*

Proof. By contradiction. Suppose some graph has n vertices, $n \geq 3$, and the minimum degree of each node is at least $n/2$, but that graph does not have a Hamiltonian circuit.

If we add edges to this graph one at a time, we eventually end up with a complete graph, which does have a Hamiltonian circuit (why?). Somewhere in the process of adding edges, we have a graph, G, that doesn't have a Hamiltonian circuit, but adding one more edge (u, v) yields a graph, G', that does have a Hamiltonian circuit. We'll get a contradiction for this G.

Since G plus the one edge (u, v) has a Hamiltonian circuit, G alone has a Hamiltonian path from u to v (just remove the one new edge). Say the path is $u = u_1, \ldots , u_n = v$; by definition this path includes all the nodes of G.

Now let's play with that path and turn it into a circuit, to get a contradiction. We will use the fact that u and v each have degree at least $n/2$ to produce two edges to replace one edge (u_i, u_{i+1}) on the path . . . .

Now let's count: Of the $n - 2$ intermediate nodes on the path $u_2, \ldots , u_{n-1}$ (all but the first and last) we know that at least $n/2$ are neighbors of u, and at least $n/2$ are neighbors of v. So it can be shown that there are two adjacent nodes, u_i and u_{i+1}, where u_i is a neighbor of v and u_{i+1} is a neighbor or u. Postpone showing this for a minute . . . .

Then cut and add edges, to produce a Hamiltonian circuit, contradiction.

Now, how do we get u_i and u_{i+1}? Just count how many are in various sets. Use a little trick:

Let S be $\{i : u_{i+1}$ is a neighbor of $u\}$.

Let T be $\{i : u_i$ is a neighbor of $v\}$.

Each of S and T has at least $n/2$ elements. Since there are only $n - 2$ possible values of i, some i must be in both sets. That is, u_i is a neighbor of v and u_{i+1} is a neighbor of u. □

This gives a special class of graphs with Hamiltonian circuits. But no one has nice criteria for finding Hamiltonian circuits in general, or for determining if they exist.

State Machines: Invariants and Termination

1 Modeling Processes

The topic for the week is the application of induction and other proof techniques to the design and analysis of algorithms and systems. We will focus on the problem of proving that some simple algorithms behave correctly.

Proving the correctness of a program is a quite different activity than debugging and testing a program. Since programs are typically intended to handle a huge, if not infinite, number of different inputs, completely testing a program on all inputs is rarely feasible, and partial testing always leaves open the possibility that something will go wrong in the untested cases. A proof of correctness ensures there are no such loopholes. Correctness proofs for hardware and software are playing a growing role in assuring system quality, especially for systems performing critical tasks such as flying airplanes, controlling traffic, and handling financial transactions.

Before we get into the abstract definitions, it will help to look at a couple of entertaining examples.

2 Die Hard

In the movie Die Hard 3, Bruce Willis and Samuel Jackson are coerced by a homicidal maniac into trying to disarm a bomb on a weight-sensitive platform near a fountain. To disarm the bomb, they need to quickly measure out exactly four gallons of water and place it on the platform. They have two empty jugs, one that holds three gallons and one that holds five gallons, and an unlimited supply of water from the fountain. Their only options are to fill a jug to the top, empty a jug completely, or pour water from one jug to the other until one is empty or the other is full. They do succeed in measuring out the four gallons while carefully obeying these rules. You can figure out how (or go see the movie or §3.3 below).

But Bruce is getting burned out on dying hard, and according to rumor, is contemplating a sequel, Die Once and For All. In this film, they will face a more devious maniac who provides them with the same three gallon jug, but with a *nine* gallon jug instead of the five gallon one. The water-pouring rules are the same. They must quickly measure out exactly four gallons or the bomb will go off.

This time the task is impossible—whether done quickly or slowly. We can prove this without much difficulty. Namely, we'll prove that it is impossible, by any sequence of moves, to get exactly four gallons of water into the large jug.

A sequence of moves is constructed one move at a time. This suggests a general approach for proofs about sequential processes: to show that some condition always holds during the executions of a process, use induction on the number, n, of steps or operations in the executions. For Die Hard, we can let n be the number of times water is poured.

All will be well if we can prove that neither jug contains four gallons after n steps for all $n \geq 0$. This is already a statement about n, and so it could potentially serve as an induction hypothesis. Let's try lunging into a proof with it:

Theorem 2.1. *Bruce dies once and for all.*

Let $P(n)$ be the predicate that neither jug contains four gallons of water after n steps. We'll try to prove $\forall n\, P(n)$ using induction hypothesis $P(n)$.

In the base case, $P(0)$ holds because both jugs are initially empty. In the inductive step, we assume that neither jug has four gallons after n steps and try to prove that neither jug has four gallons after $n + 1$ steps.

Now we are stuck; the proof cannot be completed. The fact that neither jug contains four gallons of water after n steps is not sufficient to prove that neither jug can contain four gallons after $n + 1$ steps. For example, after n steps each jug might hold two gallons of water. Pouring all water in the three-gallon jug into the nine-gallon jug would produce four gallons on the $n + 1$st step.

What to do? We use the familiar strategy of strengthening the induction hypothesis. Some experimentation suggests strengthening $P(n)$ to be the predicate that after n steps, the number of gallons of water in each jug *is a multiple of three*. This is a stronger predicate: if the number of gallons of water in each jug is a multiple of three, then neither jug contains four gallons of water. This strengthened induction hypothesis does lead to a correct proof of the theorem.

To be precise about this proof, we'll model the situation using a *state machine*.

3 State machines

3.1 Basic definitions

Mathematically speaking, a state machine is just a binary relation on states, where the pairs in the relation correspond to the allowed steps of the machine. In addition, a state machine has some states that are designated as start states—the ones in which it is allowed to begin executing.

Definition 3.1. A *state machine* has three parts:

1. a nonempty set, Q, whose elements are called *states*,
2. a nonempty subset $Q_0 \subseteq Q$, called the set of *start states*,
3. a binary relation, δ, on Q, called the *transition relation*.

Another view is that a state machine is really nothing more than a digraph whose nodes are the states and whose arrows are determined by the transition relation. Reflecting this view, we often write $q \to q'$ as alternative notation for the assertion that $(q, q') \in \delta$. The only extra state machine component beyond the digraph is its designated set of "start" nodes.

State machines come up in many areas of Computer Science. You may have seen variations of this definition in a digital logic course, a compiler course, or a theory of computation course. In these courses, the machines usually have only a *finite* number of states. Also, the edges in the state graphs are usually labelled with input and/or output tokens. We won't need inputs or output for the applications we will consider.

3.2 Examples

Here are some simple examples of state machines.

Example 3.2. A bounded counter, which counts from 0 to 99 and overflows at 100. The state graph is shown in Figure 1.

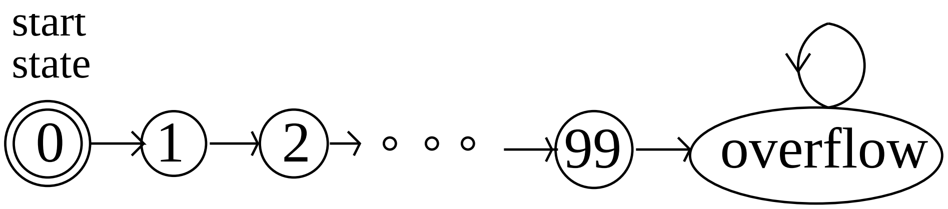


Figure 1: *The state graph of the 99-bounded counter.*

Formally, the state machine modeling the 99-counter has:

- $Q ::= \ \{0, 1, 2, \ldots, 99, \texttt{overflow}\}$,
- $Q_0 ::= \ \{0\}$,
- $\delta \ = \ \{(0, 1), (1, 2), (2, 3), \ldots, (99, \texttt{overflow}), (\texttt{overflow}, \texttt{overflow})\}$. Note the self-loop (transition to itself) for the overflow state.

Example 3.3. An unbounded counter is similar, but has an infinite state set, yielding an infinite digraph. This is harder to draw `:-)`

Example 3.4. The Die Hard 3 situation can be formalized as a state machine as well.

- $Q ::= \ \left\{(b, l) \in \mathbb{R}^2 \mid 0 \leq b \leq 5, 0 \leq l \leq 3\right\}$. Note that b and l are arbitrary real numbers, not necessarily integers. After all, Bruce could scoop any unmeasured amount of water into a bucket.
- $Q_0 = \{(0, 0)\}$ (because both jugs start empty).
- δ has several kinds of transitions:
  1. Fill the little jug: $(b, l) \to (b, 3)$ for $l < 3$.
  2. Fill the big jug: $(b, l) \to (5, l)$ for $b < 5$.
  3. Empty the little jug: $(b, l) \to (b, 0)$ for $l > 0$.

4. Empty the big jug: $(b, l) \to (0, l)$ for $b > 0$.
5. Pour from the little jug into the big jug: for $l > 0$,

$$(b, l) \to \begin{cases} (b + l, 0) & \text{if } b + l \leq 5, \\ (5, l - (5 - b)) & \text{otherwise.} \end{cases}$$

6. Pour from big jug into little jug: for $b > 0$,

$$(b, l) \to \begin{cases} (0, b + l) & \text{if } b + l \leq 3, \\ (b - (3 - l), 3) & \text{otherwise.} \end{cases}$$

Note that in contrast to the 99-counter state machine, there is more than one possible transition out of states in the Die Hard machine.

Problem 1. Which states of the Die Hard 3 machine have direct transitions to exactly two states?

A machine is called *deterministic* if its execution behavior is uniquely determined: there is only one start state and there is at most one transition out of every state.[1] Otherwise, it is *nondeterministic*. So the Die Hard machine is nondeterministic, and the counter machines are deterministic. Formally,

Definition 3.5. A state machine (Q, Q_0, δ) is *deterministic* iff the transition relation, δ, is the graph of a partial function on Q, and there is exactly one start state. Otherwise, the machine is *nondeterministic*.

3.3 Executions of state machines

The Die Hard 3 machine models every possible way of pouring water among the jugs according to the rules. Die Hard properties that we want to verify can now be expressed and proved using the state machine model. For example, Bruce will disarm the bomb if he can *reach* some state of the form $(4, l)$.

In graph language, a (possibly infinite) path through the state machine graph beginning at a start state corresponds to a possible system behavior or process execution. A state is reachable if there is a path to it starting from one of the start states. Formally,

Definition 3.6. An *execution* is a (possibly infinite) sequence $q_0, q_1, \ldots$ such that $q_0 \in Q_0$, and $\forall i \geq 0\, (q_i, q_{i+1}) \in \delta$. A state is *reachable* if appears in some execution.

Example 3.7. We said that Bruce and Samuel successfully disarm the bomb in Die Hard 3. In particular, the state (4,3) is reachable:

[1] In the case of state machines with inputs, a machine is deterministic when its execution is uniquely determined by the inputs it receives, but different inputs may lead to different executions.

action	state
start	(0,0),
fill the big jug	(5,0),
pour from big to little	(2,3),
empty the little	(2,0),
pour from big into little	(0,2),
fill the big jug,	(5,2),
pour from big into little	(4,3).

3.4 Die Hard Once and For All

Now back to Die Hard Once and For All. The problem is still to measure out four gallons, but with a nine gallon jug instead of the five gallon one. The states and transition relation are the same as for the Die Hard 3 machine, with all occurrences of "5" replaced by "9."

Now reaching any state of the form $(4, l)$ is impossible. To prove this carefully, we define $P(n)$ to be the predicate:

> At the end of any n-step execution, the number of gallons in each jug is an integer multiple of 3 gallons.

We prove $\forall n\, P(n)$ by induction.

Base case $n = 0$: $P(n)$ holds because each jug contains 0 gallons and $0 = 0 \cdot 3$ is an integer multiple of 3.

Induction step: Assume that $n \geq 0$ and some length n execution ends with b gallons in the big jug and l in the little jug. We may assume by induction that $P(n)$ holds, namely, that b and l are integer multiples of 3. We must prove $P(n+1)$. In particular, all we have to show is that after one more step, the amounts in the jugs are still integer multiples of 3.

The proof is by cases, according to which transition rule is used in the next step. For example, using the "fill the little jug" rule for the $n + 1$st transition, we arrive at state $(b, 3)$. We already know that b is an integer multiple of 3, and of course 3 is an integer multiple of 3, so the new state $(b, 3)$ has integer multiples of 3 gallons in each jug, as required. Another example is when the transition rule used is "pour from big jug into little jug" for the subcase that $b + l > 3$. Then the $n + 1$st state is $(b - (3 - l), 3)$. But since b and l are integer multiples of 3, so is $b - (3 - l)$. So in this case too, both jugs will contain an integer multiple of 3 gallons.

We won't bother to crank out the remaining cases, which can all be checked with equal ease. This completes the proof of Theorem 2.1: Bruce dies once and for all!

4 Reachability and Invariants

The induction proof about the Once and For All machine follows a proof pattern that is often used to analyze state machine behavior. Namely, we showed that the integer-multiple-of-3 property held at the start state and remained *invariant* under state transitions. So it must hold at all reachable states. In particular, since no state of the form $(4, l)$ satisfies the invariant, no such a state can be reachable.

Definition 4.1. An *invariant* for a state machine is a predicate, P, on states, such that whenever $P(q)$ is true of a state, q, and $q \to r$ for some state, r, then $P(r)$ holds.

Now we can reformulate the Induction Axiom specially for state machines:

Theorem 4.2 (Invariant Theorem). *Let P be an invariant predicate for a state machine. If P holds for all start states, then P holds for all reachable states.*

The truth of the Invariant Theorem is as obvious as the truth of the Induction Axiom. We could prove it, of course, by induction on the length of finite executions, but we won't bother.

4.1 The Robot

There is a robot. He walks around on a grid and at every step he moves one unit north or south *and* one unit east or west. (Read the last sentence again; if you do not have the robot's motion straight, you will be lost!) The robot starts at position $(0,0)$. Can the robot reach position $(1,0)$?

To get some intuition, we can simulate some robot moves. For example, starting at (0,0) the robot could move northeast to (1,1), then southeast to (0,2), then southwest to (-1, 1), then southwest again to (-2, 0).

Let's try to model the problem as a state machine and then prove a suitable invariant:

$$\begin{aligned} Q &::= \mathbb{Z} \times \mathbb{Z}, \\ Q_0 &::= \{(0,0)\}, \\ \delta &::= \left\{((i,j),(i',j')) \mid i' = i \pm 1 \wedge j' = j \pm 1\right\}. \end{aligned}$$

The problem is now to choose an appropriate predicate, P, on states and prove that it is an invariant. If this predicate is true for the start state (0,0) and false for $(1,0)$, then follows that the robot can never reach $(1,0)$. A direct attempt at an invariant is to let $P(q)$ be the predicate that $q \neq (1,0)$.

Unfortunately, this is not going to work. Consider the state $(2,1)$. Clearly $P((2,1))$ holds because $(2,1) \neq (1,0)$. And of course $P((1,0))$ does not hold. But $(2,1) \to (1,0)$, so this choice of P will not yield an invariant.

We need a stronger predicate. Looking at our example execution you might be able to guess a proper one, namely, that the sum of the coordinates is even! If we can prove that this is an invariant, then we have proven that the robot never reaches $(1,0)$ because the sum $1+0$ of its coordinates is not an even number, but the sum $0+0$ of the coordinates of the start state is an even number.

Theorem 4.3. *The sum of the robot's coordinates is always even.*

Proof. The proof uses the Invariant Theorem.

Let $P((i,j))$ be the predicate that $i+j$ is even.

First, we must show that the predicate holds for all start states. But the only start state is (0,0), and $P((0,0))$ is true because $0+0$ is even.

Next, we must show that P is an invariant. That is, we must show that for each transition $(i,j) \to (i',j')$, if $i+j$ is even, then $i'+j'$ is even. But $i' = i \pm 1$ and $j' = j \pm 1$ by definition of the transitions. Therefore, $i'+j'$ is equal to $i+j-2$, $i+j$, or $i+j+2$, all of which are even. □

Corollary 4.4. *The robot cannot reach* $(1, 0)$.

Problem 2. A robot moves on the two-dimensional integer grid. It starts out at $(0, 0)$, and is allowed to move in any of these four ways:

1. (+2,-1) Right 2, down 1
2. (-2,+1) Left 2, up 1
3. (+1,+3)
4. (-1,-3)

Prove that this robot can never reach (1,1).

Solution. A simple invariant that does the job is defined on states (i, j) by the predicate: $i + 2j$ is an integer multiple of 7. ■

5 Sequential algorithm examples

The Invariant Theorem was formulated by Robert Floyd at Carnegie Tech in 1967[2]. Floyd was already famous for work on formal grammars that had wide influence in the design of programming language syntax and parsers; in fact, that was how he got to be a professor even though he never got a Ph.D.

In that same year, Albert R. Meyer was appointed Assistant Professor in the Carnegie Tech Computation Science department where he first met Floyd. Floyd and Meyer were the only theoreticians in the department, and they were both delighted to talk about their many shared interests. After just a few conversations, Floyd's new junior colleague decided that Floyd was the smartest person he had ever met.

Naturally, one of the first things Floyd wanted to tell Meyer about was his new, as yet unpublished, Invariant Theorem. Floyd explained the result to Meyer, and Meyer could not understand what Floyd was so excited about. In fact, Meyer wondered (privately) how someone as brilliant as Floyd could be excited by such a trivial observation. Floyd had to show Meyer a bunch of examples like the ones that follow in these notes before Meyer realized that Floyd's excitement was legitimate — not at the truth of the utterly obvious Invariant Theorem, but rather at the insight that such a simple theorem could be so widely and easily applied in verifying programs.

Floyd left for Stanford the following year. He won the Turing award — the "Nobel prize" of Computer Science — in the late 1970's, in recognition both of his work on grammars and on the foundations of program verification. He remained at Stanford from 1968 until his death in September, 2001.

In this section of Notes we will describe two classic examples illustrating program verification via the Invariant Theorem: the Euclidean GCD Algorithm and "Fast" exponentiation.

[2] The following year, Carnegie Tech was renamed Carnegie-Mellon Univ.

5.1 Proving Correctness

It's generally useful to distinguish two aspects of state machine or process correctness:

1. The property that the final results, if any, of the process satisfy system requirements. This is called *partial correctness*. You might suppose that if a result was only partially correct, then it might also be partially incorrect, but that's not what's meant here. Rather, we mean that when there is a result, it is correct, but the process might not always produce a result. For example, it might run forever on some input without producing an output. The word "partial" comes from viewing such a process as computing a *partial function*.
2. The property that the process always finishes or is guaranteed to produce some desired output. This is called *termination*.

Partial correctness can commonly be proved using the Invariant Theorem.

5.2 The Euclidean Algorithm

Given two natural numbers, a, b, at least one of which is positive, the three thousand year old Euclidean algorithm will compute their GCD. The algorithm uses two registers x and y, initialized at a and b respectively. Then,

- if $y = 0$, **return** the answer x and terminate,
- else *simultaneously* set x to be y, set y to be the remainder of x divided by y, and repeat the process.

Example 5.1. Find $\gcd(414, 662)$:

1. remainder$(414, 662) = 414$ so repeat with $(662, 414)$,
2. remainder$(662, 414) = 248$, so repeat with $(414, 248)$,
3. remainder$(414, 248) = 166$, so repeat with $(248, 166)$,
4. remainder$(248, 166) = 82$, so repeat with $(166, 82)$,
5. remainder$(166, 82) = 2$, so repeat with $(82, 2)$,
6. remainder$(82, 2) = 0$, so repeat with $(2, 0)$,
7. return 2.

So $\gcd(414, 662) = 2$.

We can present this algorithm as a state machine:

- $Q ::= \ \mathbb{N} \times \mathbb{N}$,

- $Q_0 ::= \{(a, b)\}$,
- state transitions are defined by the rule

$$(x, y) \to (y, \text{remainder}(x, y)) \qquad\qquad \text{if } y \neq 0.$$

Next we consider how to prove that this state machine correctly computes $\gcd(a, b)$. We want to prove:

1. starting from $x = a$ and $y = b$, if we ever finish, then we have the right answer. That is, at termination, $x = \gcd(a, b)$. This is a *partial correctness* claim.
2. we do actually finish. This is a process *termination* claim.

5.2.1 Partial Correctness of GCD

First let's prove that if GCD gives an answer, it is a correct answer. Specifically, let $d ::= \gcd(a, b)$. We want to prove that *if* the procedure finishes in a state (x, y), then $x = d$.

Proof. So define the state predicate

$$P((x, y)) ::= \ [\gcd(x, y) = d].$$

P holds for the start state (a, b), by definition of d. Also, P is an invariant because

$$\gcd(x, y) = \gcd(y, \text{remainder}(x, y))$$

for all $x, y \in \mathbb{N}$ such that $y \neq 0$ (see Rosen Lemma 2.4.1). So by the Invariant Theorem, P holds for all reachable states.

Since the only rule for termination is that $y = 0$, it follows that if state (x, y) is terminated, then $y = 0$. So if this terminated state is reachable, then we conclude that $x = \gcd(x, 0) = d$. □

5.2.2 Termination of GCD

Now we turn to the second property, that the procedure must reach a terminated state.

To prove this, notice that y gets strictly smaller after any one transition. That's because value of y after the transition is the remainder of x divided by y, and this remainder is smaller than y by definition. But the value of y is always a natural number, so by the Least Number Principle, it reaches a minimum value among all its values at reachable states. But there can't be a transition from a state where y has its minimum value, because the transition would decrease y still further. So the reachable state where y has its minimum value is a terminated reachable state.

Note that this argument does not prove that the minimum value of y is zero, only that the minimum value occurs at termination. But we already noted that the only rule for termination is that $y = 0$, so it follows that the minimum value of y must indeed be zero.

5.3 Fast Exponentiation

The most straightforward way to compute the bth power of a number, a, is to multiply a by itself b times. This of course requires $b - 1$ multiplications. There is another way to do it using considerably fewer multiplications. This algorithm is called *Fast Exponentiation*:

Given inputs $a \in \mathbb{R}, b \in \mathbb{N}$, initialize registers x, y, z to $a, 1, b$ respectively, and repeat the following sequence of steps until termination:

```
1. if $z = 0$ return $y$ and terminate
2. $r :=$ remainder$(z, 2)$
3. $z :=$ quotient$(z, 2)$
4. if $r = 1$, then $y := xy$
5. $x := x^2$
```

We claim this algorithm always terminates and leaves $y = a^b$.

To be precise about the claim, we model this algorithm with a state machine:

1. $Q ::= \ \mathbb{R} \times \mathbb{R} \times \mathbb{N}$,
2. $Q_0 ::= \ \{(a, 1, b)\}$,
3. transitions

$$(x, y, z) \rightarrow \begin{cases} (x^2, y, \text{quotient}(z, 2)) & \text{if } z \text{ is positive and even,} \\ (x^2, xy, \text{quotient}(z, 2)) & \text{if } z \text{ is positive and odd.} \end{cases}$$

Let $d ::= a^b$. Since the machine is obviously deterministic, all we need to prove is that the machine will reach *some* state in which it returns the answer d.[3] Since the machine stops only when $z = 0$—at which time it returns y—all we need show is that a state of the form $(x, d, 0)$ is reachable.

We'll begin by proving partial correctness: *if* a state of the form $(x, y, 0)$ is reachable, then $y = d$.

We claim that predicate, P, is an invariant, where

$$P((x, y, z)) ::= \ [yx^z = d].$$

This claim is easy to check, and we leave it to the reader.

Also, P holds for the start state $(a, 1, b)$ since $1 \cdot a^b = a^b = d$ by definition. So by the Invariant Theorem, P holds for all reachable states. But only terminating states are those with $z = 0$, so if any terminating state $(x, y, 0)$ is reachable, then $y = yx^0 = d$ as required. So we have proved partial correctness.

[3] In a nondeterministic machine, there might be some states that returned the right answer, but also some that returned the wrong answer, so proving that a nondeterministic machine not only *can* return a correct answer, but *always* returns a correct one, tends to be more of a burden than for deterministic machines.

Note that as is often the case with induction proofs, the proof is completely routine once you have the right invariant (induction hypothesis). Of course, the proof is not much help in understanding how someone discovered the algorithm and its invariant. To learn more about that, you'll have to study Algorithms, say by taking 6.046.

What about termination? But notice that z is a natural-number-valued variable that gets smaller at every transition. So again by the Least Number Principle, we conclude that the algorithm will terminate. In fact, because z generally decreases by more than one at each step, we can say more:

Problem 3. Prove that it requires at most $2\log_2 b$ multiplications for the Fast Exponentiation algorithm to compute a^b for $b > 1$.

[Optional]

5.4 Extended GCD

An important elementary fact from number theory is that the gcd of two natural numbers can be expressed as an integer linear combination of them. In other words,

Theorem 5.2.

$$\forall m, n \in \mathbb{N}\, \exists k, l \in \mathbb{Z}\ \gcd(m, n) = km + ln.$$

We will prove Theorem 5.2 by extending the Euclidean Algorithm to actually calculate the desired integer coefficients k and l. In particular, given natural numbers m, n, with $n > 0$, we claim the following procedure[4] halts with integers k, l in registers K and L such that

$$km + ln = \gcd(m, n).$$

Inputs: $m, n \in \mathbb{N}, n > 0$.

Registers: X,Y,K,L,U,V,Q.

Extended Euclidean Algorithm:

```
X := m; Y := n; K := 0; L := 1; U := 1; V := 0;
loop:
if Y|X, then halt
else
  Q := quotient(X,Y);
         ;;the following assignments in braces are SIMULTANEOUS
 {X := Y,
  Y := remainder(X,Y);
  U := K,
  V := L,
  K := U - Q * K,
  L := V - Q * L};
goto loop;
```

Note that X,Y behave exactly as in the Euclidean GCD algorithm in Section 5.2, except that this extended procedure stops one step sooner, ensuring that $\gcd(m, n)$ is in Y at the end. So for all inputs m, n, this procedure terminates for the same reason as the Euclidean algorithm: the contents, y, of register Y is a natural number-valued variable that strictly decreases each time around the loop.

We claim that invariant properties that can be used to prove partial correctness are:

[4] This procedure is adapted from Aho, Hopcroft, and Ullman's text on algorithms.

- (a) $\gcd(X, Y) = \gcd(m, n)$,
- (b) $Km + Ln = Y$, and
- (c) $Um + Vn = X$.

To verify these invariants, note that invariant (a) is the same one we observed for the Euclidean algorithm. To check the other two invariants, let x, y, k, l, u, v be the contents of registers X, Y, K, L, U, V at the start of the loop and assume that all the invariants hold for these values. We must prove that (b) and (c) hold (we already know (a) does) for the new contents x', y', k', l', u', v' of these registers at the next time the loop is started.

Now according to the procedure, $u' = k, v' = l, x' = y$, so invariant (c) holds for u', v', x' because of invariant (b) for k, l, y. Also, $k' = u - qk, l' = v - ql, y' = x - qy$ where $q = \text{quotient}(x, y)$, so

$$k'm + l'n = (u - qk)m + (v - ql)n = um + vn - q(km + ln) = x - qy = y',$$

and therefore invariant (b) holds for k', l', y'.

Also, it's easy to check that all three invariants are true just before the first time around the loop. Namely, at the start $X = m, Y = n, K = 0, L = 1$ so $Km + Ln = 0m + 1n = n = Y$ so (b) holds; also $U = 1, V = 0$ and $Um + Vn = 1m + 0n = m = X$ so (c) holds. So by the Invariant Theorem, they are true at termination. But at termination, the contents, Y, of register Y divides the contents, X, of register X, so invariants (a) and (b) imply

$$\gcd(m, n) = \gcd(X, Y) = Y = Km + Ln.$$

So we have the gcd in register Y and the desired coefficients in K, L.

6 Derived Variables

The preceding termination proofs involved finding a natural-number-valued measure to assign to states. We might call this measure the "size" of the state. We then showed that the size of a state decreased with every state transition. By the Least Number Principle, the size can't decrease indefinitely, so when a minimum size state is reached, there can't be any transitions possible: the process has terminated.

More generally, the technique of assigning values to states — not necessarily natural numbers and not necessarily decreasing under transitions — is often useful in the analysis of algorithms. *Potential functions* play a similar role in physics. In the context of computational processes, such value assignments for states are called *derived variables*.

For example, for the Die Hard machines we could have introduced a derived variable, $f : Q \to \mathbb{R}$, for the amount of water in both buckets, by setting $f((a, b)) ::= a + b$. Similarly, in the robot problem, the position of the robot along the x-axis would be given by the derived variable $x\text{-coord}((i, j)) ::= i$.

We can formulate our general termination method as follows:

Definition 6.1. A derived variable $f : Q \to \mathbb{R}$ is *strictly decreasing* iff

$$q \to q' \text{ implies } f(q') < f(q).$$

Theorem 6.2. *If $f : Q \to \mathbb{N}$ is a strictly decreasing derived variable of a state machine, then the length of any execution starting at a start state q is at most $f(q)$.*

Of course we could prove Theorem 6.2 by induction on the value of $f(q)$. But think about what it says: "If you start counting down at some natural number $f(q)$, then you can't count down more than $f(q)$ times." Put this way, the theorem is so obvious that no one should feel deprived that we are not writing out a proof.

Corollary 6.3. *If there exists a strictly decreasing natural-number-valued derived variable for some state machine, then every execution of that machine terminates.*

We now define some other useful flavors of derived variables taking values over posets. It's useful to generalize the familiar notations $\leq$ and $<$ for ordering the real numbers: if $\preceq$ is a partial order on some set A, then define $\prec$ by the rule

$$a \prec a' \quad ::= \quad a \preceq a' \wedge a \neq a'.$$

A relation like $\prec$ is called a *strict* partial order. It is transitive, antisymmetric, and but *non*reflexive in the strongest sense: $a \not\prec a$ for every $a \in A$.[5]

Definition 6.4. Let $\preceq$ be partial order on a set, A. A derived variable $f : Q \to A$ is *strictly decreasing* iff

$$q \to q' \text{ implies } f(q') \prec f(q).$$

It is *weakly decreasing* iff

$$q \to q' \text{ implies } f(q') \preceq f(q).$$

Strictly increasing and *weakly increasing* derived variables are defined similarly.[6]

The existence of a natural-number-valued *weakly* decreasing derived variable does not guarantee that every execution terminates. That's because an infinite execution could proceed through states in which a weakly decreasing variable remained constant.

Predicates can be viewed as the special case of derived variables that only take the values 0 and 1. If we do this, then invariants can be characterized precisely as *weakly increasing* 0-1-valued derived variables. Namely, for any predicate, P, on states, define a derived variable, f_P, as follows:

$$f_P(q) ::= \begin{cases} 0 & \text{if } P(q) \text{ is false,} \\ 1 & \text{otherwise.} \end{cases}$$

Now P is an invariant if and only if f_P is weakly increasing. [7]

7 The Stable Marriage Problem

Okay, frequent public reference to derived variables may not help your mating prospects. But they can help with the analysis!

[5] In other words, if $a \prec b$, then it is not the case that $b \prec a$. This property is also called *a*symmetry.

[6] Weakly increasing variables are often also called *nondecreasing*. We will avoid this terminology to prevent confusion between nondecreasing variables and variables with the much weaker property of *not* being a decreasing variable.

[7] It may seem natural to call a variable whose values do not change under state transitions an "invariant variable." We will avoid this terminology, because the weakly increasing variable f_P associated with an invariant predicate, P, is not necessarily an invariant variable.

7.1 The Problem

Suppose that there are n boys and n girls. Each boy ranks all of the girls according to his preference, and each girl ranks all of the boys. For example, Bob might like Alice most, Carol second, Hildegard third, etc. There are no ties in anyone's rankings; Bob cannot like Carol and Hildegard equally. Furthermore, rankings are known at the start and stay fixed for all time.

The general goal is to marry off boys and girls so that everyone is happy; we'll be more precise in a moment. Every boy must marry exactly one girl and vice-versa—no polygamy.

If we want these marriages to last, then we want to avoid an unstable arrangement:

Definition 7.1. A set of marriages is *unstable* if there is a boy and a girl who prefer each other to their spouses.

For example, suppose that Bob is married to Carol, and Ted is married to Alice. Unfortunately, Carol likes Ted more than Bob *and* Ted likes Carol more than Alice. The situation is shown in Figure 2. So Carol and Ted would both be happier if they ran off together. We say that Carol and Ted are a *rogue couple*, because this is a situation which encourages roguish behavior.

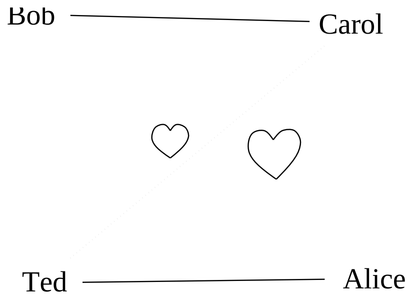


Figure 2: *Bob is married to Carol, and Ted is married to Alice, but Ted prefers Carol his mate, and Carol prefers Ted to her mate. So Ted and Carol form a rogue couple, making their present marriages unstable.*

Definition 7.2. A set of marriages is *stable* if there are no rogue couples or, equivalently, if the set of marriages is not unstable.

Now we can state the *Stable Marriage Problem* precisely: find spouses for everybody so that the resulting set of marriages is stable. It is not obvious that this goal is achievable! In fact, in the gender blind *Stable Buddy* version of the problem, where people of any gender can pair up as buddies, there may be no stable pairing! However, for the "boy-girl" marriage problem, a stable set of marriages does always exist.

Incidentally, although the classical "boy-girl-marriage" terminology for the problem makes some of the definitions easier to remember (we hope without offending anyone), solutions to the Stable Marriage Problem are really useful. The stable marriage algorithm we describe was first published in a paper by D. Gale and L.S. Shapley in 1962. At the time of publication, Gale and Shapley were unaware of perhaps the most impressive example of the algorithm. It was used to assign residents to hospitals by the National Resident Matching Program (NRMP) that actually predates their paper by ten years. Acting on behalf of a consortium of major hospitals (playing the role of the girls), the NRMP has, since the turn of the twentieth century, assigned each year's pool of medical school graduates (playing the role of boys) to hospital residencies (formerly called "internships"). Before the procedure ensuring stable matches was adopted, there were chronic disruptions and awkward countermeasures taken to preserve assignments of graduates to residencies. The stable marriage procedure was discovered and applied by the NRMP in 1952. It resolved the problem so successfully, that the procedure continued to be used essentially without change at least through 1989.[8]

Let's find a stable set of marriages in one possible situation, and then try to translate our method to a general algorithm. The table below shows the preferences of each girl and boy in decreasing order.

$boys$	$girls$
$1 : CBEAD$	$A : 35214$
$2 : ABECD$	$B : 52143$
$3 : DCBAE$	$C : 43512$
$4 : ACDBE$	$D : 12345$
$5 : ABDEC$	$E : 23415$

How about we try a "greedy" strategy?[9] We simply take each boy in turn and pack him off with his favorite among the girls still available. This gives the following assignment.

$$\begin{aligned}1 &\to C\\ 2 &\to A\\ 3 &\to D\\ 4 &\to B\\ 5 &\to E\end{aligned}$$

To determine whether this set of marriages is stable, we have to check whether there are any rogue couples. Boys 1, 2, and 3 all got their top pick among the girls; none would even think of running off. Boy 4 may be a problem because he likes girl A better than his mate, but she ranks him dead last. However, boy 4 also likes girl C better than his mate, and she rates him above her own mate. Therefore, boy 4 and girl C form a rogue couple! The marriages are not stable. We could try to make ad hoc repairs, but we're really trying to develop a general strategy.

Another approach would be to use induction. Suppose we pair boy 1 with his favorite girl, C, show that these two will not join a rogue couple, and then solve the remaining problem by induction. Clearly boy 1 will never leave his top pick, girl C. But girl C might very well dump him– she might even rate him last!

[8] Much more about the Stable Marriage Problem can be found in the very readable mathematical monograph by Dan Gusfield and Robert W. Irving, The Stable Marriage Problem: Structure and Algorithms, MIT Press, Cambridge, Massachusetts, 1989, 240 pp.

[9] "Greedy" is not any moral judgment. It is an algorithm classification that you can learn about in an Algorithms course like 6.046.

This turns out to be a tricky problem. The best approach is to use a mating ritual that is reputed to have been popular in some mythic past.

7.2 The Mating Algorithm

We'll describe the algorithm as a Mating Ritual that takes place over several days. The following events happen each day:

Morning: Each girl stands on her balcony. Each boy stands under the balcony of his favorite among the girls on his list, and he serenades her. If a boy has no girls left on his list, he stays home and does his 6.042 homework.

Afternoon: Each girl who has one or more suitors serenading her, says to her favorite suitor, "Maybe . . . , come back tomorrow." To the others, she says, "No. I will never marry you! Take a hike!"

Evening: Any boy who is told by a girl to take a hike, crosses that girl off his list.

Termination condition: When every girl has at most one suitor, the ritual ends with each girl marrying her suitor, if she has one.

There are a number of facts about this algorithm that we would like to prove:

- The algorithm terminates.
- Everybody ends up married.
- The resulting marriages are stable.

Furthermore, we would like to know if the algorithm is fair. Do both boys and girls end up equally happy?

7.3 The State Machine Model

Before we can prove anything, we should have clear mathematical definitions of what we're talking about. In this section we describe a state machine model of the Marriage Problem, and show how to give precise mathematical definitions of concepts we need to explain it, *e.g.*, who serenades who, who's a favorite suitor, *etc*. It's probably a good idea to skim the rest of this section and refer back to it only if a completely precise definition is needed.

[Optional] So let's begin by defining the problem.

Definition 7.3. A Marriage Problem consists of two disjoint sets of size $n \in \mathbb{N}$ called the-Boys and the-Girls. The members of the-Boys are called *boys*, and members of the-Girls are called *girls*. For each boy, B, there is a strict total order, $<_B$, on the-Girls, and for each girl, G, there is a strict total order, $<_G$, on the-Boys.

The idea is that $<_B$ is boy B's preference ranking of the girls. That is, $G_1 <_B G_2$ means that B prefers girl G_2 to girl G_1. Similarly, $<_G$ is girl G's preference ranking of the boys.

Next we model the Mating Algorithm with a state machine (Q, Q_0, δ). A key observation is that to determine what happens on any day of the ritual, all we need to know is which girls are on which boys' lists that day. So we define the boys' lists to be the states of our machine. Formally,

$$Q ::= \ [\text{the-Boys} \to \mathcal{P}(\text{the-Girls})]$$

where [the-Boys $\to \mathcal{P}$(the-Girls)] is the set of all total functions mappings boys to sets of girls. Here the idea is that if q : the-Boys $\to \mathcal{P}$(the-Girls) is a state, then $q(B)$ is the set of girls that boy B has left on his list, that is, the girls he has *not* crossed off yet.

We start the Mating Algorithm with no girls crossed off. So in the start state, q_0, every girl is on every boy's list. Formally,

$$\begin{aligned} q_0(B) &::= \text{ the-Girls}, && \text{for all boys, } B, \\ Q_0 &::= \{q_0\}. \end{aligned}$$

According to the Mating ritual, on any given morning, a boy will serenade the girl he most prefers among those he has not as yet crossed out. If he has crossed out all the girls, then we'll say he is serenading none, where none can be any convenient mathematical object that is not equal to any girl or boy. So we can define the derived variable *serenading*$_q$ to be a function mapping each boy to the girl (or none) he is serenading in state q. There is a simple way to define *serenading*$_q$: the-Boys $\to$ the-Girls $\cup$ {none} using the max operator for the boy's preference order. Namely,

$$\mathit{serenading}_q(B) ::= \begin{cases} \max_B q(B) & \text{if } q(B) \neq \emptyset, \\ \mathsf{none} & \text{otherwise.} \end{cases}$$

where $\max_B(S)$ is the maximum element in a nonempty set, $S \subseteq$ the-Girls ordered by $<_B$.

Another useful derived variable is the set of suitors who are serenading a girl on a given morning. That is, the derived variable is the function *suitors*$_q$: the-Girls $\to \mathcal{P}$(the-Boys) mapping each girl to her set of suitors:

$$\mathit{suitors}_q(G) ::= \{B \in \text{the-Boys} \mid G = \mathit{serenading}_q(B)\}.$$

The final derived variable we need is the favorite suitor of each girl on a given evening. That is, *favorite*$_q$: the-Girls $\to$ the-Boys $\cup$ {none} is defined by the rule:

$$\mathit{favorite}_q(G) ::= \begin{cases} \max_G \mathit{suitors}_q(G) & \text{if } \mathit{suitors}_q(G) \neq \emptyset, \\ \mathsf{none} & \text{otherwise.} \end{cases}$$

Notice that no boy is the favorite of two girls, because a boy serenades only one girl at a time.

Now we're ready to define the transitions. A state changes in the evening when some of the boys cross the girl they are serenading off their lists. If a boy is serenading a girl and he is not her favorite suitor, then he crosses her off. If the boy is not serenading any girl (because his list is empty), or if he is the current favorite of the girl he is serenading, then he doesn't cross anyone off his list. So for any state, q, this uniquely defines tomorrow morning's state, *next*$_q$.[10]

Notice that *next*$_q$ is always defined, so we still have to say when to terminate. The rule is to terminate when every girl has at most one suitor – that's who she will marry. But this is exactly the situation when it is no longer possible for any boy to cross any girl off his list, which means that *next*$_q = q$. So we define the transition relation by the rule

$$q \to q' \text{ if and only if } [q' = \mathit{next}_q \text{ and } q \neq q'].$$

There's one very useful fact to notice about the ritual: if a girl has a favorite boy suitor on some morning of the ritual, then that favorite suitor will still be serenading her the next morning — because his list won't have changed. So she is sure to have today's favorite boy among her suitors tomorrow. That means she will be able to choose a favorite suitor tomorrow who is at least as desirable to her as today's favorite.

It's helpful if we agree to include none as a least preferred object by every boy and girl. This allows us to say that, even if a girl has no suitor today, she will be no worse off tomorrow. So day by day, her favorite suitor can stay the same or get better, never worse. In others words, we have just proved the following:

[10] The preceding description of *next*$_q$ in words is a good mathematical definition. But for people who prefer formulas, here's one:

$$\mathit{next}_q(B) ::= \begin{cases} q(B) - \mathit{serenading}_q(B) & \text{if } \mathit{serenading}_q(B) \neq \mathsf{none} \text{ and } B \neq \mathit{favorite}_q(\mathit{serenading}_q(B)), \\ q(B) & \text{otherwise.} \end{cases}$$

Lemma 7.4. $\text{favorite}_q(G)$ *is a weakly* $<_G$*-increasing derived variable, for every girl,* G*.*

Similarly, a boy keeps serenading the girl he most prefers among those on his list until he must cross her off, at which point he serenades the next most preferred girl on his list. So we also have:

Lemma 7.5. $\text{serenading}_q(B)$ *is a weakly* $<_B$*-decreasing derived variable, for every boy,* B*.*

7.4 Termination

It's easy to see why the Mating Algorithm terminates: every day at least one boy will cross a girl off his list. If no girl can be crossed off any list, then the ritual has terminated. But initially there are n girls on each of the n boys' lists for a total of n^2 list entries. Since no girl ever gets added to a list, the total number of entries on the lists decreases every day that the ritual continues, and so the ritual can continue for at most n^2 days.[11]

7.5 They All Live Happily Every After . . .

We still have to prove that the Mating Algorithm leaves everyone in a stable marriage. One simple invariant predicate, P, captures what's going on:

> For every girl, G, and every boy, B, if G is crossed off B's list, then G has a favorite suitor and she prefers him over B.[12]

Why is P invariant? Well, we know from Lemma 7.4 that G's favorite tomorrow will be at least as desirable as her favorite today, and since her favorite today is more desirable than B, tomorrow's favorite will be too.

Notice that P also holds on the first day, since every girl is on every list. So by the Invariant Theorem, we know that P holds on every day that the Mating ritual runs. Knowing the invariant holds when the Mating Algorithm terminates will let us complete the proofs.

Theorem 7.6. *Everyone is married by the Mating Algorithm.*

[11] Here's the version with formulas. Define the derived variable *total-girls-names* of a state, q, to be the total number of list entries:

$$\textit{total-girls-names}(q) ::= \sum_{B \in \text{the-Boys}} |q(B)| \,.$$

Then *total-girls-names* is a strictly decreasing natural-number-valued derived variable whose initial value, *total-girls-names*(q_0), is

$$\sum_{B \in \text{the-Boys}} |\text{the-Girls}| = |\text{the-Boys}| \cdot |\text{the-Girls}| = n^2.$$

So by Theorem 6.2, the state machine terminates in at most n^2 steps.

[12] The formula would be:

$$P(q) ::= \quad [\forall B \in \text{the-Boys}\, \forall G \in \text{the-Girls}\; G \notin q(B) \longrightarrow B <_G \textit{favorite}_q(G)],$$

with $<_G$ extended to the-Boys $\cup$ $\{\texttt{none}\}$ by the rule that $\texttt{none} <_G B$ for all boys, B.

Proof. Suppose, for the sake of contradiction, that some boy is not married on the last day of the Mating ritual. So he can't be serenading anybody, that is, his list must be empty. So by invariant P, every girl has a favorite boy whom she prefers to that boy. In particular, every girl has a favorite boy that she marries on the last day. So all the girls are married. What's more there is no bigamy: we know that no two girls have the same favorite.

But there are the same number of girls as boys, so all the boys must be married too. □

Theorem 7.7. *The Mating Algorithm produces stable marriages.*

Proof. Let Bob be some boy and Carole some girl that he does *not* marry on the last day of the Mating ritual. We will prove that Bob and Carole are not a rogue couple. Since Bob was an arbitrary boy, it follows that no boy is part of a rogue couple. Hence the marriages on the last day are stable.

To prove the claim, we consider two cases:

Case 1. Carole is not on Bob's list. Then since invariant P holds, we know that Carole prefers her husband to Bob. So she's not going to run off with Bob: the claim holds in this case.

Case 2. Otherwise, Carole is on Bob's list. But since Bob is not married to Carole, he must have chosen to serenade his wife instead of Carole, so he must prefer his wife. So he's not going to run off with Carole: the claim also holds in this case. □

7.6 . . . And the Boys Live Especially Happily

Who is favored by the Mating Algorithm, the boys or the girls? The girls seem to have all the power: they stand on their balconies choosing the finest among their suitors and spurning the rest. What's more, their suitors can only change for the better as the Algorithm progresses (Lemma 7.4). And from the boy's point of view, the girl he is serenading can only change for the worse (Lemma 7.5). Sounds like a good deal for the girls.

But it's not! The fact is that from the beginning the boys are serenading their first choice girl, and the desirability of the girl being serenaded decreases only enough to give the boy his most desirable possible spouse. So the mating algorithm does as well as possible for all the boys and actually does the worst possible job for the girls.

To explain all this we need some definitions. Let's begin by observing that while the mating algorithm produces one set of stable marriages, there may be other ways to arrange stable marriages among the same set of boys and girls. For example, reversing the roles of boys and girls will often yield a different set of stable marriages among them.

Definition 7.8. If there is some stable set of marriages in which a girl, G, is married to a boy, B, then G is said to be a *possible spouse* for B, and likewise, B is a *possible spouse* for G.

This captures the idea that incompetent nerd Ben Bitdiddle has no chance of marrying movie star Heather Graham: she is not a possible spouse. No matter how the cookie crumbles, there will always be some guy that she likes better than Ben and who likes her more than his own mate. No marriage of Ben and Heather can be stable.

Note that since the mating algorithm always produces one stable set of marriages, the set of possible spouses for a person — even Ben — is never empty.

Definition 7.9. A person's *optimal* spouse is the possible spouse that person most prefers. A person's *pessimal* spouse is the possible spouse that person least prefers.

Here is the shocking truth about the Mating Algorithm:

Theorem 7.10. *The Mating Algorithm marries every boy to his optimal mate and every girl to her pessimal mate.*

Proof. The proof is in two parts. First, we show that every boy is married to his optimal mate. The proof is by contradiction.

Assume for the purpose of contradiction that some boy does not get his optimal girl. There must have been a day when he crossed off his optimal girl — otherwise he would still be serenading her or some even more desirable girl.

By the Least Number Principle, there must be a first day when a boy crosses off his optimal girl. Let B be one of these boys who first crosses off his optimal girl, and let G be B's optimal girl.

According to the rules of the ritual, B crosses off G because she has a favorite suitor, B', whom she prefers to B. So on the morning of the day that B crosses off G, her favorite B' has not crossed off his own optimal mate. This means that B' must like G more than any other possible spouse. (Of course, she may not be a possible spouse; she may dump him later.)

Since G is a possible spouse for B, there must be a stable set of marriages where B marries G, and B' marries someone else. But B' and G are a rogue couple in this set of marriages: G likes B' more than her mate, B, and B' likes G more than the possible spouse he is married to. This contradicts the assertion that the marriages were stable.

Now for the second part of the proof: showing that every girl is married to her pessimal mate. Again, the proof is by contradiction.

Suppose for the purpose of contradiction that there exists a stable set of marriages, $\mathcal{M}$, where there is a girl, G, who fares worse than in the Mating Algorithm. Let B be her spouse in the Mating Algorithm. By the preceding argument, G is B's optimal spouse. Let B' be her spouse in $\mathcal{M}$, a boy whom she likes even less than B. Then B and G form a rogue couple in $\mathcal{M}$: B prefers G to his mate, because she is optimal for him, and G prefers B to her mate, B', by assumption. This contradicts the assertion that $\mathcal{M}$ was a stable set of marriages. □

8 Well-Founded Orderings and Termination

8.1 Another Robot

Suppose we had a robot positioned at a point in the plane with natural number coordinates, that is, at an integer lattice-point in the Northeast quadrant of the plane. At every second the robot must move a unit distance South or West until it can no longer move without leaving the quadrant. It may also jump *any* integer distance East, but at every point in its travels, the number of jumps East is not allowed to be more than twice the number of previous moves South.

For example, suppose the robot starts at the position (9,8). It can now move South to (9,7) or West to (8,8); it can't jump East because there haven't been any previous South moves.

The robot's moves might continue along the following trajectory: South to (9,7), East to (23,7), South to (23,6), East to (399,6), West to (398,6), East to (511,6), West to (510,6), and East to $(10^5, 6)$. At this point it has moved South twice and East four times, so it can't jump East again until it makes another move South.

Claim 8.1. *The robot will always get stuck at the origin.*

If we think of the robot as a nondeterministic state machine, then Claim 8.1 is a termination assertion. The Claim may seem obvious, but it really has a different character than the termination results for the algorithms we've considered so far. That's because, even knowing that the starting position was, (9,8), for example, there is no way to bound the total number of moves the robot can make before it gets stuck. So we will not be able to prove termination using the natural-number-valued decreasing variable method of Theorem 6.2. The robot can delay getting stuck at the origin for as many seconds as it wants; nevertheless, it can't avoid getting stuck eventually.

Does Claim 8.1 still seem obvious? Before reading further, it's worth thinking how you might prove it.

We will prove that the robot always gets stuck at the origin by generalizing the decreasing variable method, but with decreasing values that are more general than natural numbers. Namely, the traveling robot can be modeled with a state machine with states of the form $((x, y), s, e)$ where

- $(x, y) \in \mathbb{N}^2$ is the robot's position,
- s is the number of moves South the robot took to get to this position, and
- $e \leq 2s$ is the number of moves East the robot took to get to this position.

Now we define a derived variable size : States $\rightarrow \mathbb{N}^3$:

$$\text{size}(((x, y), s, e)) \;::=\; (y, 2s - e, x),$$

and we order "sizes" of states with the *lexicographic* order, $\preceq_{\text{lex}}$, on $\mathbb{N}^3$:

$$(k, l, m) \preceq_{\text{lex}} (k', l', m') \;::=\; k < k' \text{ or } (k = k' \text{ and } l < l') \text{ or } (k = k' \text{ and } l = l' \text{ and } m \leq m') \qquad (1)$$

Let's check that size is lexicographically decreasing. Suppose the robot is in state $((x, y), s, e)$.

- If the robot moves West it enters state $((x - 1, y), s, e)$, and

  $$\text{size}(((x - 1, y), s, e)) = (y, 2s - e, x - 1) \prec_{\text{lex}} (y, s - 2e, x) = \text{size}(((x, y), s, e)),$$

  as required.

- If the robot jumps East it enters a state $((z, y), s, e + 1)$ for some $z > x$. Now

  $$\text{size}(((z, y), s, e + 1)) = (y, 2s - (e + 1), z) = (y, 2s - e - 1, z),$$

  but since $2s - e - 1 < 2s - e$, the rule (1) implies that

  $$\text{size}(((z, y), s, e + 1)) = (y, 2s - e - 1, z) \prec_{\text{lex}} (y, 2s - e, x) = \text{size}(((x, y), s, e)),$$

  as required.

- If the robot moves South it enters state $((x, y-1), s+1, e)$, and

  $$\text{size}(((x, y-1), s+1, e)) = (y-1, 2(s+1)-e, x) \prec_{\text{lex}} (y, s-2e, x) = \text{size}(((x, y-1), s+1, e)),$$

  as required.

So indeed state-size is a decreasing variable under lexicographic order. But as we'll show in the next section, lexicographic order has the property of being *well-founded*, which means that it is impossible for a lexicographic-order valued variable to decrease an infinite number of times. That's just what we need to finish verifying Claim 8.1.

8.2 Well-founded Partial Orders

The natural number triples $\mathbb{N}^3$ happen to be *totally* ordered under lexicographic order, but it's useful to formulate the concept of well-foundedness in the more general setting of *partial* orders.

First we generalize coordinatewise and lexicographic partial order to pairs of elements from *any* partial orders, not just natural numbers.

Definition 8.2. Let $(P_1, \preceq_1)$ and $(P_2, \preceq_2)$ be posets. The *lexicographic partial order*, $\preceq_{\text{lex}}$, on $P_1 \times P_2$ is defined by the condition that

$$(p_1, p_2) \preceq_{\text{lex}} (q_1, q_2) \ ::= \quad p_1 \prec_1 q_1 \text{ or } (p_1 = q_1 \wedge p_2 \preceq_2 q_2).$$

The *coordinatewise partial order*, $\preceq_{\text{c}}$, on $P_1 \times P_2$ is defined by the condition that

$$(p_1, p_2) \preceq_{\text{c}} (q_1, q_2) \ ::= \quad (p_1 \preceq_1 q_1 \wedge p_2 \preceq_2 q_2).$$

By the way, our "partial order" terminology is justified: it's easy to verify that $(P_1 \times P_2, \preceq_{\text{lex}})$ and $(P_1 \times P_2, \preceq_{\text{c}})$ are both posets.

Of course, the state sizes for the robot in the previous section were triples not pairs, but we can always treat the triples $\mathbb{N}^3$ as pairs whose first element is a pair. In other words, treat (l, m, n) as though it was $((l, m), n)$. So we define $(l, m, n) \preceq_{\text{lex}} (l', m', n')$ to mean that $(l, m) \preceq_{\text{lex}} (l', m')$ and $n \leq n'$. Likewise for $\preceq_{\text{c}}$.

Note that it wouldn't make any difference if we broke $\mathbb{N}^3$ into pairs another way, *e.g.*, treating, (l, m, n) as though it was $(l, (m, n))$. We wind up either way with the same partial order on triples given in rule (1).

Definition 8.3. A poset $(P, \preceq)$ is *well-founded* iff every nonempty subset $S \subseteq P$ has a *minimal element*.

Remember, an element is minimal in a set when no other element in the set is less than it. Also remember that a minim*al* element need not be a minum*um* element, that is, it need not be $\preceq$ all the elements in the set.

For example, suppose S is the set of natural number pairs whose sum is positive, that is $S = \mathbb{N}^2 - \{0, 0\}$. Then S has two minimal elements under $\preceq_{\text{c}}$, namely, (1,0) and (0,1), but it has no minimum element.

What's important for us is that both lexicographic and coordinatewise partial orders on $P_1 \times P_2$ will be well-founded providing P_1 and P_2 are well-founded posets. Namely,

Lemma 8.4. *Suppose $(P_1, \preceq_1)$ and $(P_2, \preceq_2)$ are posets. Then*

1. *so are $(P_1 \times P_2, \preceq_{lex})$ and $(P_1 \times P_2, \preceq_c)$. Moreover,*
2. *if $(P_1, \preceq_1)$ and $(P_2, \preceq_2)$ are both well-founded, then so are $(P_1 \times P_2, \preceq_{lex})$ and $(P_1 \times P_2, \preceq_c)$.*
3. *if $(P_1, \preceq_1)$ and $(P_2, \preceq_2)$ are both totally ordered, then so is $(P_1 \times P_2, \preceq_{lex})$.*

Proof. Parts 1. and 3. follow straightforwardly from the definitions, and we leave them to the reader.

To prove part 2., suppose $\emptyset \neq S \subseteq P_1 \times P_2$. Then the set

$$S_1 ::= \{p_1 \in P_1 \mid (p_1, p_2) \in S \text{ for some } p_2 \in P_2\}$$

is a nonempty subset of P_1, and so has a $\preceq_1$-minimal element, m_1. This means the set

$$S_{12} ::= \{p_2 \in P_2 \mid (m_1, p_2) \in S\}$$

is a nonempty subset of P_2 and so has a $\preceq_2$-minimal element, m_2. We claim that (m_1, m_2) is a minimal element of S under *both* the coordinatewise and the lexicographic partial orders on $P_1 \times P_2$.

To check this, we consider any element $(n_1, n_2) \in S$ such that $(n_1, n_2) \preceq (m_1, m_2)$ and prove that $(n_1, n_2) = (m_1, m_2)$, where $\preceq$ may be either coordinatewise or lexicographic order.

Now we know that

$$n_1 \preceq_1 m_1$$

by definition of $\preceq$. Also $n_1 \in S_1$, so

$$m_1 \preceq_1 n_1$$

by definition of m_1. Hence,

$$n_1 = m_1.$$

This means that $n_2 \in S_{12}$, so

$$m_2 \preceq_2 n_2$$

by definition of m_2. But since $n_1 = m_1$, we have by definition of $\preceq$, that

$$n_2 \preceq_2 m_2,$$

proving that

$$n_2 = m_2,$$

as claimed. □

There is another helpful way to characterize well-founded partial orders:

Lemma 8.5. *A poset is well-founded iff it has no infinite decreasing chain.*

Saying that the poset $(P, \preceq)$ has no infinite decreasing chain means there is no infinite sequence $p_1, p_2, \ldots, p_n \ldots$ of elements in P such that

$$p_1 \succ p_2 \succ \cdots \succ p_n \ldots.$$

Here we're using the notation "$p \succ q$" to mean $q \prec p$. That's so we can read the decreasing chain left to right, as usual in English.

Proof. ($\leftarrow$) (By contradiction) If there was such an infinite decreasing sequence, then the set of elements in the sequence itself would be a nonempty subset without a minimal element.

($\rightarrow$) (By contradiction) Suppose $(P, \preceq)$ was not well-founded. So there is some set $S \subseteq P$ such that S has at least one element s_1, but S has no minimal element. In particular, since s_1 is not minimal, there must be *another* element $s_2 \in S$ such that $s_2 \prec s_1$. Similarly, since s_2 is not minimal, there must be still another element $s_3 \in S$ such that $s_3 \prec s_2$. Continuing in this way, we can construct an infinite decreasing chain $s_1 \succ s_2 \succ s_3 \ldots$ in S. □

In the previous section, we wanted to conclude that the robot always got stuck because the size of its state, which decreases at every step, could not decrease forever. This now follows from Lemma 8.5 and the fact that, by Lemma 8.4.2., the lexicographic order on $\mathbb{N}^3$ is a well-founded.

8.3 Terminating Games of Perfect Information

The idea of well-founded partial orders will allow us to prove a fundamental theorem about games like chess, checkers, or tic-tac-toe. These are games in which two players alternate moves that depend only on the visible board position or state of the game. Such games are technically called *two person games of perfect information* because the players know the complete state of the game. (Most card games are *not* games of perfect information because neither player can see the other's hand.)

Two person games of perfect information can be represented abstractly by their "game trees." The root of a game tree corresponds to the start position of the game. It has a child for each position after a possible starting move of the first player. Each of these children in turn has children (these are grandchildren of the root) corresponding to the positions of the game after possible moves by the second player, and so on.

For example, in the game tree for tic-tac-toe, the root has nine children corresponding to the nine boxes that the first player could mark with an "X." Each of these nodes has eight children corresponding to the eight remaining boxes which the second player could mark with an "O," and so on.

In general, each node corresponds to a state of the game, and its children correspond to the states which result from each of the possible moves by the player whose turn it is. The leaves of the tree correspond to situations where no moves are possible—the game is over. A path from the root to a leaf describes an individual game or *play* of the game. (In English, "game" can be used in two senses: we can say that chess is a game, and we can also play a game of chess. The first usage refers to the game tree, and the second usage refers to what we call a "play.")

Problem 4. How many children does the root of a game tree for chess have? What is the first level of the tree where two nodes on that level have different numbers of children?

In many games like tic-tac-toe or chess, a play may end with a draw, but it's slightly simpler to develop our game theory if we forbid draws (for example, we could rule that all draws count as wins for the second player). So we label the leaves of the game tree with `win` or `lose` indicating the outcome for the first player. (A win for one player is a loss for the other—no "everybody wins"-type games at MIT. `:-)`

A *strategy* for a player in a game specifies which move the player should make in any state in which it is that player's turn to move. A *winning* strategy ensures that the player will win no matter what moves the other player makes.

In tic-tac-toe for example, most elementary school children figure out strategies for both players that each ensure that the other player can never get "tic-tac-toe". If we count a play that ends without a tic-tac-toe as a win for the second player, then the strategy which prevents the first player from getting "tic-tac-toe" is a winning strategy for the second player. (Of course the first player can win if his opponent plays childishly, but not if the second player follows the winning strategy.)

Theorem 8.6. ***Fundamental Theorem for two-person games of perfect information****: For games in which every play is finite and ends in win or lose, there is a winning strategy for one of the players.*

Although Theorem 8.6 guarantees a winning strategy, it gives no clue which player may have it. For chess, no one knows which player has a winning strategy.

Problem 5. Why must every play of chess eventually end?

Familiar games like tic-tac-toe and chess have finite game-trees, but not all games do. For example, in the Class Problems for Friday, Week 5, a winning strategy is developed for a game in which a move can be made to any integer-valued point in the nonnegative quadrant of the plane. The game tree for this choose-a-pair game is infinite. In fact, the tree is infinitely "wide." Namely, there are nodes that have an infinite number of children. The root is such a node, since the first move can be made to any pair of natural numbers. There is no finite bound on the length of plays, so the tree is also infinitely "deep."

But even though the choose-a-pair tree is infinite, the Class Problem shows that every choose-a-pair play terminates. Abstractly, this means that although the tree has *longer and longer finite paths* down from the root, it has *no infinite paths* from the root.

Definition 8.7. A rooted tree is called *finite-path* iff it has no infinite paths away from the root.

A simple finite-path tree of unbounded depth is illustrated in Figure 3. It consists of a root with an infinite number of children, where each child is at the top of a finite chain of descendents one longer than the previous child's.

So the two-person games of perfect information in which every play eventually ends in win or lose are precisely those games whose game trees are finite-path. These are the games to which the Fundamental Theorem applies.

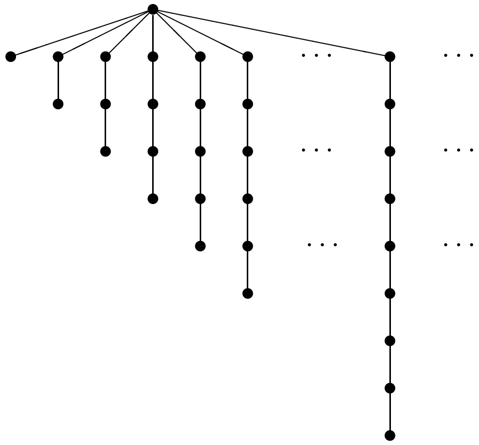


Figure 3: *A finite-path tree with paths of every finite length.*

Let $\prec$ be the binary relation on rooted trees "is (isomorphic to) a *proper* subtree of." More precisely, $T_1 \prec T_2$ iff there is a *non-root* vertex v of T_2 such that T_1 is isomorphic to the subtree of T_2 rooted at v.

Theorem 8.8. *The relation $\prec$ is a strict partial order on finite-path trees. Moreover, $\preceq$ is a* well-founded partial order*: every nonempty set of finite-path trees contains a minimal element.*

Assuming Theorem 8.8 for the moment, we can prove the Fundamental Theorem 8.6.

Proof. (of Theorem 8.6).

Suppose to the contrary that there is a game with a finite-path game tree that has no winning strategy. In particular, the set of finite-path game trees without winning strategies is nonempty, so by Theorem 8.8 there is a minimal such game tree under the "proper subtree" partial order.

Let T be a minimal tree without a winning strategy for either player. Now if C is a child of the root of T, then $C \prec T$ by definition. But each child, C, is also a game tree, and since T is minimal, there must be a winning strategy for one of the players in the game starting at the root of C.

Suppose some child, C_0, defines a game with a winning strategy for its second player. Then Player 1 on T has a winning strategy: choose C_0 on the first move, and then follow the second player's winning strategy on C_0. Since there is no such winning strategy on T, it must be that every child C defines a game with a winning strategy for the first player in the game starting at its root. But that means that the Player 2 on T has a winning strategy: if Player 1 on T picks C, then Player 2 will follow the winning strategy for the first player on C. This contradicts the hypothesis that T has no winning strategy. □

To finish the story, we have have to prove that the relation $\prec$ is a strict partial order on finite-path trees.

Proof. (of Theorem 8.8)

- The relation $\prec$ is transitive: Suppose a copy of T_1 is a subtree of T_2; then there is path, P, from the root of the T_1-copy to the root of T_2. Suppose also that a copy of T_2 is a subtree of T_3; then there is path, Q, from the root of the T_2-copy to the root of T_3. Now the T_2-copy has a copy of the T_1-copy as a subtree; call this subtree T_1'. So the path P has a copy, P', from the root of T_1' to the root of the T_2-copy. This means that the path P' followed by the path Q leads from the root of T_1' to the root of T_3. So indeed, there is a copy of T_1, namely T_1', which is a proper subtree of T_3.

- The relation $\prec$ is asymmetric: Suppose not. That is, suppose (a copy of) T_1 is a proper subtree of T_2 and also T_2 is a proper subtree of T_1. Then T_1 has a copy of itself as a proper subsubtree. Therefore there is a path—which must be of length two or more—from the root of T_1 to the root of the copy of T_1. This path extends to a path of length four or more to the root of the copy in the copy, and then extends to the root of the copy in the copy in the copy . . . . In this way we can find an infinite path from the root of T_1, contradicting the assumption that T_1 is finite-path.

- The relation $\prec$ is well-founded on finite-path trees.

  Suppose not. Then, by Lemma 8.5 there must be an infinite $\prec$-decreasing sequence of distinct elements. That is, an infinite sequence $T_1 \succ T_2 \succ T_3 \succ \dots$ of distinct finite-path trees. By definition, $T_n \succ T_{n+1}$ means there is a path from the root of T_n to a vertex which is the root of an isomorphic copy of T_{n+1}. It's convenient to assume that T_{n+1} itself—rather than an isomorphic copy—is the subtree of T_n . This assumption is justified because we could always replace T_{n+1} in the sequence above with the actual subtree of T_n isomorphic to it.

  Now $T_n \succ T_{n+1}$ implies there is a path P_n (necessarily of length one or more) from the root of T_n to the root of T_{n+1}. But then P_1 followed by P_2 followed by $P_3 \dots$ is an infinite path from the root of T_1, contradicting the hypothesis that T_1 is finite-path. Hence there cannot be such an infinite decreasing sequence of finite-path trees, and this implies that the set of finite-path trees partially ordered by $\prec$ is a well-founded poset.

□

Problem 6. Here is a generalization of the "choose-a-pair" game to "choose-a-tuple". The rules are:

Player 1 chooses any integer $n \geq 1$. Then Player 2 chooses any n-tuple of natural numbers. After that, the players alternate moves, choosing as a move any n-tuple, t, of natural numbers such that no previous move is $\preceq_c t$. A player wins when the other player chooses the origin (0, . . . ,0).

For example, Player 1 might begin by choosing $n = 3$. Then Player 2 might choose the 3-tuple (8, 9, 10). Possible subsequent choices might then be

$$(7, 8, 9), (0, 1, 67), (83, 0, 0), (1, 0, 0), (0, 0, 1)(0, 1, 0)$$

This finally leaves Player 1 with only the move (0,0,0), and the game now ends with his loss.

Prove that any choose-a-tuple game must end.

Recursive Definitions and Structural Induction

1 Recursive Definitions

Recursive definitions say how to build something from a simpler version of the same thing. They have two parts:

- Base case(s) that don't depend on anything else.
- Combination case(s) that depend on simpler cases.

Here are some examples of recursive definitions:

Example 1.1. Define a set, E, recursively as follows:

1. $0 \in E$,
2. if $n \in E$, then $n + 2 \in E$,
3. if $n \in E$, then $-n \in E$.

Using this definition, we can see that since $0 \in E$ by 1., it follows from 2. that $0 + 2 = 2 \in E$, and so $2 + 2 = 4 \in E$, $4 + 2 = 6 \in E$, . . . , and in fact any nonegative even number is in E. Also, by 3., $-2, -4, -6, \dots \in E$.

Is anything else in E? The definition doesn't say so explicitly, but an implicit condition on a recursive definition is that the only way things get into E is as a consequence of 1., 2., and 3. So clearly E is exactly the set of even integers.

Example 1.2. Define a set, S, of strings of a's and b's recursively as follows:

1. $\lambda \in S$, where λ is the *empty* string,
2. if $x \in S$, then $\mathsf{a}x\mathsf{b} \in S$,
3. if $x \in S$, then $\mathsf{b}x\mathsf{a} \in S$,
4. if $x, y \in S$, then $xy \in S$.

Here we're writing xy to indicate the *concatenation* of the strings x and y, namely, xy is the string that starts with the sequence of $\mathtt{a}$'s and $\mathtt{b}$'s in x followed by the $\mathtt{a}$'s and $\mathtt{b}$'s in y.

Using this definition, we can see that since $\lambda \in S$ by 1., it follows from 2. that $\mathtt{a}\lambda\mathtt{b} = \mathtt{ab} \in S$, so $\mathtt{aabb} \in S$ by 2. and $\mathtt{baba} \in S$ by 3. Likewise, $\mathtt{b}\lambda\mathtt{a} = \mathtt{ba} \in S$ by 3., so $\mathtt{abab} \in S$ by 2. and $\mathtt{bbaa} \in S$ by 3. Also, since $\mathtt{ab} \in S$ and $\mathtt{ba} \in S$, we have $\mathtt{abba} \in S$ as well as as $\mathtt{baab} \in S$ by 4.

Notice that every string in S has an equal number of $\mathtt{a}$'s and $\mathtt{b}$'s. This is easy to prove by induction on how a string gets to be in S.

Definition 1.3.

$$L ::= \{x \in \{\mathtt{a}, \mathtt{b}\}^* \mid \#\mathtt{a}\text{'s in } x = \#\mathtt{b}\text{'s in } x\}.$$

Lemma 1.4.

$$S \subseteq L.$$

Proof. Let $P(n)$ be the predicate that if any string, s, is in S as a consequence of n applications of rules 2., 3., and 4., then $s \in L$. We will prove that $P(n)$ holds for all $n \in \mathbb{N}$ by strong induction.

Base case ($n = 0$). The only way to get an $s \in S$ without using any of the rules 2., 3., and 4., is by rule 1., so $s = \lambda$ and s indeed has an equal number of $\mathtt{a}$'s and $\mathtt{b}$'s, namely zero of each.

Inductive step ($n \geq 0$). Assume $P(0), \ldots, P(n)$ to prove $P(n + 1)$.

So suppose $s \in S$ as a consequence of $n + 1$ applications of rules 2., 3., and 4.

Case 1: The $n + 1$st rule was 2. That is $s = \mathtt{a}x\mathtt{b}$ for some x that is in S as a consequence of n applications of 2., 3., and 4. By induction hypothesis, x has an equal number of $\mathtt{a}$'s and $\mathtt{b}$'s, say k of each. Then, s also has an equal number of $\mathtt{a}$'s and $\mathtt{b}$'s, namely $k + 1$ of each, so $s \in L$.

Case 2: The $n + 1$st rule was 2. Symmetric to Case 1.

Case 3: The $n + 1$st rule was 4. So $s = xy$ where x and y are each in S as a consequence of at most n applications of 2., 3., and 4. By induction hypothesis, x has the same number, say k_x, of $\mathtt{a}$'s and $\mathtt{b}$'s, and y has the same number, say k_y, of $\mathtt{a}$'s and $\mathtt{b}$'s. So xy has has an equal number of $\mathtt{a}$'s and $\mathtt{b}$'s, namely $k_x + k_y$ of each, and again $s \in L$ as required. □

It's also not hard to show that every string with an equal number of $\mathtt{a}$'s and $\mathtt{b}$'s is in S:

Problem 1. Prove that $L = S$.

Example 1.5. The set, F, of *Fully Bracketed Arithmetic Expressions in* x is a set of strings over the alphabet $\{\mathtt{]}, \mathtt{[}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{x}\}$ defined recursively as follows:

1. The symbols $\mathtt{0,1,x}$ are in F.
2. If e and e' are in F, then the string $\mathtt{[}e\mathtt{+}e'\mathtt{]}$ is in F,
3. if e and e' are in F then the string $\mathtt{[}e\mathtt{*}e'\mathtt{]}$ is in F,

4. if e is in F then the string **[-**e**]** is in F.

Several basic examples of recursively defined data types are based on *rooted trees*. These are possibly infinite directed trees, $T = (V_T, E_T)$, with a necessarily unique "root" vertex root(T), such that every vertex is reachable by a directed path from the root. Finite-Path Trees from Week 5 Notes, for example, are the class of rooted trees which do not have any infinite directed path from the root.

Another important class of trees are the *ordered binary trees*. These are possibly infinite rooted trees with labelled edges, such that at most two edges leave each vertex. If two edges leave a vertex, one is labelled `left` and the other is labelled `right`. If one edge leaves a vertex, it is labelled either `left` or `right`.

Example 1.6. We will define a special class of ordered binary trees called the *recursive ordered binary trees*, RecBinT:

1. If G is a graph with one vertex and no edges, then G is a RecBinT. That is, $(\{v\}, \emptyset) \in$ RecBinT and root$((\{v\}, \emptyset)) = v$.

2. If $T = (V, E)$ is in RecBinT, and $\mathbf{n}$ is a "new" node not in V, then the graph, makeleft(T), made by adding an edge labelled `left` from $\mathbf{n}$ to root(T) is a RecBinT. That is,

$$\text{makeleft}(T) ::= (V \cup \{\mathbf{n}\}, E \cup \{(\mathbf{n}, \text{root}(T), \texttt{left})\}) \in \text{RecBinT},$$

where (v, w, ℓ) is the directed edge from vertex v to vertex w with label ℓ.

3. Same as above, with "right" in place of "left."

4. If $T_1 = (V_1, E_1)$ and $T_2 = (V_2, E_2)$ are in RecBinT, V_1 and V_2 are disjoint, and $\mathbf{n}$ is a "new" node not in $V_1 \cup V_2$, then the graph, makeboth(T_1, T_2), made by adding an edge labelled `left` from $\mathbf{n}$ to root(T_1) and an edge labelled `right` from $\mathbf{n}$ to root(T_2) is a RecBinT. That is,

$$\text{makeboth}(T_1, T_2) ::= (V_1 \cup V_2 \cup \{\mathbf{n}\}, E_1 \cup E_2 \cup \{(\mathbf{n}, \text{root}(T_1), \texttt{left}), (\mathbf{n}, \text{root}(T_2), \texttt{right})\}) \in \text{RecBinT}.$$

These cases are illustrated in Figure 1, with edges labelled "left" shown going down to the left, and edges labelled "right" shown going down to the right.

A special case of RecBinT's are the *full* ordered binary trees, FullBinT. These are RecBinT's in which every node is either a *leaf* (*i.e.*, has out-degree zero) or has both a left and right subtree (see Figure 2). In other words rules 2. and 3. in the definition of RecBinT are not used in defining the subset FullBinT $\subset$ RecBinT.

Note that RecBinT is precisely the set of *finite* ordered binary trees. Properly speaking, we should prove this. Namely, we should prove that every RecBinT is finite, and prove that every finite ordered binary tree is an RecBinT. We'll hold off on this, but will do a more interesting proof of this kind for infinite trees in Theorem 3.2 below.

We can generalize RecBinT's to the class of *Countable Ordered Trees*, CT, in which each vertex may have any finite number, $n \geq 0$, of edges consecutively labelled $1, 2, \ldots, n$ leaving it, or may even have an infinite set of edges consecutively labelled with all the positive integers.

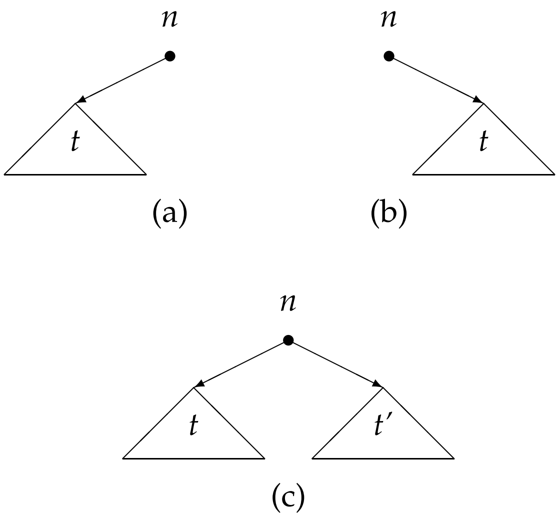


Figure 1: Building a binary tree: (a) makeleft(T), (b) makeright(T), and (c) makeboth(T_1, T_2).

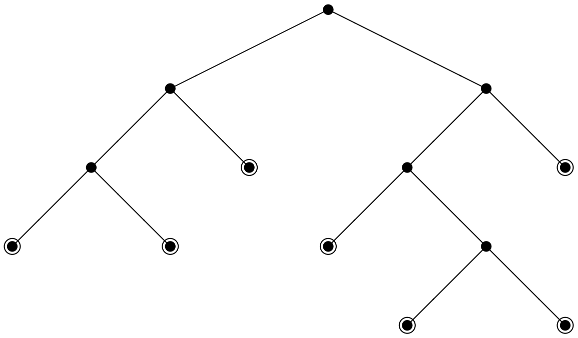


Figure 2: A full binary tree.

Example 1.7. We recursively define the set, RecCT, of *Recursive Countable Trees* as follows:

1. If G is a graph with one vertex and no edges, then G is a RecCT.
2. If $T_1 = (V_1, E_1), T_2 = (V_2, E_2), \dots$ is a finite or infinite sequence of RecCT's, such that V_i and V_j are disjoint for all $i \neq j$, and $\mathbf{n}$ is a "new" node not in $\bigcup_1^\infty V_i$, then the graph, maketree$(T_1, T_2, \dots)$, made by adding an edge labelled `i` from $\mathbf{n}$ to root(T_i) for all $i > 0$ is a RecCT. That is,

$$\text{maketree}(T_1, T_2, \dots) ::= (\bigcup_1^\infty V_i \cup \{\mathbf{n}\}, \bigcup_1^\infty E_i \cup \left\{(\mathbf{n}, \text{root}(T_i), i) \mid i \in \mathbb{Z}^+\right\}) \in \text{RecCT}.$$

Example 1.8. The set, List, of *pure lists* is defined recursively by:

1. The 0-tuple, (), is in List.
2. If ℓ_1 and ℓ_2 are in List, then the pair (ℓ_1, ℓ_2) is in List.

In Lisp-like programming languages, the pairing operation is called `cons` and the 0-tuple is called `nil`.

2 Structural Induction on Recursive Data Type Definitions

Structural induction is used to prove a property P of all the elements of some recursively-defined data type. The proof consists of two steps:

- Prove P for the "base cases" of the definition.
- Prove P for the result of any combination rule, assuming that it is true for all the parts.

For example, structural induction on Fully Bracketed Arithmetic Expressions takes the form:

Proof. To prove $\forall e \in F\ P(e)$, show:

Base ($e =$ `0`). $P($`0`$)$ holds.

Base ($e =$ `1`). $P($`1`$)$ holds.

Base ($e =$ `x`). $P($`x`$)$ holds.

Inductive step (`[`e`+`e'`]`). Assume $P(e)$ and $P(e')$ to prove $P($`[`e`+`e'`]`$)$.

Inductive step (`[`e`*`e'`]`). Assume $P(e)$ and $P(e')$ to prove $P($`[`e`*`e'`]`$)$.

Inductive step (`[-`e`]`). Assume $P(e)$ to prove $P($`[-`e`]`$)$. □

Here's an actual example:

Theorem 2.1. *Every Fully Bracketed Arithmetic Expression has the same number of left and right brackets.*

Proof. This is just like the proof of Lemma 1.4 above, except we proved Lemma 1.4 by ordinary strong induction explicitly on the number of rules used to contruct on element. Here we use structural induction, which lets us carry out the proof without having to count rule applications.

Define $P(e)$::= expression e has the same number of left brackets, `]`, and right brackets, `[`.

Base Cases ($e = \mathtt{0}$, $e = \mathtt{1}$, or $e = \mathtt{x}$). The expression e has no brackets.

Inductive step ($\mathtt{[}e\mathtt{+}e'\mathtt{]}$). Assume $P(e)$ and $P(e')$ to prove $P(\mathtt{[}e\mathtt{+}e'\mathtt{]})$.

But $P(e)$ implies e has k_e left brackets and k_e right brackets for some $k_e \in \mathbb{N}$; likewise e' has $k_{e'}$ left brackets and $k_{e'}$ right brackets. So $\mathtt{[}e\mathtt{+}e'\mathtt{]}$ has $k_e + k_{e'} + 1$ right brackets and the same number of left brackets.

Inductive step ($\mathtt{[}e\mathtt{*}e'\mathtt{]}$). Similar.

Inductive step ($\mathtt{[-}e\mathtt{]}$). Similar. □

3 Induction on Trees

Here's a proof using structural induction on binary trees:

Theorem 3.1. *The number of edges in any RecBinT is exactly one fewer than the number of vertices.*

Proof. Define

$$P(T) ::= \quad T = (V, E) \in \text{RecBinT and } |V| - 1 = |E|.$$

Base Case (T is the single-node tree). There are 0 edges and 1 node, so $P(T)$ holds.

Inductive step ($T = \text{makeleft}(S)$). Assume $P(S)$ to prove $P(T)$. That is, assuming

$$|V_S| - 1 = |E_S|, \tag{1}$$

show that

$$|V_T| - 1 = |E_T| \tag{2}$$

But

$$\begin{aligned} E_T =& E_S \cup \{(\mathbf{n}, \text{root}S, \mathtt{left})\} \qquad \text{so,} \\ |E_T| =& |E_S| + 1, \end{aligned} \tag{3}$$

and

$$\begin{aligned} |E_T| =& |V_S|, \qquad \text{(by (3) and (1)), so,} \\ |E_T| + 1 =& |V_S| + 1. \end{aligned} \tag{4}$$

But

$$\begin{aligned} V_T =& V_S \cup \{\mathbf{n}\}, \qquad \text{so} \\ |V_T| =& |V_S| + 1 \end{aligned} \tag{5}$$

Combining (5) and (4) immediately implies (2).

Inductive step ($T = \text{makeright}(S)$). Same as previous case with "right" replacing "left."

Inductive step ($T = \text{makeboth}(T_1, T_2)$). Assume $P(T_1)$ and $P(T_2)$ to prove $P(T)$. That is, assuming

$$|E_{T_i}| \quad = \quad |V_{T_i}| - 1, \tag{6}$$

for $i = 1, 2$, prove

$$|E_T| = |V_T| - 1, \tag{7}$$

The similar proof for this case is left to the reader. □

Structural induction is a correct and useful proof method on recursive data types even when they are infinite. For example, using structural induction on the definition of RecCT's, we can easily prove that every RecCT is a Finite-path Countable Ordered Tree.

Theorem 3.2. *Every RecCT is a Finite-Path CT.*

Proof. Define

$$P(T) ::= T \text{ is a Finite-path CT.}$$

We prove that $P(T)$ holds for all $T \in$ RecCT by Structural Induction on the definition of RecCT.

Base case (T has one vertex) T is a CT by definition of CT. There is only an "empty" of length zero in T. So T is also Finite-path.

Inductive step ($T = \text{maketree}(T_1, T_2, \ldots)$). By structural induction hypothesis, we may assume that $P(T_i)$ holds for all T_i.

By definition of T, the edges from the root of T are labelled with consecutive integers. Any other vertex of T is a vertex in V_i which is labelled with consecutive integers because $T_i \in$ CT by hypothesis. Hence $T \in$ CT.

Also by definition of T, the second node on any path from the root of T must be the root some T_i. The rest of the path is a directed path from the root of T_i, and so is finite because T_i is Finite-path by induction hypothesis. So the whole path from the root of T is also finite.

□

This proof by Structural Induction may seem obvious, but it is actually different in character from all the other Structural Induction proofs. Namely, all the other proofs could be reformulated as ordinary induction on the number of rule applications used in constructing an element of the recursive data type. But because a Recursive Countable Tree can be built by combining an infinite sequence of subtrees, which in total already take an infinite number of rule applications, the number of rule applications to construct a single RecCT may not be finite. So ordinary induction on the number of rule applications to construct a recursively defined object won't work. As a matter of fact, Mathematical Logicians have proved that this kind structural induction on infinite data objects is actually *strictly more powerful* than ordinary induction. But nevertheless, Structural

Induction is a completely correct proof method, even for data types like RecCT that build up elements in an infinite number of steps.

Problem 2. Prove the converse of Theorem 3.2:

Theorem 3.3. *Every Finite-path CT is a RCT.*

Hint: Suppose not, and consider a minimal Finite-path CT (under the well-founded "strict-subtree" partial order on Finite-path trees defined in Week 5 Notes) that is not in RecCT.

4 Recursively-defined Functions on Recursively-defined Data Types

4.1 Some Recursively Defined Functions

Recursive definitions provide a natural way to define functions whose domains are recursively-defined data types.

Example 4.1. (Binary Trees)

Define the function numnodes(T) (a recursive definition of $|V_T|$ for a RecBinT as follows:

1. numnodes(single node) $::= 1$,
2. numnodes(makeleft(S)) $::=$ numnodes$(S) + 1$,
3. numnodes(makeright(S)) $::=$ numnodes$(S) + 1$,
4. numnodes(makeboth(T_1, T_2)) $::=$ numnodes(T_1) $+$ numnodes$(T_2) + 1$.

Similarly, define numedges(T):

1. numedges(single node) $::= 0$,
2. numedges(makeleft(S)) $::=$ numedges$(S) + 1$,
3. numedges(makeright(S)) $::=$ numedges$(S) + 1$,
4. numedges(makeboth(T_1, T_2)) $::=$ numedges(T_1) $+$ numedges$(T_2) + 2$.

Example 4.2. (Arithmetic Expressions) Given a value, n, for the variable, x, we can easily calculate the numerical value of any expression, e, in the set, F, of Fully Bracketed Arithmetic Expressions in x. The function, eval(e, n), giving the value of expression e when x has value n has a simple recursive definition based on the definition of the recursive data type F. Namely, define

1. eval$(\mathtt{0}, n) ::= 0$,
2. eval$(\mathtt{1}, n) ::= 1$,

3. $\text{eval}(\texttt{x}, n) ::= n$,

4. $\text{eval}(\texttt{[}e\texttt{+}e'\texttt{]}, n) ::= \text{eval}(e, n) + \text{eval}(e', n)$,

5. $\text{eval}(\texttt{[}e\texttt{*}e'\texttt{]}, n) ::= \text{eval}(e, n) \cdot \text{eval}(e', n)$,

6. $\text{eval}(\texttt{[-}e\texttt{]}, n) = -\text{eval}(e, n)$.

Another useful operation on arithmetic expressions is substituting one into another. Let $\text{subst}(e, f)$ be the result of substituting expression f for all occurrences of $\texttt{x}$ in e. The function subst also has a simple definition based on the definition of the recursive data type F. Namely, define $\text{subst}(e, f)$ recursively in e as follows:

1. $\text{subst}(\texttt{0}, f) ::= 0$,

2. $\text{subst}(\texttt{1}, f) ::= 1$,

3. $\text{subst}(\texttt{x}, f) ::= f$,

4. $\text{subst}(\texttt{[}e\texttt{+}e'\texttt{]}, f) ::= \texttt{[}\text{subst}(e, f)\texttt{+}\text{subst}(e', f)\texttt{]}$,

5. $\text{subst}(\texttt{[}e\texttt{*}e'\texttt{]}, f) ::= \texttt{[}\text{subst}(e, f)\texttt{*}\text{subst}(e', f)\texttt{]}$,

6. $\text{subst}(\texttt{[-}e\texttt{]}, f) = \texttt{[-}\text{subst}(e, f)\texttt{]}$.

4.2 Recursive Function Definitions

In general, we can define a function, f, on a recursively defined data type by defining f on each of the elements in the base cases of the definition. Then define $f(d)$ in terms of $f(d_1), f(d_2), \ldots$ where d is an element built from elements $d_1, d_2, \ldots$ by a combination rule. One warning though: f is only guaranteed to be well-defined if each element, d, can be constructed in *only one way*: only by a unique combination rule applied to unique elements $d_1, d_2, \ldots$. So this kind of recursive definition will work fine for Fully Bracketed Arithmetic Expressions, binary trees, lists, and recursive trees, but might *not* yield a well-defined function if the definition of f was based on the recursive definition we gave for the set, S, of strings with an equal number of a's and b's. The reason is that some strings in S can be constructed in more than one way.

Problem 3. **(a)** What is the smallest string in S which can be constructed in two different ways using Definition 1.4 of S above?

(b) Find a recursive definition of S similar to the one above, but under which every string in S is constructed in exactly one way.

4.3 Proving Properties of Recursive Functions

When we have a recursive definition of a function on a recursive data type, we can use structural induction to prove properties of the function. We already did this in proving Theorem 3.1 relating the number of edges and nodes in a binary tree. A more interesting example using structural induction on the definition of Fully Bracketed Arithmetic Expressions to prove a fundamental relationship between numerical and symbolic calculation.

Namely, suppose we have an arithmetic expression e with variable x and we substitute another expression, f, for all the x in e to obtain a new expression, $e' ::= \text{subst}(e, f)$. Now suppose we want to evaluate e' when x has the value n. One way to do it would simply be to evaluate e' recursively, ignoring the way that e' was obtained from e and f. But another, usually more efficient, approach would be to find the value of f when x has the value n, and then evaluate e when x is given the value obtained from f. In Lisp programming terminology, the first approach corresponds to evaluation using a "substitution model," and the second approach corresponds to evaluation using an "environment model." We will prove that both models yield the same answer. More precisely, what we want to prove is

Theorem 4.3. *For all expressions $e, f \in F$ and $n \in \mathbb{Z}$,*

$$\textit{eval}(\textit{subst}(e, f), n) = \textit{eval}(e, \textit{eval}(f, n)). \tag{8}$$

Proof. The proof is by structural induction on e.

Base cases ($e = \mathtt{0}$, $e = \mathtt{1}$). Then the lefthand side of equation (8) equals e by the base cases of the definition of subst, and the righthand side equals e by the base cases of the definition of eval.

Base case ($e = \mathtt{x}$). Then the lefthand side of equation (8) equals $\text{eval}(f, n)$ by the base case for $\mathtt{x}$ of the definition of subst, and the righthand side equals $\text{eval}(f, n)$ by the base case for $\mathtt{x}$ of the definition of eval.

Inductive step ($\mathtt{[}e\mathtt{+}e'\mathtt{]}$). Assume that for all $f \in F$ and $n \in \mathbb{N}$,

$$\begin{aligned} \text{eval}(\text{subst}(e, f), n) &= \text{eval}(e, \text{eval}(f, n)) & (9) \\ \text{eval}(\text{subst}(e', f), n) &= \text{eval}(e', \text{eval}(f, n)), & (10) \end{aligned}$$

to prove that for all $f \in F$ and $n \in \mathbb{N}$,

$$\text{eval}(\text{subst}(\mathtt{[}e\mathtt{+}e'\mathtt{]}, f), n) = \text{eval}(\mathtt{[}e\mathtt{+}e'\mathtt{]}, \text{eval}(f, n)) \tag{11}$$

But the lefthand side of (11) equals

$$\text{eval}(\mathtt{[}\text{subst}(e, f)\mathtt{+}\text{subst}(e', f)\mathtt{]}, n)$$

by definition of subst for a sum expression, which equals

$$\text{eval}(\text{subst}(e, f), n) + \text{eval}(\text{subst}(e', f), n)$$

by definition of eval for a sum expression. By induction hypothesis, this equals

$$\text{eval}(e, \text{eval}(f, n)) + \text{eval}(e', \text{eval}(f, n)),$$

which equals the righthand side of (11) by definition of eval for a sum expression. This proves (11) in this case.

Inductive step ($\mathtt{[}e\mathtt{*}e'\mathtt{]}$). Similar.

Inductive step ($\mathtt{[-}e\mathtt{]}$). Similar. □

4.4 Recursive Functions on Natural Numbers

The recursive definitions of functions on recursively defined data types also applies to recursively defined functions on the natural numbers. One can think of the natural numbers, $\mathbb{N}$, as recursively defined by

1. $0 \in \mathbb{N}$,
2. if $n \in \mathbb{N}$, then $n + 1 \in \mathbb{N}$.

Now ordinary induction is exactly structural induction based on this recursive definition. Treating the natural numbers as a recursively defined data type also justifies the use of familiar recursive definitions of functions on the natural numbers.

The Factorial function: A very useful function for counting and probability.

- $0! ::= 1$
- $(n + 1)! ::= (n + 1)n!$ for $n \geq 0$.

The Fibonacci numbers: These are interesting numbers that arise, e.g., in biology, where they model some types of growth processes (plants, cells, rabbit populations, etc.). The Fibonacci numbers are written as F_i, $i = 0, 1, 2, \ldots$. They are defined recursively by:

$$\begin{aligned} F_0 &::= 0, \\ F_1 &::= 1, \\ F_i &::= F_{i-1} + F_{i-2} \text{ for } i \geq 2. \end{aligned}$$

Note there are two base cases, since each combination relies on previous two values.

What is F_4? Well, $F_2 = F_1 + F_0 = 1$, $F_3 = F_2 + F_1 = 2$, so $F_4 = 3$. The sequence starts out $0, 1, 1, 2, 3, 5, 8, 13, 21, \ldots$.

Σ notation (the traditional style): We've been using this notation informally already.

- $\Sigma_{i=1}^{0} f(i) ::= 0$,
- $\Sigma_{i=1}^{n+1} f(i) ::= \Sigma_{i=1}^{n} f(i) + f(n + 1)$, for $n \geq 0$.

Simultaneous recursive definitions: You can define several things at the same time, in terms of each other. For example, we may define two functions f and g from $\mathbb{N}$ to $\mathbb{N}$, recursively, by:

- $f(0) ::= 1$,
- $g(0) ::= 1$,
- $f(n + 1) ::= f(n) + g(n)$, for $n \geq 0$,
- $g(n + 1) ::= f(n) \times g(n)$, for $n \geq 0$.

We can use the recursive definitions of functions in proving their properties. As an illustration, we'll prove a cute identity involving Fibonacci numbers. Fibonacci numbers provide lots of fun for mathematicians because they satisfy many such identities.

Proposition 4.4. $\forall n \geq 0(\Sigma_{i=0}^{n} F_i^2 = F_n F_{n+1})$.

Example: $n = 4$:

$$0^2 + 1^2 + 1^2 + 2^2 + 3^2 = 15 = 3 \cdot 5.$$

Let's try a proof by (standard, not strong) induction. The theorem statement suggests trying it with $P(n)$ defined as:

$$\sum_{i=0}^{n} F_i^2 = F_n F_{n+1}.$$

Base case ($n = 0$). $\Sigma_{i=0}^{0} F_i^2 ::= (F_0)^2 = 0 = F_0 F_1$ because $F_0 ::= 0$.

Inductive step ($n \geq 0$). Now we stare at the gap between $P(n)$ and $P(n+1)$. $P(n+1)$ is given by a summation that's obtained from that for $P(n)$ by adding one term; this suggests that, once again, we subtract. The difference is just the term F_{n+1}^2. Now, we are assuming that the original $P(n)$ summation totals $F_n F_{n+1}$ and want to show that the new $P(n+1)$ summation totals $F_{n+1} F_{n+2}$. So we would *like* the difference to be

$$F_{n+1} F_{n+2} - F_n F_{n+1}.$$

So, the actual difference is F_{n+1}^2 and the difference we want is $F_{n+1} F_{n+2} - F_n F_{n+1}$. Are these the same? We want to check that:

$$F_{n+1}^2 = F_{n+1} F_{n+2} - F_n F_{n+1}.$$

But this is true, because it is really the Fibonacci definition in disguise: to see this, divide by F_{n+1}.

5 Ill-formed Definitions

We must take care that functions defined recursively are well-defined. Below are some function specifications that look like definitions but aren't.

$f(n) = 2 + f(n-1)$**.** This "definition" has no base case. If some function, f, satisfied this equatioin, so would a function obtained by adding a constant, k, to the value of f. So this "definition" does not uniquely define f.

$f(n) = 0$ *if n is divisible by 2,* $f(n) = 1$ *if n is divisible by 3, and* $f(n) = 2$ *otherwise.* This "definition" is inconsistent: it requires $f(6) = 0$ and $f(6) = 1$, so it doesn't define anything.

$f(0) = 0$*, otherwise* $f(n) = f(n+1) + 1$. From this "definition," it follows that $f(1) > f(2) > f(3) > \ldots$, so $f(1)$ cannot equal any integer. No total function on the natural numbers can satisfy this definition. However, it does uniquely determine a *partial* function on the natural numbers, namely, the function that is 0 at 0 and undefined everywhere else.

$f(0) = 0$*, otherwise* $f(n) = f(n+1)$. Lots of total functions satisfy the equations in the "definition." Namely, any function that is 0 at 0 and constant eveywhere else.

A mystery. Mathematicians have been wondering about this one for a while:

- $f(1) = 1$.
- If n is even, then $f(n) = f(n/2)$.
- If n is odd, then $f(n) = f(3n + 1)$.

This "definition" of f in some cases defines $f(n)$ in terms of f applied to arguments larger than n, and so cannot be justified by induction on $\mathbb{N}$. It has been proven that if f is a function that satisfies the above equations, then $f(n) = 1$ for all n up to at least a billion, and it's a good guess that the only f that works is the constant function equal to 1. But nobody knows if there is more than one function satisfying these equations.

6 Induction in Computer Science

We've spent a lot of time studying induction. This is because induction comes up all the time in analyzing computation. Why? Well, ordinary induction on natural numbers is a "one step at a time" proof method. Computations also evolve "one step at a time."

Structural induction on recursive definitions lets us go beyond simple natural number counting. We can explain how to define recursive functions on recursive data types. And we can prove properties of recursively defined data and functions by structural induction on the definition of the data type. We even noted that Structural Induction is technically more powerful than ordinary induction. It is a technique which every Computer Scientist should firmly grasp.

Massachusetts Institute of Technology
6.042J/18.062J, Fall '02: Mathematics for Computer Science
Professor Albert Meyer and Dr. Radhika Nagpal

Course Notes 7

Sums, Products & Asymptotics

1 Closed Forms and Approximations

Sums and products arise regularly in the analysis of algorithms and in other technical areas such as finance and probabilistic systems. We've already seen that

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}.$$

Having a simple *closed form* expression such as $n(n+1)/2$ makes the sum a lot easier to understand and evaluate. We proved by induction that this formula is correct, but not where it came from. In Section 4, we'll discuss ways to find such closed forms. Even when there are no closed forms exactly equal to a sum, we may still be able to find a closed form that *approximates* a sum with useful accuracy.

The product we focus on in these notes is the familiar factorial:

$$n! \ ::= \ 1 \cdot 2 \cdots (n-1) \cdot n = \prod_{i=1}^{n} i.$$

We'll describe a closed form approximation for it called *Stirling's Formula*.

Finally, when there isn't a good closed form approximation for some expression, there may still be a closed form that characterizes its growth rate. We'll introduce *asymptotic notation*, such as "big Oh", to describe growth rates.

2 Annuities

Would you prefer a million dollars today or \$50,000 a year for the rest of your life? On the one hand, instant gratification is nice. On the other hand, the total dollars received at \$50K per year is much larger if you live long enough.

Formally, this is a question about the value of an annuity. An *annuity* is a financial instrument that pays out a fixed amount of money at the beginning of every year for some specified number of years. In particular, an n-year, m-payment annuity pays m dollars at the start of each year for n years. In some cases, n is finite, but not always. Examples include lottery payouts, student loans, and home mortgages. There are even Wall Street people who specialize in trading annuities.

A key question is what an annuity is worth. For example, lotteries often pay out jackpots over many years. Intuitively, $\$50,000$ a year for 20 years ought to be worth less than a million dollars right now. If you had all the cash right away, you could invest it and begin collecting interest. But what if the choice were between $\$50,000$ a year for 20 years and a *half* million dollars today? Now it is not clear which option is better.

In order to answer such questions, we need to know what a dollar paid out in the future is worth today. To model this, let's assume that money can be invested at a fixed annual interest rate p. We'll assume an 8% rate[1] for the rest of the discussion.

Here is why the interest rate p matters. Ten dollars invested today at interest rate p will become $(1+p)\cdot 10 = 10.80$ dollars in a year, $(1+p)^2 \cdot 10 \approx 11.66$ dollars in two years, and so forth. Looked at another way, ten dollars paid out a year from now are only really worth $1/(1+p)\cdot 10 \approx 9.26$ dollars today. The reason is that if we had the $\$9.26$ today, we could invest it and would have $\$10.00$ in a year anyway. Therefore, p determines the value of money paid out in the future.

2.1 The Value of an Annuity

Our goal is to determine the value of an n-year, m-payment annuity. The first payment of m dollars is truly worth m dollars. But the second payment a year later is worth only $m/(1+p)$ dollars. Similarly, the third payment is worth $m/(1+p)^2$, and the n-th payment is worth only $m/(1+p)^{n-1}$. The total value V of the annuity is equal to the sum of the payment values. This gives:

$$V = \sum_{i=1}^{n} \frac{m}{(1+p)^{i-1}}.$$

To compute the real value of the annuity, we need to evaluate this sum. One way is to plug in m, n, and p, compute each term explicitly, and then add them up. However, this sum has a special closed form that makes the job easier. (The phrase "closed form" refers to a mathematical expression without any summation or product notation.) First, lets make the summation prettier with some substitutions.

$$\begin{aligned} V &= \sum_{i=1}^{n} \frac{m}{(1+p)^{i-1}} \\ &= \sum_{j=0}^{n-1} \frac{m}{(1+p)^{j}} \quad \text{(substitute } j = i-1\text{)} \\ &= m\sum_{j=0}^{n-1} x^j \quad \text{(substitute } x = \frac{1}{1+p}\text{).} \end{aligned}$$

The goal of these substitutions is to put the summation into a special form so that we can bash it with a theorem given in the next section.

[1] U.S. interest rates have dropped steadily for several years, and ordinary bank deposits now earn around 3%. But just a few years ago the rate was 8%; this rate makes some of our examples a little more dramatic. The rate has been as high as 17% in the past twenty years.

In Japan, the standard interest rate is near zero%, and on a few ocasions in the past few years has even been slightly negative. It's a mystery to U.S. economists why the Japanese populace keeps any money in their banks.

2.2 The Sum of a Geometric Series

Theorem 2.1. *For all $n \geq 1$ and all $x \neq 1$,*

$$\sum_{i=0}^{n-1} x^i = \frac{1 - x^n}{1 - x}.$$

The terms of the summation in this theorem form a *geometric series*. The distinguishing feature of a geometric series is that each term is a constant times the one before; in this case, the constant is *x*. The theorem gives a closed form for the sum of a geometric series that starts with 1.

Proof. The proof is by induction on n. Let $P(n)$ be the predicate that for all $x \neq 1$, $\sum_{i=0}^{n-1} x^i = (1 - x^n)/(1 - x)$. In the base case, $P(1)$ holds because $\sum_{i=0}^{0} x^i = x^0 = 1$ and $(1 - x^1)/(1 - x) = 1$.

In the inductive step, for $n \geq 1$ assume that for all $x \neq 1$, $\sum_{i=0}^{n-1} x^i = (1 - x^n)/(1 - x)$. We will use this to prove that for all $x \neq 1$, $\sum_{i=0}^{n} x^i = (1 - x^{n+1})/(1 - x)$.

$$\begin{aligned}
\sum_{i=0}^{n} x^i &= x^n + \sum_{i=0}^{n-1} x^i \\
&= x^n + \frac{1 - x^n}{1 - x} \\
&= \frac{x^n(1 - x) + 1 - x^n}{1 - x} \\
&= \frac{1 - x^{n+1}}{1 - x}.
\end{aligned}$$

The second line follows from the first by the induction hypothesis. The remaining steps are only simplifications. □

As if often the case, the proof by induction gives no hint about how the formula was found in the first place. Here is a more insightful derivation. The trick is to let S be the value of the sum and then observe what $-xS$ is:

$$\begin{array}{rcllllllll}
S &=& 1 & +x & +x^2 & +x^3 & + & \cdots & +x^{n-1} & \\
-xS &=& & -x & -x^2 & -x^3 & - & \cdots & -x^{n-1} & - x^n.
\end{array}$$

Adding these two equations gives:

$$\begin{aligned}
S - xS &= 1 - x^n, \text{ so} \\
S &= \frac{1 - x^n}{1 - x}.
\end{aligned}$$

We'll say more about finding (as opposed to just proving) summation formulas later.

2.3 Return of the Annuity Problem

Now we can solve the annuity pricing problem. The value of an annuity that pays m dollars at the start of each year for n years is computed as follows:

$$\begin{aligned} V &= m \sum_{j=0}^{n-1} x^j \\ &= m \frac{1 - x^n}{1 - x} \\ &= m \frac{1 - (\frac{1}{1+p})^n}{1 - \frac{1}{1+p}} \\ &= m \frac{1 + p - (\frac{1}{1+p})^{n-1}}{p}. \end{aligned}$$

The first line is a restatement of the summation we obtained earlier for the value of an annuity. The second line follows by applying the theorem for the summation of a geometric series. In the third line, we undo the earlier substitution $x = 1/(1+p)$. In the final step, both the numerator and denominator are multiplied by $1 + p$ to simplify the expression.

The resulting formula is much easier to use than a summation with dozens of terms. For example, what is the real value of a winning lottery ticket that pays $\$50,000$ per year for 20 years? Plugging in $m = \$50,000$, $n = 20$, and $p = 0.08$ gives $V \approx \$530,180$. Because payments are deferred, the million dollar lottery is really only worth about a half million dollars! This is a good trick for the lottery advertisers!

2.4 Infinite Geometric Series

The question at the beginning of this section was whether you would prefer a million dollars today or $\$50,000$ a year for the rest of your life. Of course, this depends on how long you live, so optimistically assume that the second option is to receive $\$50,000$ a year *forever*. This sounds like infinite money!

We can compute the value of an annuity with an infinite number of payments by taking the limit of our geometric sum in Theorem 2.1 as n tends to infinity. This one is worth remembering!

Theorem 2.2. *If* $|x| < 1$, *then*

$$\sum_{i=0}^{\infty} x^i = \frac{1}{1-x}.$$

Proof.

$$\begin{aligned} \sum_{i=0}^{\infty} x^i &= \lim_{n \to \infty} \sum_{i=0}^{n-1} x^i \\ &= \lim_{n \to \infty} \frac{1 - x^n}{1 - x} \\ &= \frac{1}{1 - x}. \end{aligned}$$

The first equality follows from the definition of an infinite summation. In the second line, we apply the formula for the sum of an n-term geometric series given in Theorem 2.1. The final line follows by evaluating the limit; the x^n term vanishes since we assumed that $|x| < 1$. □

In our annuity problem, $x = 1/(1 + p) < 1$, so the theorem applies. Substituting for x, we get an annuity value of

$$\begin{aligned} V &= m \cdot \frac{1}{1-x} \\ &= m \cdot \frac{1}{1 - 1/(1+p)} \\ &= m \cdot \frac{1+p}{(1+p)-1} \\ &= m \cdot \frac{1+p}{p}. \end{aligned}$$

Plugging in $m = \$50,000$ and $p = 0.08$ gives only $\$675,000$. Amazingly, a million dollars today is worth much more than $\$50,000$ paid every year forever! Then again, if we had a million dollars today in the bank earning 8% interest, we could take out and spend $\$80,000$ a year forever. So the answer makes some sense.

2.5 Examples

We now have formulas enabling us to sum both finite and infinite geometric series. Some examples are given below. In each case, the solution follows immediately from either Theorem 2.1 (for finite series) or Theorem 2.2 (for infinite series).

$$1 + 1/2 + 1/4 + 1/8 + \cdots = \sum_{i=0}^{\infty} (1/2)^i = \frac{1}{1-(1/2)} = 2 \tag{1}$$

$$0.999999999\ldots = 0.9 \sum_{i=0}^{\infty} (1/10)^i = 0.9 \frac{1}{1 - 1/10} = 0.9 \frac{10}{9} = 1 \tag{2}$$

$$1 - 1/2 + 1/4 - 1/8 + \cdots = \sum_{i=0}^{\infty} (-1/2)^i = \frac{1}{1-(-1/2)} = 2/3 \tag{3}$$

$$1 + 2 + 4 + 8 + \cdots + 2^{n-1} = \sum_{i=0}^{n-1} 2^i = \frac{1-2^n}{1-2} = 2^n - 1 \tag{4}$$

$$1 + 3 + 9 + 27 + \cdots + 3^{n-1} = \sum_{i=0}^{n-1} 3^i = \frac{1-3^n}{1-3} = \frac{3^n - 1}{2} \tag{5}$$

If the terms in a geometric series grow smaller as in equation (1), then the series is said to be *geometrically decreasing*. If the terms in a geometric series grow progressively larger as in equations (4) and (5), then the series is said to be *geometrically increasing*.

Here is a good rule of thumb: *the sum of a geometric series is approximately equal to the term with greatest absolute value*. In equations (1) and (3), the largest term is equal to 1 and the sums are 2 and 2/3, both relatively close to 1. In equation (4), the sum is about twice the largest term. In the final equation (5), the largest term is 3^{n-1} and the sum is $(3^n - 1)/2$, which is only about a factor of 1.5 greater.

2.6 Related Sums

We now know all about sums of geometric series. But in practice one often encounters sums that cannot be transformed by simple variable substitutions to the form $\sum x^i$.

A non-obvious, but useful way to obtain new summation formulas from old is by differentiating or integrating with respect to x. As an example, consider the following series.

$$\sum_{i=1}^{n} ix^i = x + 2x^2 + 3x^3 + \cdots + nx^n$$

This is not a geometric series, since the ratio between successive terms is not constant. Our formula for the sum of a geometric series cannot be directly applied. But suppose that we differentiate that formula:

$$\begin{aligned}
\frac{d}{dx}\sum_{i=0}^{n} x^i &= \frac{d}{dx}\frac{1-x^{n+1}}{1-x} \\
\sum_{i=0}^{n} ix^{i-1} &= \frac{-(n+1)x^n(1-x)-(-1)(1-x^{n+1})}{(1-x)^2} \\
&= \frac{-(n+1)x^n+(n+1)x^{n+1}+1-x^{n+1}}{(1-x)^2} \\
&= \frac{1-(n+1)x^n+nx^{n+1}}{(1-x)^2}.
\end{aligned}$$

Often differentiating or integrating messes up the exponent of x in every term. In this case, we now have a formula for a series of the form $\sum ix^{i-1}$, but we want a formula for the series $\sum ix^i$. The solution is simple: multiply by x. This gives:

$$\sum_{i=0}^{n} ix^i = \frac{x-(n+1)x^{n+1}+nx^{n+2}}{(1-x)^2}$$

Since we could easily have made a mistake, it is a good idea to go back and validate a formula obtained this way with a proof by induction.

Notice that if $|x| < 1$, then this sum converges to a finite value even if there are infinitely many terms. Taking the limit as n tends infinity gives the following theorem:

Theorem 2.3. *If $|x| < 1$, then*

$$\sum_{i=0}^{\infty} ix^i = \frac{x}{(1-x)^2}.$$

As a consequence, suppose there is an annuity that pays im dollars at the *end* of each year i forever. For example, if $m = \$50,000$, then the payouts are $\$50,000$ and then $\$100,000$ and then $\$150,000$

and so on. It is hard to believe that the value of this annuity is finite! But we can use the preceding theorem to compute the value:

$$\begin{aligned} V &= \sum_{i=1}^{\infty} \frac{im}{(1+p)^i} \\ &= m \frac{\frac{1}{1+p}}{(1 - \frac{1}{1+p})^2} \\ &= m \frac{1+p}{p^2}. \end{aligned}$$

The second line follows by an application of Theorem 2.3. The third line is obtained by multiplying the numerator and denominator by $(1+p)^2$.

For example, if $m = \$50,000$, and $p = 0.08$ as usual, then the value of the annuity is $V = \$8,437,500$. Even though payments increase every year, the increase is only additive with time; by contrast, dollars paid out in the future decrease in value exponentially with time. The geometric decrease swamps out the additive increase. Payments in the distant future are almost worthless, so the value of the annuity is finite.

The important thing to remember is the trick of taking the derivative (or integral) of a summation formula. Of course, this technique requires one to compute nasty derivatives correctly, but this is at least theoretically possible!

3 Book Stacking

Suppose you have a pile of books and you want to stack them on a table in some off-center way so the top book sticks out past books below it. How far past the edge of the table do you think you could get the top book to go without having the stack fall over? Could the top book stick out completely beyond the edge of table?

Most people's first response to this question—sometimes also their second and third responses—is "No, the top book will never get completely past the edge of the table." But in fact, you can get the top book to stick out as far as you want: one booklength, two booklengths, any number of booklengths!

3.1 Formalizing the Problem

We'll approach this problem recursively. How far past the end of the table can we get one book to stick out? It won't tip as long as its center of mass is over the table, we so can get it to stick out half its length, as shown in Figure 1.

Now suppose we have a stack of books that will stick out past the table edge without tipping over—call that a *stable* stack. Let's define the *overhang* of a stable stack to be the largest horizontal distance from the center of mass of the stack to the furthest edge of a book. If we place the center of mass of the stable stack at the edge of the table as in Figure 2, that's how far we can get a book in the stack to stick out past the edge.

So we want a formula for the maximum possible overhang, B_n, achievable with a stack of n books.

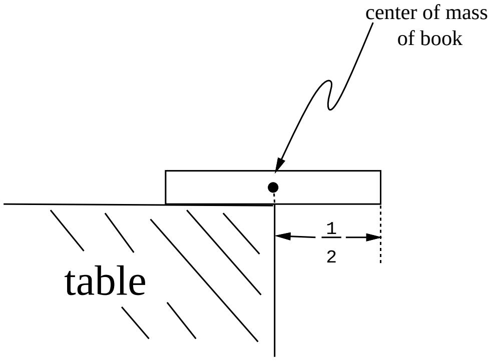


Figure 1: One book can overhang half a book length.

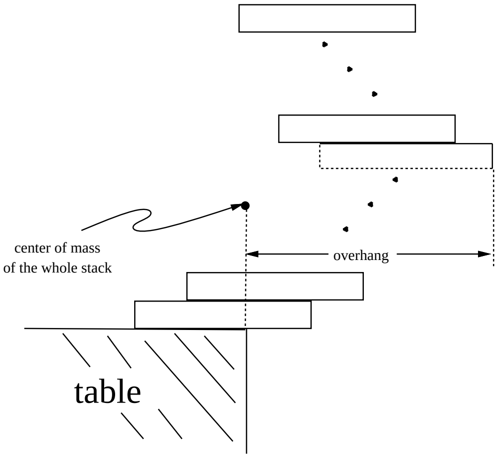


Figure 2: Overhanging the edge of the table.

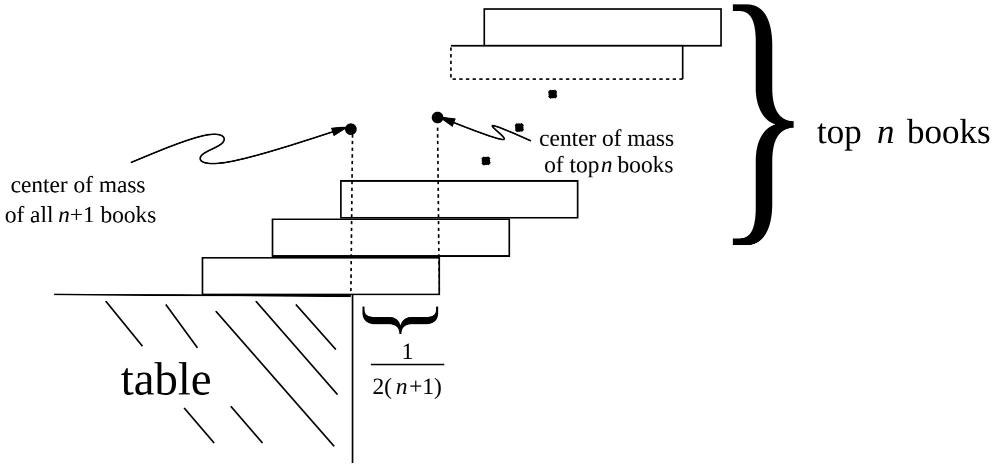


Figure 3: Additional overhang with $n + 1$ books.

We've already observed that the overhang of one book is 1/2 a book length. That is,

$$B_1 = \frac{1}{2}.$$

Now suppose we have a stable stack of $n + 1$ books with maximum overhang. If the overhang of the n books on top of the bottom book was not maximum, we could get a book to stick out further by replacing the top stack with a stack of n books with larger overhang. So the maximum overhang, B_{n+1}, of a stack of $n+1$ books is obtained by placing a maximum overhang stable stack of n books on top of the bottom book. And we get the biggest overhang for the stack of $n+1$ books by placing the center of mass of the n books right over the edge of the bottom book as in Figure 3.

So we know where to place the $n + 1$st book to get maximum overhang, and all we have to do is calculate what it is. The simplest way to do that is to let the center of mass of the top n books be the origin. That way the horizontal coordinate of the center of mass of the whole stack of $n + 1$ books will equal the increase in the overhang. But now the center of mass of the bottom book has horizontal coordinate $1/2$, so the horizontal coordinate of center of mass of the whole stack of $n+1$ books is

$$\frac{0 \cdot n + (1/2) \cdot 1}{n + 1} = \frac{1}{2(n + 1)}.$$

In other words,

$$B_{n+1} = B_n + \frac{1}{2(n + 1)}, \tag{6}$$

as shown in Figure 3.

Expanding equation (6), we have

$$\begin{aligned} B_{n+1} &= B_{n-1} + \frac{1}{2n} + \frac{1}{2(n+1)} \\ &= B_1 + \frac{1}{2 \cdot 2} + \cdots + \frac{1}{2n} + \frac{1}{2(n+1)} \\ &= \frac{1}{2} \sum_{i=1}^{n+1} \frac{1}{i}. \end{aligned}$$

Define

$$H_n ::= \sum_{i=1}^{n} \frac{1}{i}.$$

H_n is called the nth *Harmonic number*, and we have just shown that

$$B_n = \frac{H_n}{2}.$$

The first few Harmonic numbers are easy to compute. For example, $H_1 = 1$, $H_2 = 1 + \frac{1}{2} = \frac{3}{2}$, $H_3 = 1 + \frac{1}{2} + \frac{1}{3} = \frac{11}{6}$, $H_4 = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} = \frac{25}{12}$. The fact that H_4 is greater than 2 has special significance; it implies that the total extension of a 4-book stack is greater than one full book! This is the situation shown in Figure 4.

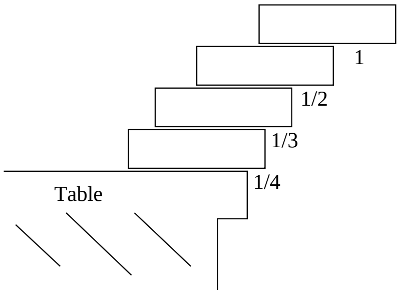


Figure 4: Stack of four books with maximum overhang.

In the next section we will prove that H_n grows slowly, but *unboundedly* with n. That means we can get books to overhang *any distance* past the edge of the table by piling them high enough!

3.2 Evaluating the Sum—The Integral Method

It would be nice to answer questions like, "How many books are needed to build a stack extending 100 book lengths beyond the table?" One approach to this question would be to keep computing Harmonic numbers until we found one exceeding 200. However, as we will see, this is not such a keen idea.

Such questions would be settled if we could express H_n in a closed form. Unfortunately, no closed form is known, and probably none exists. As a second best, however, we can find closed forms for very good approximations to H_n using the Integral Method. The idea of the Integral Method

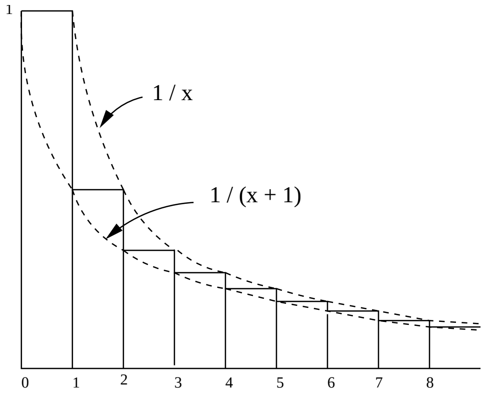


Figure 5: *This figure illustrates the Integral Method for bounding a sum. The area under the "stairstep" curve over the interval* $[0, n]$ *is equal to* $H_n = \sum_{i=1}^{n} 1/i$. *The function* $1/x$ *is everywhere greater than or equal to the stairstep and so the integral of* $1/x$ *over this interval is an upper bound on the sum. Similarly,* $1/(x+1)$ *is everywhere less than or equal to the stairstep and so the integral of* $1/(x+1)$ *is a lower bound on the sum.*

is to bound terms of the sum above and below by simple functions as suggested in Figure 5. The integrals of these functions then bound the value of the sum above and below.

The Integral Method gives the following upper and lower bounds on the harmonic number H_n:

$$\begin{aligned} H_n &\leq 1 + \int_1^n \frac{1}{x}\,dx = 1 + \ln n \\ H_n &\geq \int_0^n \frac{1}{x+1}\,dx = \int_1^{n+1} \frac{1}{x}\,dx = \ln(n+1). \end{aligned}$$

These bounds imply that the harmonic number H_n is around $\ln n$. Since $\ln n$ grows without bound, albeit slowly, we can make a stack of books that extends arbitrarily far.

For example, to build a stack extending three book lengths beyond the table, we need a number of books n so that $H_n \geq 6$. Exponentiating the above inequalities gives

$$e^{H_n - 1} \leq n \leq e^{H_n} - 1. \tag{7}$$

This implies that we will need somewhere between 149 and 403 books. Actual calculation of H_n shows that 227 books will be the minimum number to overhang three book lengths.

3.3 More about Harmonic Numbers

In the preceding section, we showed that H_n is about $\ln n$. A even better approximation is known:

$$H_n = \ln n + \gamma + \frac{1}{2n} + \frac{1}{12n^2} + \frac{\epsilon(n)}{120n^4}$$

Here γ is a value $0.577215664\ldots$ called Euler's constant, and $\epsilon(n)$ is between 0 and 1 for all n. We will not prove this formula.

The shorthand $H_n \sim \ln n$ is used to indicate that the leading term of H_n is $\ln n$. More precisely:

Definition 3.1. For functions $f, g : \mathbb{R} \to \mathbb{R}$, we say f is *asymptotically equal* to g, in symbols,

$$f(x) \sim g(x)$$

iff

$$\lim_{x\to\infty} f(x)/g(x) = 1.$$

We also might write $H_n \sim \ln n + \gamma$ to indicate two leading terms. While this notation is widely used, it is not really right. Referring to the definition of $\sim$, we see that while $H_n \sim \ln n + \gamma$ is a true statement, so is $H_n \sim \ln n + c$ where c is any constant. The correct way to indicate that γ is the second-largest term is $H_n - \ln n \sim \gamma$.

The reason that the $\sim$ notation is useful is that often we do not care about lower order terms. For example, if $n = 100$, then we can compute $H(n)$ to great precision using only the two leading terms:

$$|H_n - \ln n - \gamma| \leq \left| \frac{1}{200} - \frac{1}{120000} + \frac{1}{120 \cdot 100^4} \right| < \frac{1}{200}.$$

4 Finding Summation Formulas

The source of the simple formula $\sum_{i=1}^{n} i = n(n+1)/2$ is still a mystery! Sure, we can prove this statement true by induction, but where did the expression on the right come from? Even more inexplicable is the summation formula for consecutive squares:

$$\begin{aligned}
\sum_{i=1}^{n} i^2 &= \frac{(2n+1)(n+1)n}{6} \\
&= \frac{n^3}{3} + \frac{n^2}{2} + \frac{n}{6} \\
&\sim \frac{n^3}{3}.
\end{aligned}$$

Here is how we might find the sum-of-squares formula if we forgot it or had never seen it. First, the Integral Method gives a quick estimate of the sum:

$$\begin{gathered}
\int_0^n x^2\,dx \leq \sum_{i=1}^{n} i^2 \leq \int_0^n (x+1)^2\,dx \\
\frac{n^3}{3} \leq \sum_{i=1}^{n} i^2 \leq \frac{(n+1)^3}{3} - \frac{1}{3}.
\end{gathered}$$

These upper and lower bounds obtained by the Integral Method show that $\sum_{i=1}^{n} i^2 \sim n^3/3$. To get an exact formula, we then guess the general form of the solution. Where we are uncertain, we can add parameters $a, b, c, \ldots$. For example, we might make the guess:

$$\sum_{i=1}^{n} i^2 = an^3 + bn^2 + cn + d.$$

If the guess is correct, then we can determine the parameters a, b, c, and d by plugging in a few values for n. Each such value gives a linear equation in a, b, c, and d. If we plug in enough values, we may get a linear system with a unique solution. Applying this method to our example gives:

$$\begin{aligned} n = 0 &\quad\rightarrow\quad 0 = d \\ n = 1 &\quad\rightarrow\quad 1 = a + b + c + d \\ n = 2 &\quad\rightarrow\quad 5 = 8a + 4b + 2c + d \\ n = 3 &\quad\rightarrow\quad 14 = 27a + 9b + 3c + d. \end{aligned}$$

Solving this system gives the solution $a = 1/3, b = 1/2, c = 1/6, d = 0$. Therefore, if our initial guess at the form of the solution was correct, then the summation is equal to $n^3/3 + n^2/2 + n/6$. In fact, our initial guess *was* correct, this is the right formula for the sum of squares!

Be careful! After obtaining a formula by this method, always go back and prove it using induction or some other method. This is not merely a check for algebra blunders; if the initial guess at the solution was not of the right form, then the resulting formula will be completely wrong!

5 Double Sums

Sometimes we have to evaluate sums of sums, otherwise known as *double summations*. Sometimes it is easy: we can evaluate the inner sum, replace it with a closed form, and then evaluate the outer sum which no longer has a summation inside it.

But there's a special trick that is often extremely useful for sums, which is *exchanging the order of summation.* It's best demonstrated by example. Suppose we want to compute the sum of the harmonic numbers

$$\sum_{k=1}^{n} H_k = \sum_{k=1}^{n} \sum_{j=1}^{k} 1/j$$

For intuition about this sum, we can try the integral method:

$$\sum_{k=1}^{n} H_k \approx \int_{k=1}^{n} \ln n \approx n \ln n - n.$$

Now let's look for an exact answer. If we think about the pairs (k, j) over which we are summing, they form a triangle:

		j						
		1	2	3	4	5	...	n
k	1	1						
	2	1	1/2					
	3	1	1/2	1/3				
	4	1	1/2	1/3	1/4			
		...						
	n	1	1/2		...			$1/n$

The summation above is summing each row and then adding the row sums. Instead, we can sum the columns and then add the column sums. Inspecting the table we see that this double sum can be written as

$$\begin{aligned}\sum_{k=1}^{n} H_k &= \sum_{k=1}^{n}\sum_{j=1}^{k} 1/j \\ &= \sum_{j=1}^{n}\sum_{k=j}^{n} 1/j \\ &= \sum_{j=1}^{n} 1/j \sum_{k=j}^{n} 1 \\ &= \sum_{j=1}^{n} \frac{1}{j}(n-j+1) \\ &= \sum_{j=1}^{n} \frac{n-j+1}{j} \\ &= \sum_{j=1}^{n} \frac{n+1}{j} - \sum_{j=1}^{n} \frac{j}{j} \\ &= (n+1)\sum_{j=1}^{n} \frac{1}{j} - \sum_{j=1}^{n} 1 \\ &= (n+1)H_n - n.\end{aligned}$$

6 Stirling's Approximation

The familiar factorial notation, $n!$, is an abbreviation for the product

$$\prod_{i=1}^{n} i.$$

This is by far the most common product in Discrete Mathematics. In this section we describe a good closed-form estimate of $n!$ called *Stirling's Approximation*. Unfortunately, all we can do is estimate: there is no closed form for $n!$ — though proving so would take us beyond the scope of 6.042.

A good way to handle a product is often to convert it into a sum by taking the logarithm. In the case of factorial, this gives

$$\begin{aligned}\ln(n!) &= \ln(1\cdot 2\cdot 3\cdots(n-1)\cdot n) \\ &= \ln 1 + \ln 2 + \ln 3 + \cdots + \ln(n-1) + \ln n \\ &= \sum_{i=1}^{n} \ln i.\end{aligned}$$

We've not seen a summation containing a logarithm before! Fortunately, one tool that we used in evaluating sums is still applicable: the Integral Method. We can bound the terms of this sum with

$\ln x$ and $\ln(x+1)$ as shown in Figure 6. This gives bounds on $\ln(n!)$ as follows:

$$\begin{aligned}
\int_1^n \ln x\,dx \leq \quad & \textstyle\sum_{i=1}^n \ln i & \leq \int_0^n \ln(x+1)\,dx \\
n\ln(\frac{n}{e})+1 \leq \quad & \textstyle\sum_{i=1}^n \ln i & \leq (n+1)\ln\left(\frac{n+1}{e}\right)+1 \\
\left(\frac{n}{e}\right)^n e \leq \quad & n! & \leq \left(\frac{n+1}{e}\right)^{n+1} e.
\end{aligned}$$

The second line follows from the first by completing the integrations. The third line is obtained by exponentiating.

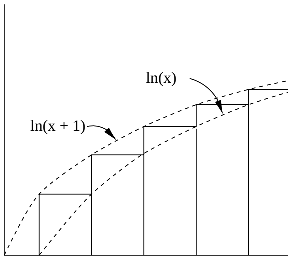


Figure 6: *This figure illustrates the Integral Method for bounding the sum $\sum_{i=1}^n \ln i$.*

So $n!$ behaves something like the closed form formula $(\frac{n}{e})^n$. A more careful analysis yields an unexpected closed form formula that is asymptotically exact:

Lemma (Stirling's Formula).

$$n! \sim \left(\frac{n}{e}\right)^n \sqrt{2\pi n},$$

Stirling's Formula describes how $n!$ behaves in the limit, but to use it effectively, we need to know how close it is to the limit for different values of n. That information is given by the bounding formulas:

Fact (Stirling's Approximation).

$$\sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{1/(12n+1)} \leq n! \leq \sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{1/12n}.$$

The Approximation implies the asymptotic Formula, since since $e^{1/(12n+1)}$ and $e^{1/12n}$ both approach 1 as n grows large. These inequalities can be verified by induction, but the details are nasty.

The bounds in Stirling's formula are very tight. For example, if $n = 100$, then Stirling's bounds are:

$$\begin{aligned}
100! &\geq \sqrt{200\pi}\left(\frac{100}{e}\right)^{100} e^{1/1201} \\
100! &\leq \sqrt{200\pi}\left(\frac{100}{e}\right)^{100} e^{1/1200}
\end{aligned}$$

The only difference between the upper bound and the lower bound is in the final term. In particular $e^{1/1201} \approx 1.00083299$ and $e^{1/1200} \approx 1.00083368$. As result, the upper bound is no more than $1 + 10^{-6}$ times the lower bound. This is amazingly tight! Remember Stirling's formula; we will use it often.

6.1 Double Summing

Another way to derive Stirling's approximation is to remember that $\ln n$ is roughly the same as H_n. This lets us use the result we derived before for $\sum H_k$ via double summation. Our approximation for H_k told us that $\ln(k+1) \leq H_k \leq 1 + \ln k$. Rewriting, we find that $H_k - 1 \leq \ln k \leq H_{k-1}$. It follows that (leaving out the $i = 1$ term in the sum, which contributes 0),

$$\begin{aligned}
\sum_{i=2}^{n} \ln i &\leq \sum_{i=2}^{n} H_{i-1} \\
&= \sum_{i=1}^{n-1} H_i \\
&= nH_{n-1} - n \\
&\leq n(1 + \ln(n-1)) - n \\
&= n \ln(n-1),
\end{aligned}$$

roughly the same bound as we proved before via the integral method. We can derive a similar lower bound.

7 Asymptotic Notation

Asymptotic notation is a shorthand used to give a quick measure of the behavior of a function $f(n)$ as n grows large.

7.1 Little Oh

The asymptotic notation $\sim$ is an equivalence relation indicating that $\sim$-equivalent functions grow at exactly the same rate. There is a corresponding strict partial order on functions indicating that one function grows at a significantly slower rate. Namely,

Definition 7.1. For functions $f, g : \mathbb{R} \to \mathbb{R}$, we say f is *asymptotically smaller* than g, in symbols,

$$f(x) = o(g(x)),$$

iff

$$\lim_{x\to\infty} f(x)/g(x) = 0.$$

For example, $1000x^{1.9} = o(x^2)$, because $1000x^{1.9}/x^2 = 1000/x^{0.1}$ and since $x^{0.1}$ goes to infinity with x and 1000 is constant, we have $\lim_{x\to\infty} 1000x^{1.9}/x^2 = 0$. This argument generalizes directly to yield

Lemma 7.2. $x^a = o(x^b)$ *for all nonnegative constants* $a < b$.

Using the familiar fact that $\log x < x$ for all $x > 1$, we can prove

Lemma 7.3. $\log x = o(x^\epsilon)$ *for all* $\epsilon > 0$ *and* $x > 1$.

Proof. Choose $\epsilon > \delta > 0$ and let $x = z^\delta$ in the inequality $\log x < x$. This implies

$$\log z < z^\delta/\delta = o(z^\epsilon) \qquad \text{by Lemma 7.2.} \tag{8}$$

□

Corollary 7.4. $x^b = o(a^x)$ *for any* $a, b \in \mathbb{R}$ *with* $a > 1$.

Proof. From (8),

$$\log z < z^\delta/\delta \tag{9}$$

for all $z > 1$, $\delta > 0$. Hence

$$\begin{aligned} (e^b)^{\log z} &< (e^b)^{z^\delta/\delta} && (10)\\ z^b &< (e^{\log ab/\log a})^{z^\delta/\delta} && (11)\\ &= a^{(b/\delta\log a)z^\delta} && (12)\\ &< a^z && (13) \end{aligned}$$

for all z such that $(b/\delta \log a)z^\delta < z$. But since $z^\delta = o(z)$, this last inequality holds for all large enough z. □

Lemma 7.3 and Corollary 7.4 can also be proved easily in several other ways, *e.g.*, using L'Hopital's Rule or the McLaurin Series for $\log x$ and e^x. Proofs can be found in most calculus texts.

Problem 1. Prove the initial claim that $\log x < x$ for all $x > 1$ (requires elementary calculus).

Problem 2. Prove that the relation, R, on functions such that fRg iff $f = o(g)$ is a strict partial order, namely, R is transitive and *asymmetric*: if fRg then $\neg gRf$.

Problem 3. Prove that $f \sim g$ iff $f = g + h$ for some function $h = o(f)$.

7.2 Big Oh

Big Oh is the most frequently used asymptotic notation. It is used to give an upper bound on the growth of a function, such as the running time of an algorithm.

Definition 7.5. Given functions $f, g : \mathbb{R} \mapsto \mathbb{R}$, with g nonnegative, we say that

$$f = O(g)$$

iff

$$\limsup_{x\to\infty} |f(x)| / g(x) < \infty.$$

This definition[2] makes it clear that

Lemma 7.6. *If $f = o(g)$ or $f \sim g$, then $f = O(g)$.*

Proof. $\lim f/g = 0$ or $\lim f/g = 1$ implies $\lim f/g < \infty$. □

It is easy to see that the converse of Lemma 7.6 is not true. For example, $2x = O(x)$, but $2x \not\sim x$ and $2x \neq o(x)$.

We also observe,

Lemma 7.7. *If $f = o(g)$, then it is* not *true that $g = O(f)$.*

Proof. $\limsup g/f = 1/\limsup f/g = 1/0 = \infty$, so $g \neq O(f)$. □

The usual formulation of Big Oh spells out the definition of $\limsup$ without mentioning it. Namely, here is an equivalent definition:

Definition 7.8. Given functions $f, g : \mathbb{R} \mapsto \mathbb{R}$, we say that

$$f = O(g)$$

iff there exists a constant $c \geq 0$ and an x_0 such that for all $x \geq x_0$, $|f(x)| \leq cg(x)$.

This definition is rather complicated, but the idea is simple: $f(x) = O(g(x))$ means $f(x)$ is less than or equal to $g(x)$, except that we're willing to ignore a constant factor (*i.e.*, c) and to allow exceptions for small x (*i.e.*, $x < x_0$).

Proposition 7.9. $100x^2 = O(x^2)$.

[2]

$$\limsup_{x\to\infty} h(x) ::= \lim_{x\to\infty} \text{lub}_{y\geq x} h(y).$$

We need the $\limsup$ in the definition of $O()$ because if $f(x)/g(x)$ oscillates between, say, 3 and 5 as x grows, then $f = O(g)$ because $f \leq 5g$, but $\lim_{x\to\infty} f(x)/g(x)$ does not exist. However, in this case we would have $\limsup_{x\to\infty} f(x)/g(x) = 5$.

Proof. Choose $c = 100$ and $x_0 = 1$. Then the proposition holds, since for all $x \geq 1$, $\left|100x^2\right| \leq 100x^2$. □

Proposition 7.10. $x^2 + 100x + 10 = O(x^2)$.

Proof. $(x^2+100x+10)/x^2 = 1+100/x+10/x^2$ and so its limit as x approaches infinity is $1+0+0 = 1$. So in fact, $x^2 + 100x + 10 \sim x^2$, and therefore $x^2 + 100x + 10 = O(x^2)$. Indeed, it's conversely true that $x^2 = O(x^2 + 100x + 10)$. □

Proposition 7.10 generalizes to an arbitrary polynomial by a similar proof, which we omit.

Proposition 7.11. *For $a_k \neq 0$, $a_k x^k + a_{k-1}x^{k-1} + \cdots + a_1 x + a_0 = O(x^k)$.*

Big Oh notation is especially useful when describing the running time of an algorithm. For example, the usual algorithm for multiplying $n \times n$ matrices requires proportional to n^3 operations in the worst case. This fact can be expressed concisely by saying that the running time is $O(n^3)$. So this asymptotic notation allows the speed of the algorithm to be discussed without reference to constant factors or lower-order terms that might be machine specific. In this case there is another, ingenious matrix multiplication procedure that requires $O(n^{2.55})$ operations. This procedure will therefore be much more efficient on large enough matrices. Unfortunately, the $O(n^{2.55})$-operation multiplication procedure is almost never used because it happens to be less efficient than the usual $O(n^3)$ procedure on matrices of practical size. It is even conceivable that there is an $O(n^2)$ matrix multiplication procedure, but none is known.

7.3 Theta

Definition 7.12.

$$f = \Theta(g) \quad \text{iff} \quad f = O(g) \wedge g = O(f).$$

The statement $f = \Theta(g)$ can be paraphrased intuitively as "f and g are equal to within a constant factor."

The value of these notations is that they highlight growth rates and allow suppression of distracting factors and low-order terms. For example, if the running time of an algorithm is

$$T(n) = 10n^3 - 20n^2 + 1,$$

then

$$T(n) = \Theta(n^3).$$

In this case, we would say that T *is of order* x^3 or that $T(n)$ *grows cubically*.

Another such example is

$$\pi^2 3^{x-7} + \frac{(2.7x^{113} + x^9 - 86)^4}{\sqrt{x}} - 1.08^{3x} = \Theta(3^x).$$

Just knowing that the running time of an algorithm is $\Theta(n^3)$, for example, is useful, because if n doubles we can predict that the running time will *by and large* [3] increase by a factor of at most 8 for large n. In this way, Theta notation preserves information about the scalability of an algorithm or system. Scalability is, of course, a big issue in the design of algorithms and systems.

Figure 7 illustrates the relationships among the asymptotic growth notations we have considered.

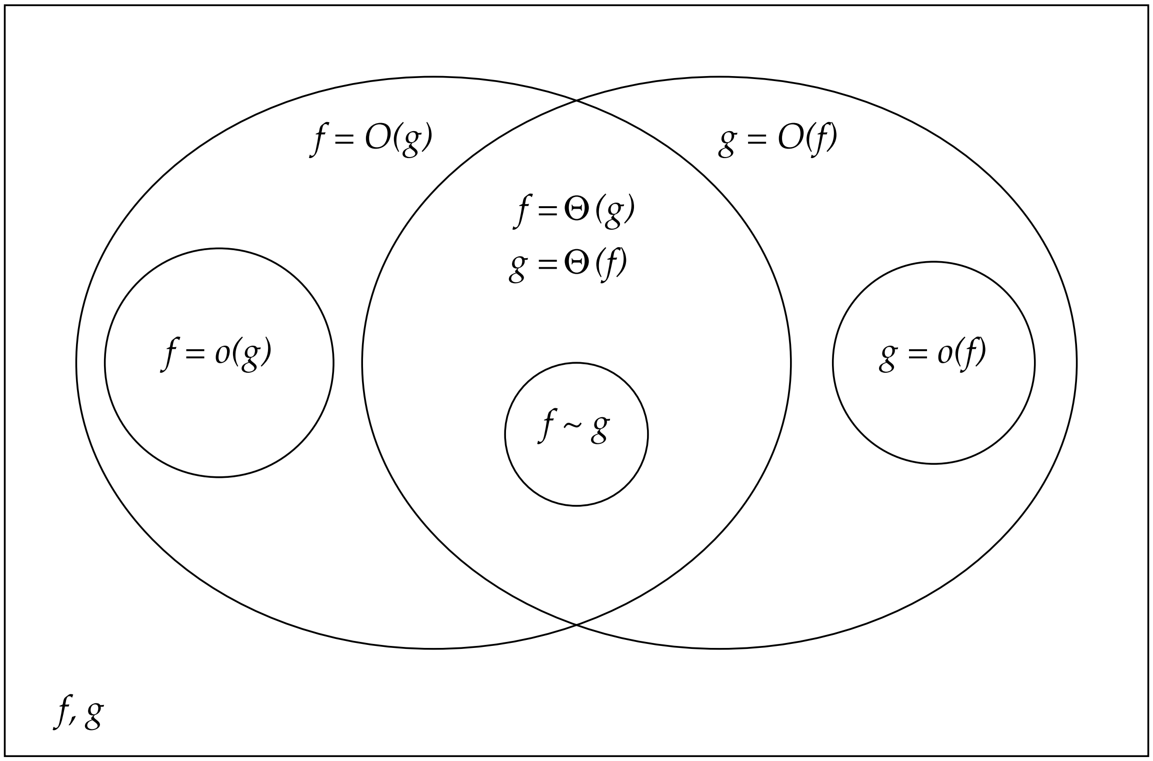


Figure 7: Venn Diagram describing Asymptotic Relations

7.4 Pitfalls with Big Oh

There is a long list of ways to make mistakes with Big Oh notation. This section presents some of the ways that Big Oh notation can lead to ruin and despair.

7.4.1 The Exponential Fiasco

Sometimes relationships involving Big Oh are not so obvious. For example, one might guess that $4^x = O(2^x)$ since 4 is only a constant factor larger than 2. This reasoning is incorrect, however; actually 4^x grows much faster than 2^x.

Proposition 7.13. $4^x \neq O(2^x)$

[3] Since $\Theta(n^3)$ only implies that the running time, $T(n)$, is between cn^3 and dn^3 for constants $0 < c < d$, the time $T(2n)$ could regularly exceed $T(n)$ by a factor as large as $8d/c$. The factor is sure to be close to 8 for all large n only if $T(n) \sim n^3$.

Proof. $2^x/4^x = 2^x/(2^x 2^x) = 1/2^x$. Hence, $\lim_{x\to\infty} 2^x/4^x = 0$, so in fact $2^x = o(4^x)$. We observed earlier that this implies that $4^x \neq O(2^x)$. □

7.4.2 Constant Confusion

Every constant is $O(1)$. For example, $17 = O(1)$. This is true because if we let $f(x) = 17$ and $g(x) = 1$, then there exists a $c > 0$ and an x_0 such that $|f(x)| \leq cg(x)$. In particular, we could choose c = 17 and $x_0 = 1$, since $|17| \leq 17 \cdot 1$ for all $x \geq 1$. We can construct a false theorem that exploits this fact.

False Theorem 7.14.

$$\sum_{i=1}^{n} i = O(n)$$

Proof. Define $f(n) = \sum_{i=1}^{n} i = 1 + 2 + 3 + \cdots + n$. Since we have shown that every constant i is $O(1)$, $f(n) = O(1) + O(1) + \cdots + O(1) = O(n)$. □

Of course in reality $\sum_{i=1}^{n} i = n(n+1)/2 \neq O(n)$.

The error stems from confusion over what is meant in the statement $i = O(1)$. For any *constant* $i \in \mathbb{N}$ it is true that $i = O(1)$. More precisely, if f is any constant function, then $f = O(1)$. But in this False Theorem, i is not constant but ranges over a set of values 0,1, . . . ,n that depends on n.

And anyway, we should not be adding $O(1)$'s as though they were numbers. We never even defined what $O(g)$ means by itself; it should only be used in the context "$f = O(g)$" to describe a relation between functions f and g.

7.4.3 Lower Bound Blunder

Sometimes people incorrectly use Big Oh in the context of a lower bound. For example, they might say, "The running time, $T(n)$, is at least $O(n^2)$," when they probably mean something like "$O(T(n)) = n^2$," or more properly, "$n^2 = O(T(n))$."

7.4.4 Equality Blunder

The notation $f = O(g)$ is too firmly entrenched to avoid, but the use of "=" is really regrettable. For example, if $f = O(g)$, it seems quite reasonable to write $O(g) = f$. But doing so might tempt us to the following blunder: because $2n = O(n)$, we can say $O(n) = 2n$. But $n = O(n)$, so we conclude that $n = O(n) = 2n$, and therefore $n = 2n$. To avoid such nonsense, we will never write "$O(f) = g$."

Basic Counting, Pigeonholing, Permutations

1 Counting by Matching

Counting is a theme throughout discrete mathematics: how many leaves in a tree, minimal colorings of a graph, trees with a given set of vertices, five-card hands in a deck of fifty-two, consistent rankings of players in a tournament, stable marriages given boy's and girl's preferences, and so on.

A good way to count things is to match up things to be counted with other things that we know how to count. We saw an example of this early in the term when we counted the size of a powerset of a set of size n by finding an exact matching between elements of the powerset and the 2^n binary strings of length n.

The matching doesn't have to be exact, *i.e.*, a bijection, to be informative. For example, suppose we want to determine the cardinality of the set of watches in the 6.042 classroom on a typical day. The set of watches can be correlated with the set of people in the room; specifically, for each person there is at most one watch (at least, let's assume this). Now we know something about the cardinality of the set of students, since there are only 146 people signed up for 6.042. There are also three lecturers and eight TA's, and these would typically be the only nonstudents in the room. So we can conclude that there are *at most* 157 watches in the classroom on a typical day.

This type of argument is very simple, but also quite powerful. We will see how to use such simple arguments to prove results that are hard to obtain any other way.

2 Matchings as Bijections

The "matching up" we talked about more precisely refers to finding injections, surjections, and bijections between things we want to count and things to be counted. The following Theorem formally justifies this kind of counting:

Theorem 2.1. *Let A and B be finite sets and f : from A to B be a function. If*

1. *f is a bijection, then $|A| = |B|$,*
2. *f is an injection, then $|A| \leq |B|$,*
3. *f is a surjection, then $|A| \geq |B|$.*

This is one of those theorems that is so fundamental that it's not clear what simpler axioms are appropriate to use in proving it. In fact, we can't prove it yet, because we haven't defined the concept that it's all about, namely, the size or *cardinality*, $|A|$, of a finite set, A. Intuitively, a set, A, has n elements if it equals $\{a_1, a_2, \ldots, a_n\}$ where the a_i are all different. Now ellipsis is dangerous, so we should avoid it in a definition this basic. What is the notation "$a_1, a_2, \ldots, a_n$" intended to convey? It means that there is a first element, a_1, and a second element, a_2, and in general, given any $i \leq n$, there is an ith element a_i. Also, all the a_i's for different i's are different. This explains how we arrive at a rigorous definition:

Definition 2.2. A set A has *cardinality* $n \in \mathbb{N}$, in symbols, $|A| = n$, iff there is a **bijection** from $\{1, 2, \ldots, n\}$ to A. The special case when $n = 0$ is that $|\emptyset| = 0$. A set is *finite* iff it has cardinality n for some $n \in \mathbb{N}$.

With this definition, we could prove Theorem 2.1 by appeal to basic properties of functions and natural numbers. For example, if $f : A$ to B is a bijection, and $|A| = n$, then we can prove that $|B| = n$ as follows: since $|A| = n$, there is a bijection $g : \{1, 2, \ldots, n\}$ to A. Then $f \circ g$ is a bijection from $\{1, 2, \ldots, n\}$ to B, so by definition, $|B| = n$.

Here we used the fact that the composition of bijections is a bijection. This fact itself follows just from the logical properties of equality and the definition of a bijection; it does not even depend on any properties of numbers. So we can say that we proved part 1 of Theorem 2.1 from more fundamental mathematical concepts. The other two parts can be proved using similar properties of functions along with ordinary induction, but we'll skip them: the proofs are exercises in formal logic that are not very informative about counting.

Notice that the condition in Theorem 2.1 that A and B are *finite* sets is important. It's not even clear what the size of an infinite set ought to be, or whether it's possible for one infinite set to be "larger" than another. We'll avoid this issue in these notes by only counting the sizes of finite sets.

2.1 Counting Functions

The bijection between length n binary strings and a powerset can be generalized to help in counting the number of functions from one set to another:

Question: How many different functions are there from finite set A to finite set B?

Theorem 2.3. *If A and B are finite sets, with $|A| = n$ and $|B| = m$, then the cardinality of the set of functions from A to B is m^n.*

Proof. We will use a bijection from $\{f \mid f : A \rightarrow B\}$ to $\{s \mid s$ is a length n string of elements from $B\}$. The mapping is $f \mapsto s_f$ where $s_f ::= f(a_1)f(a_2)\cdots f(a_n)$, *i.e.*, the value of the ith position in s_f is equal to $f(a_i)$.

We will prove that this mapping is a bijection. First we prove that the mapping is injective (one-to-one) by contradiction. Suppose that $f \neq g$ and $s_f = s_g$. But $f \neq g$ implies that there exists an $i \in \mathbb{N}$, where $1 \leq i \leq n$, we have $f(a_i) \neq g(a_i)$. But that implies that the ith position in s_f and s_g are different, which is a contradiction.

Next we prove that the mapping is surjective (onto), *i.e.*, that every length n string, s, of elements from B equals s_f for some function $f : A \rightarrow B$. Denote the ith position in such a string, s, by $s[i]$.

Now define a function f by the rule that $f(a_i) ::= s[i]$, for $1 \leq i \leq n$. This defines the required $f : A \to B$ such that $s_f = s$.

Since this mapping is a bijection, we know that the number of functions from A to B is equal to the number of strings of length n from the elements of B. In section 5 we will see that the number of such strings is m^n by the Product Rule. □

3 The Pigeonhole Principle

Theorem 2.1 part 2 tells us that if there is an injection from A to B, then $|A| \leq |B|$. The contrapositive of this statement is that if $|A| > |B|$, and f is a function from A to B, then f is not an injection.

Corollary 3.1. *Let A and B be finite sets. If $|A| > |B|$ and $f : A \to B$, then there exist distinct elements a and a' in A such that $f(a) = f(a')$.*

This Corollary is known as the *Pigeonhole Principle* because it can be paraphrased as:

The Pigeonhole Principle: If there are more pigeons than pigeonholes, then there must be at least two pigeons in one hole.

Proof. Let A be the set of "pigeons", let B be the set of "holes", and let the function $f : A \to B$ define the assignment of pigeons to holes. Since $|A| > |B|$, Corollary 3.1 implies that there exist two distinct pigeons, $a \neq a'$, assigned to the same hole, $f(a)$. □

As a trivial application of the Pigeonhole Principle, suppose that there are three people in a room. The pigeonhole principle implies that two have the same gender. In this case, the "pigeons" are the three people and the "pigeonholes" are the two possible genders, male and female. Since there are more pigeons than holes, two pigeons must be in the same pigeonhole; that is, two people must have the same gender.

Claim 3.2. *In New York (City) there live at least two people with the same number of hairs.*

Proof (found on the web). I ran experiments with members of my family. My teenage son secured himself the highest marks sporting, in my estimate, about 900 hairs per square inch. Even assuming a pathological case of a 6 feet (two-sided) fellow 50 inch across, covered with hair head, neck, shoulders and so on down to the toes, the fellow would have somewhere in the vicinity of 7,000,000 hairs which is probably a very gross over-estimate to start with. The Hammond's World Atlas I purchased some 15 years ago, estimates the population of the New York City between 7,500,000 and 9,000,000. The assertion therefore follows from the pigeonhole principle. □

The pigeonhole principle seems too obvious to be really useful, but the next two examples show how it gives short proofs of results that are difficult to obtain by other means.

3.1 Pigeonhole Principle Example: A Final Exam Question

A problem on an old final exam was to prove the following claim:

Claim. *In every set of 1000 integers, there are two integers x and y such that $573 \mid (x - y)$.*

At first glance, this looks very hard! Those 1000 numbers could be anything! Since there are no less than 1000 integer-valued variables here, even our old standby, induction, seems hopeless. Surprisingly, however, there is a short proof using the Pigeonhole Principle.

To apply the Pigeonhole Principle, we must identify two things: pigeons and holes. Furthermore, to prove anything with the Pigeonhole Principle, we must have more pigeons than holes. Since there are only two numbers mentioned in this problem, a natural thing to try is 1000 pigeons and 573 holes.

Under this interpretation, a pigeon is an integer, but what is a hole? Ideally, the existence of two pigeons and in the same hole should be equivalent to the existence of two numbers x and y such that $573 \mid (x - y)$. This suggests numbering the holes $0, 1, \ldots 572$ and putting in hole n all integers congruent to n modulo 573. Now we can construct a proof:

Proof. Let S be a set of 1000 integers. Let $M = \{0, 1, \ldots 572\}$. Let f from S to M be the function defined by $f(n) = n \bmod 573$. Since $|S| > |M|$, Corollary 3.1 implies that there exist distinct elements x and y in S such that $f(x) = f(y)$. This means $(x \bmod 573) = (y \bmod 573)$ and so $573 \mid (x - y)$. □

Really there was nothing special about the numbers 1000 and 573 other than the fact that $1000 > 573$. We could have made a stronger claim: if $n > m$, then in every set of n integers, there are two integers x and y such that $m \mid (x - y)$.

3.2 Example: Subsets of a List of Numbers

Show that any given 10 distinct positive numbers less than 100, that two completely different subsets sum to the same quantity.

The numbers all vary between 1 and 99. Therefore the maximum sum of any 10 chosen numbers is $90 + 91 + 92 + \ldots 99 = 945$. The number of different subsets of the 10 numbers is $2^{10} - 1$ (excluding the null set) $= 1023$. We have 1023 pigeons and 945 holes. Using the pigeonhole principle, we can argue that two different subsets map to the same sum. If these subsets have a common number or numbers, we can always remove the common numbers to produce two completely different subsets that sum to the same quantity.

3.3 20 Questions and Binary Search

Here is a game. I think of an animal. You can ask me 20 questions that take a yes/no answer such as, "Is the animal bigger than a breadbox?" To win the game, you must ask a question like, "Is the animal a walrus?" or "Is the animal a zebra?" and receive a "yes" answer. In effect, you have 19 questions to determine which animal I am thinking of, and then you must use 1 question to confirm your guess.

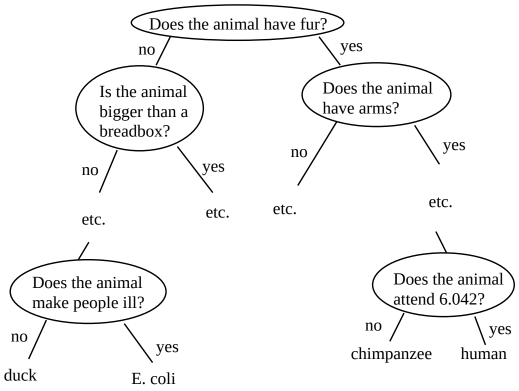

Does the animal have fur?
no
yes
Is the animal bigger than a breadbox?
Does the animal have arms?
no
yes
no
yes
etc.
etc.
etc.
etc.
Does the animal make people ill?
Does the animal attend 6.042?
no
yes
no
yes
duck
E. coli
chimpanzee
human

Figure 1: A strategy for the animal game can be represented by a depth-19 binary tree. Each internal node represents a yes/no question such as, "Does the animal have fur?" Each leaf node represents a final guess. A run of the algorithm corresponds to a path from the root to a leaf.

Suppose that I know a million animals. Can you always determine which animal I am thinking of? Any questioning strategy you use can be represented by a depth-19 binary tree as shown in Figure 1. Each internal node in the tree represents a question, and each leaf represents a final guess at my animal. A depth-19 binary tree can have at most $2^{19} = 524,288$ leaves, and I can use any of a million animals. By the Pigeonhole Principle, at least two animals must be associated with some leaf in the tree; this implies that you cannot always determine which animal I am thinking of with only 19 questions. More generally, if I know n animals, then $\lceil \log_2 n \rceil$ questions are necessary to always identify the one I'm thinking of; a binary tree of lower depth must have fewer than n leaves, and so some animals cannot be distinguished.

A similar argument applies to a binary search algorithm. In a binary search, we are looking for a particular item in a *sorted* list. We begin by comparing the middle element in the list to the item we are looking for. If our item precedes the middle element, then we continue the search recursively in the first half of the list. Similarly, if our item follows the middle element, then we search recursively in the second half of the list. For example, binary search could be applied to the animal game. You could sort the list of a million animals, pick out the middle one, and begin with a question like, "Does the animal alphabetically precede marmot?" Not surprisingly, given the similarity between the animal game and binary search, a Pigeonhole Principle argument shows that binary search requires at least $\log n$ comparisons to find an item in an n-element list in the worst case.

3.4 Example: Weighing Coins

Now let's consider the problem of identifying an off-weight counterfeit coin among a collection of coins using a balance scale. In this example, we'll do a refined analysis using the Pigeonhole Principle.

Let's consider 12 coins of which 11 have the same weight and a counterfeit one with a different weight. With three weighings on a balance scale, you must identify the counterfeit coin and determine whether it is heavier or lighter than the rest. (A balance scale has a left pan and a right pan. In a weighing you put some coins in each pan. The scale then reveals whether the left pan is heavier, the right pan is heavier, or the two are equal.)

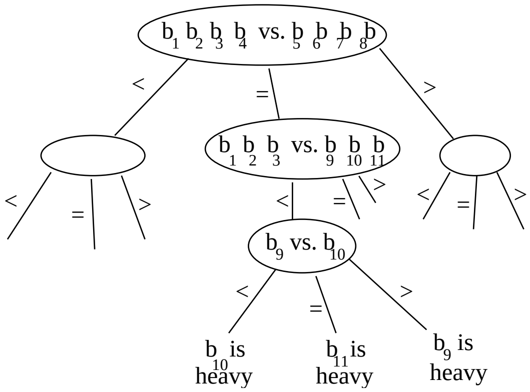


Figure 2: A strategy for the weighing problem can be represented by a ternary tree. Each internal node represents a weighing. Each leaf node represents a result. A run of the algorithm corresponds to a path from the root to a leaf. At each internal node, we perform the weighing associated with the node and descend to a child based on the result. At a leaf, we output the indicated result.

The problem is solvable using a tricky algorithm represented by a ternary tree as shown in Figure 2. Each internal node in the tree represents a weighing such as, "put coins 1, 3, and 5 on one side and 2, 8, and 10 on the other side". Each leaf node represents a result such as, "the 11th coin is heavier than the rest". A run of the algorithm corresponds to a path from the root to a leaf. At each internal node, we perform the weighing associated with the node. Based on the result, we descend to one of the three children. When we reach a leaf, we output the indicated result.

Suppose that we wanted to solve the same problem with more coins, but still with only three weighings. The case with 13 coins is complicated and we'll put it off a bit. However, if there are 14 coins, then we can prove that no solution exists. This would seem to require an elaborate case analysis to rule out every possible strategy. Actually, we can use the Pigeonhole Principle to give a short proof that no weighing strategy exists for 14 coins.

Theorem 3.3. *The weighing problem cannot be solved with 14 coins and 3 weighings.*

Proof. The pigeons are the possible situations. Since any one of the 14 coins could be the counterfeit one and the counterfeit coin could be either heavier or lighter than the rest, there are 28 possible situations and so 28 pigeons. The holes are the 27 leaves of the depth-3 ternary tree used to represent the weighing strategy. Since there are more pigeons than holes, the Pigeonhole Principle implies that some hole must have two pigeons; that is, some leaf is not associated with a unique situation. Therefore, for every weighing strategy, there is some pair of situations that the strategy cannot distinguish. In at least one of these situations, the strategy must give the wrong answer. □

What if we had four weighings—could we identify the counterfeit coin from among 41? In this case, there are $2 \cdot 41 = 82$ possible situations or pigeons. Every four-weighing strategy can be represented by a depth four ternary tree with $3^4 = 81$ leaves or pigeonholes. Once again, the Pigeonhole Principle implies that in every weighing strategy, some leaf of the tree must be associated with two or more situations. The strategy cannot distinguish these situations and so must sometimes give the wrong answer.

In general, if there are n coins, we can compute a lower bound on the number of weighings necessary. Every strategy with w weighings can be represented by a ternary tree with 3^w leaves. In this case, there are $2n$ possible situations. For a correct weighing strategy to exist, there must be as many leaves as situations. That is, we must have $3^w \geq 2n$ or, equivalently, $w \geq \log_3 2n$.

Now we can address the tricky case of 13 coins and 3 weighings. In this case, we cannot use the Pigeonhole Principle directly to prove that there is no solution, because there are 26 situations and a depth three tree has 27 leaves. There may be a solution! But note that just because our Pigeonhole Principle argument does not rule out a weighing strategy, it does not follow that a such a strategy exists! In fact, a closer analysis shows that there is no solution.

[Optional]

Theorem 3.4. *The weighing problem cannot be solved with 13 coins and 3 weighings.*

To prove the claim, we must do some of the hairy case analysis that the Pigeonhole Principle lets us circumvent in the case of 14 coins. Nevertheless, appeals to the Pigeonhole Principle do greatly simplify the proof.

Proof. Let n be the number of coins placed in each pan of the balance scale on the first weighing. Since there are 13 coins, n could be 1, 2, 3, 4, 5, or 6. We can actually reduce these 6 cases to just two.

(To gather some intuition, first consider $n = 1$. Then there are 11 coins that were not used in the first weighing. If the first weighing came out equal, we have 22 possibilities left, but only two more weighings to use. But two weighings can only distinguish 9 cases. Then consider $n = 6$. Suppose that the right pan was heavier in the first weighing. That leaves 12 possibilities, either one of the left coins are light or one of the right coins are heavy. Again, we only have two weighings left. Now we just extend the above ideas to cover all n.)

In the first case, suppose that $n \leq 4$. This means that at least 5 coins are not used in the first weighing. As a result, there are at least 10 situations in which the scale will be balanced on the first weighing. (The counterfeit coin could be any of the unweighed coins, and it could be either heavy or light.) By the Pigeonhole Principle, we cannot distinguish between these 10 or more cases with only two additional weighings.

In the second case, suppose that $n \geq 5$. Now there are $2n \geq 10$ situations in which the right pan is heavier: n when the counterfeit coin is heavy and in the right pan and n more when the counterfeit coin is light and in the left pan. Again, by the Pigeonhole Principle, we cannot distinguish these 10 or more cases with only two weighings. □

4 The Cardinality of a Union of Sets

Suppose that we know the sizes of some sets $A_1, A_2, \ldots A_n$. How many elements are in the union? If the sets are disjoint, then the size of the union is given by the simple *Sum Rule*. If the sets are not necessarily disjoint, then the size is given by the more general, but more complicated *Inclusion-Exclusion Principle*.

4.1 The Sum Rule

Lemma 4.1. *If A and B are disjoint finite sets, then*

$$|A \cup B| = |A| + |B| .$$

Like the Pigeonhole Principle, Lemma 4.1 could be proved from the definition of cardinality, using properties of bijections and natural numbers. But again, the formal proof is really only of interest to Logicians. So we'll accept Lemma 4.1 without proof as an axiom. It generalizes straightforwardly to

Theorem 4.2 (Sum Rule). *If $A_1, A_2, \dots A_n$ are disjoint sets, then:*

$$|A_1 \cup A_2 \cup \dots \cup A_n| = |A_1| + |A_2| + \dots + |A_n|.$$

The Sum Rule says that the number of elements in a union of disjoint sets is equal to the sum of the sizes of all the sets. The Sum Rule can be proved from Lemma 4.1 by induction on the number of sets.

As an example use of the Sum Rule, suppose that MIT graduates 60 majors in math, 200 majors in EECS, and 40 majors in physics. How many students graduate from MIT in these three departments? Let A_1 be the set of math majors, A_2 be the set of EECS majors, and A_3 be the set of physics majors. The set of graduating students in these three departments is $A_1 \cup A_2 \cup A_3$. Assume for now that these sets are disjoint; that is, there are no double or triple majors. Then we can apply the Sum Rule to determine the total number of graduating students.

$$\begin{aligned} |A_1 \cup A_2 \cup A_3| &= |A_1| + |A_2| + |A_3| \\ &= 60 + 200 + 40 \\ &= 300 \end{aligned}$$

4.2 Inclusion-Exclusion Principle (special cases)

The Sum Rule gives the cardinality of a union of disjoint sets. The Inclusion-Exclusion Principle gives the cardinality of a union of sets that may intersect. The Inclusion-Exclusion Principle for n sets is messy to write down, so we'll start with the simple special cases $n = 2$ and $n = 3$.

Theorem 4.3 (Inclusion-Exclusion Principle for 2 sets). *Let A and B be sets, not necessarily disjoint.*

$$|A \cup B| = |A| + |B| - |A \cap B|$$

Here's a standard proof you'll find in many texts, including Rosen (p. 47):

Proof. Items in the union of A and B that are in the intersection of A and B are counted twice in the sum $|A| + |B|$. Therefore, by subtracting $|A \cap B|$, every element is counted once overall. □

Here's how to prove it without handwaving about "counting twice."

Lemma 4.4. *For any finite set, B, and set, A,*

$$|A \cap B| + |B - A| = |B|.$$

Proof. The definitions of intersection and set difference imply that

$$(A \cap B) \cup (B - A) = B,$$

and that $A \cap B$ and $B - A$ are disjoint. So Lemma 4.4 follows immediately by substituting $A \cap B$ for A and $B - A$ for B in Lemma 4.1. □

But now we can observe that $A \cup B = A \cup (B - A)$, and A and $B - A$ are disjoint, so

$$\begin{aligned} |A \cup B| &= |A| + |B - A| && \text{by Lemma 4.1} \\ &= |A| + (|B| - |A \cap B|) && \text{by Lemma 4.4} \\ &= |A| + |B| - |A \cap B| . \end{aligned}$$

Theorem 4.5 (Inclusion-Exclusion Principle for 3 sets). *Let A, B, and C be sets, not necessarily disjoint.*

$$\begin{aligned} |A \cup B \cup C| \quad = \quad & |A| + |B| + |C| \\ & - |A \cap B| - |A \cap C| - |B \cap C| \\ & + |A \cap B \cap C| \end{aligned}$$

Though this formula contains many terms, the general pattern is easy to remember: *add* the sizes of individual sets (first line), *subtract* intersections of pairs of sets (second line), and *add* the intersection of all three sets (third line).

Proof. Items contained in just one of the sets A, B, or C are counted once on the first line. Since these items are not contained in any intersection of sets, they are not subtracted away on the second line or counted again on the third line. In total, these items are counted just once.

Items contained in exactly two of the sets A, B, and C (not in all three) are counted twice on the first line, subtracted away once on the second line, and not counted again on the third line. Again, in total, these items are counted just once.

Items contained in all three sets A, B, and C are counted three times on the first line, subtracted away three times on the second line, and added back once on the third line. Since $3 - 3 + 1 = 1$, these items are also counted just once overall. □

The name "Inclusion-Exclusion" comes from the way items are counted, subtracted away, counted again, etc.

Earlier we applied the Sum Rule to count MIT graduates in math, EECS, and physics, assuming that there were no double or triple majors. Using Inclusion-Exclusion, we can count the number of graduates even if some people have multiple majors. (After all, this is MIT, where some students even have 4 majors...) Suppose the numbers are as follows:

60	math majors	$(\lvert A\rvert = 60)$
200	EECS majors	$(\lvert B\rvert = 200)$
40	physics majors	$(\lvert C\rvert = 40)$
10	math/EECS double majors	$(\lvert A \cap B\rvert = 10)$
4	math/physics double majors	$(\lvert A \cap C\rvert = 4)$
15	EECS/physics double majors	$(\lvert B \cap C\rvert = 15)$
2	triple majors	$(\lvert A \cap B \cap C\rvert = 2)$

We can compute the total number of graduates, $|A \cup B \cup C|$, by plugging numbers into the Inclusion-Exclusion formula:

$$\begin{aligned}
|A \cup B \cup C| &= |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C| \\
&= 60 + 200 + 40 - 10 - 4 - 15 + 2 \\
&= 273
\end{aligned}$$

4.3 Inclusion-Exclusion Principle (general case)

Here is the nasty Inclusion-Exclusion formula for n sets. The formula is given both in words and in symbols. The word version is the one to remember, but look over the symbolic version to make sure that you really know what the theorem says.

Theorem 4.6 (Inclusion-Exclusion for n sets). *Let $A_1, A_2, \ldots, A_n$ be sets, not necessarily disjoint. The cardinality of the union is computed as follows:*

> add *the sizes of all individual sets* subtract *the sizes of all two-way intersection* add *the sizes of all three-way intersections* subtract *the sizes of all four-way intersections* add *the sizes of all five-way intersections etc.*

Restated in symbols, the cardinality of the union is:

$$\begin{aligned}
|A_1 \cup A_2 \cup \ldots \cup A_n| &= \sum_{1 \le i \le n} |A_i| - \sum_{1 \le i < j \le n} |A_i \cap A_j| + \sum_{1 \le i < j < k \le n} |A_i \cap A_j \cap A_k| + \ldots \\
&\quad (-1)^{n+1} |A_1 \cap A_2 \cap \cdots \cap A_n| \\
&= \sum_{k=1}^{n} (-1)^{k+1} \sum_{S \subseteq \{1,\ldots,n\}, |S|=k} |\bigcap_{i \in S} A_i|
\end{aligned}$$

This final double summation is the usual form used in stating the Inclusion-Exclusion Principle, but it is sometimes useful to express it as an equivalent single sum:

$$|A_1 \cup A_2 \cup \ldots \cup A_n| = \sum_{\emptyset \neq S \subseteq \{1,\ldots,n\}} (-1)^{|S|+1} |\bigcap_{i \in S} A_i|$$

Even better, using the convention that an empty intersection equals the whole universe of discourse, that is, $\cap_{i \in \emptyset} A_i ::= \cup_{i=1}^{n} A_i$, we can write

$$\sum_{S \subseteq \{1,\ldots,n\}} (-1)^{|S|+1} |\bigcap_{i \in S} A_i| = 0$$

Theorem 4.6 can be proved by induction on n using the two-set version as a lemma. In the inductive step going from n to $n+1$, we observe that

$$\begin{aligned}
|A_1 \cup \cdots \cup A_n \cup A_{n+1}| &= |(A_1 \cup \cdots \cup A_n) \cup A_{n+1}| && (1) \\
&= |A_1 \cup \cdots \cup A_n| + |A_{n+1}| - |(A_1 \cup \cdots \cup A_n) \cap A_{n+1}| && (2) \\
&= |A_1 \cup \cdots \cup A_n| + |A_{n+1}| && (3) \\
&\quad - |(A_1 \cap A_{n+1}) \cup (A_2 \cap A_{n+1}) \cup \cdots \cup (A_n \cap A_{n+1})| && (4)
\end{aligned}$$

where equation (2) follows from two-set inclusion-exclusion, and the expression on line (4) follows from application of the distributive law for intersection over union. The whole final expression starting at line (3) involves two large unions, but each of these is a union of only n sets, so we can apply the inductive hypothesis to count them. We'll let the reader fill in the remaining details of the proof.

There is a different, slicker proof that can be given once we learn about binomial coefficients.

4.4 Counting Primes

How many of the numbers $1, 2, \ldots, 100$ are prime? One way to answer this question is to test each number up to 100 for primality and keep a count. This requires considerable effort. (Is 57 prime? How about 67?)

Another approach is to use the Inclusion-Exclusion Principle. This requires one trick: to determine the number of primes, we will first count the number of *non-primes*. We can then find the number of primes by subtraction. We will use this trick of "counting the complement" several times in coming weeks.

Reducing the Problem to the Cardinality of a Union

The set of non-primes in the range $1, \ldots, 100$ consists of the set C of composite numbers in this range ($4, 6, 8, 9, \ldots$, 99, 100) and the number 1, which is neither prime nor composite. The main job is to determine the size of the set C of composite numbers. For this purpose, define:

$$A_p = \{x \mid 1 \leq x \leq 100, p \mid x, \text{ and } x \neq p\}$$

In words, A_p is the set of numbers in the range $1, \ldots, 100$ that are divisible by p, but not equal to p. For example, $A_2 = \{4, 6, 8, \ldots, 100\}$.

Claim 4.7. $C = A_2 \cup A_3 \cup A_5 \cup A_7$

The claim explains the point of these funny A_p sets: we can write the set C of composite numbers as a union of them. We can compute the cardinality of the union using Inclusion-Exclusion, and this will tell us the number of composite numbers in the range $1, \ldots, 100$.

Proof. We prove the two sets equal by showing that each contains the other.

First, we show that $A_2 \cup A_3 \cup A_5 \cup A_7 \subseteq C$. Let n be an element of $A_2 \cup A_3 \cup A_5 \cup A_7$. Then $n \in A_p$ for $p = 2, 3, 5$ or 7. This implies that n is in the range $1, \ldots, 100$, n is divisible by p, and n is not equal to p. This implies that n is a composite in the range $1, \ldots, 100$, and so $n \in C$.

Second, we show that $C \subseteq A_2 \cup A_3 \cup A_5 \cup A_7$. Let n be an element of C. Then n is a composite number in the range $1, \ldots, 100$. This means that n has at least two prime factors p and q. One of these must be 2, 3, 5, or 7. (Otherwise, both p and q are at least 11, and so $n \geq pq \geq 11 \cdot 11 = 121$, a contradiction.) This implies that n is an element of A_2, A_3, A_5, or A_7, and so $n \in A_2 \cup A_3 \cup A_5 \cup A_7$. □

Corollary 4.8. $|C| = |A_2 \cup A_3 \cup A_5 \cup A_7|$

Computing the Cardinality of the Union

We have reduced the problem of counting primes to a problem about the cardinality of a union of sets. Specifically, we must evaluate $|A_2 \cup A_3 \cup A_5 \cup A_7|$. This will give us the number of composites in the range $1, \ldots, 100$, and from this we can figure out the number of primes. As a stepping stone, we can compute the cardinality of each set A_p:

$$|A_p| = \left\lfloor \frac{100}{p} \right\rfloor - 1$$

The first term, $\lfloor \frac{100}{p} \rfloor$, is the number of values in the range $1, \ldots, 100$ that are divisible by p. The second term, -1, arises because we defined A_p to exclude p itself. This formula gives:

$$\begin{aligned}
|A_2| &= \lfloor \tfrac{100}{2} \rfloor - 1 = & 49 \\
|A_3| &= \lfloor \tfrac{100}{3} \rfloor - 1 = & 32 \\
|A_5| &= \lfloor \tfrac{100}{5} \rfloor - 1 = & 19 \\
|A_7| &= \lfloor \tfrac{100}{7} \rfloor - 1 = & 13
\end{aligned}$$

Here is an erroneous way to compute the number of composites in the range 1 to 100.

$$\begin{aligned}
|C| &= |A_2 \cup A_3 \cup A_5 \cup A_7| \\
&= |A_2| + |A_3| + |A_5| + |A_7| \\
&= 49 + 32 + 19 + 13 \\
&= 113
\end{aligned}$$

In the first step, we applied the Sum Rule and the rest was substitution and simplification. The result is obviously wrong because there are only 100 numbers in the range 1 to 100! The problem is that the Sum Rule is inapplicable; the Sum Rule requires that the sets A_2, A_3, A_5, and A_7 be disjoint. However, 6 is in both A_2 and A_3, for example. Since the sets intersect, we must use the Inclusion-Exclusion Principle instead:

$$\begin{aligned}
|C| = \; & |A_2 \cup A_3 \cup A_5 \cup A_7| \\
= \; & |A_2| + |A_3| + |A_5| + |A_7| \\
& - |A_2 \cap A_3| - |A_2 \cap A_5| - |A_2 \cap A_7| - |A_3 \cap A_5| - |A_3 \cap A_7| - |A_5 \cap A_7| \\
& + |A_2 \cap A_3 \cap A_5| + |A_2 \cap A_3 \cap A_7| + |A_2 \cap A_5 \cap A_7| + |A_3 \cap A_5 \cap A_7| \\
& - |A_2 \cap A_3 \cap A_5 \cap A_7|
\end{aligned}$$

There are a lot of terms here! Fortunately, all of them are easy to evaluate. For example, $|A_3 \cap A_7|$ is the number of multiples of $3 \cdot 7 = 21$ in the range 1 to 100, which is $\lfloor \frac{100}{21} \rfloor = 4$. (Note that there is no reason to subtract 1 as we did when evaluating $|A_p|$ above.) Substituting values for all of the terms above gives:

$$\begin{aligned}
|C| &= 49 + 32 + 19 + 13 \\
&\quad - 16 - 10 - 7 - 6 - 4 - 2 \\
&\quad + 3 + 2 + 1 + 0 \\
&\quad - 0 \\
&= 74
\end{aligned}$$

This calculation shows that there are 74 composite numbers in the range 1 to 100. Since the number 1 is neither composite nor prime, there are $100 - 74 - 1 = 25$ primes in this range.

In retrospect, checking each number from 1 to 100 for primality and keeping a count of primes might have been easier! However, the Inclusion-Exclusion approach used here is asymptotically faster as the range of numbers grows large. The naive strategy requires N runs of a primality test if the upper bound is N. The Inclusion-Exclusion approach seems to require summing an immense number of terms, but fewer than N of these are non-zero and the rest can be ignored.

5 Products of Sets

Recall the definition of the Cartesian product of two sets:

$$A \times B = \{(a, b) \mid a \in A, b \in B\}$$

For example, if $A = \{x, y, z\}$ and $B = \{1, 2\}$, then $A \times B = \{(x, 1), (y, 1), (z, 1), (x, 2), (y, 2), (z, 2)\}$. In this case, A contains three items, B contains two items, and the product contains $3 \cdot 2 = 6$ items.

Exercise: Suppose $|A| = m$ and $|B| = n$. Prove that $|A \times B| = mn$ by defining a bijection from $\{1, 2, \ldots, mn\}$ to $A \times B$.

Solution: Arrange the elements in $A \times B$ as an $m \times n$ matrix with (a_i, b_j) in the ith row and jth column. Number the pairs in the first row, namely, $(a_1, b_1), \ldots (a_1, b_n)$, with numbers 1 through n. Number the pairs $(a_2, b_1), \ldots (a_2, b_n)$ in the second row with numbers $n + 1$ through $2n$. Continue numbering in this way through the mth row with pairs $(a_m, b_1), \ldots (a_m, b_n)$ numbered $(m-1)n+1$ through $(m - 1)n + n = mn$. This numbering defines the required bijection, f.

Another elegant, though perhaps more obscure way to describe f is by a formula:

$$f(i) ::= (a_{\lfloor (i-1)/n \rfloor + 1}, b_{((i-1) \bmod n) + 1}).$$

More generally, the size of a product of sets is given by the Product Rule:

Theorem 5.1 (Product Rule). *If $A_1, A_2, \ldots A_n$ are sets, then:*

$$|A_1 \times A_2 \times \cdots \times A_n| = |A_1| \cdot |A_2| \cdot \cdots \cdot |A_n|$$

The proof can be done by induction on n, but we leave it to the reader.

Here is another way to look at the product rule: the number of ways to pick one item from A_1, 1 item from A_2, ..., and 1 item from A_n is $\prod_{i=1}^{n} |A_i|$.

The Product Rule yields a useful generalization of the Pigeonhole Principle:

Theorem 5.2 (Generalized Pigeonhole Principle). *If there are m pigeons and n holes, then at least one hole contains $\lceil m/n \rceil$ pigeons.*

Proof. Let p_i be the number of pigeons in the ith hole, and assume that $p_i < \lceil m/n \rceil$ for $1 \leq i \leq n$. But since p_i is an integer, this implies that $p_i < m/n$. By the Product Rule, the total number of pigeons in the holes is $p_1 \cdot p_2 \dots p_n < (m/n) \cdot n = m$. But all m pigeons are supposed to be in the holes, a contradiction. So some p_i must not be less than $\lceil m/n \rceil$. □

Even this Generalized Pigeonhole Priniciple is so obvious that we often take it for granted without explicitly saying so. For example, if you look back in Week 3 Notes at Dilworth's theorem about the number of chains and antichains in a partial order, you will see we aleady used the Generalized Pigeonhole Principle in proving it.

Four-Course Italian Meals

As an example of the Product Rule, suppose that an Italian restaurant menu lists 15 antipasti, 6 pastas, 10 main courses, and 4 desserts. How many four course-meals are possible? In this case, A_1 is the set of antipasti, A_2 is the sets of pastas, A_3 is the set of main courses, and A_4 is the set of desserts. The number of four-course meals is the number of ways of picking one item from each set. By the Product Rule this is:

$$\begin{aligned} |A_1 \times A_2 \times A_3 \times A_4| &= |A_1| \cdot |A_2| \cdot |A_3| \cdot |A_4| \\ &= 15 \cdot 6 \cdot 10 \cdot 4 \\ &= 3600 \end{aligned}$$

Now suppose there are 7201 guests at a banquet catered by the restaurant. Then there must be at least three guests who choose exactly the same meal. This follows from the Generalized Pigeonhole Principle, letting guests be pigeons, four-course meals be holes, and noting that $\lceil 7201/3600 \rceil = 3$.

Binary Strings

How many n-bit binary strings are there? If we let $B = \{0, 1\}$, then the set of n-bit binary strings is:

$$\underbrace{B \times B \times \cdots \times B}_{n \text{ terms}}$$

By the Product Rule, the number of binary strings is $|B|^n = 2^n$.

Telephone Numbers

How many telephone numbers are there? There are actually two formats for telephone numbers, an old one and a new one. The formats can be defined in terms of three sets of digits:

$$\begin{aligned} A &= \{0, 1, \ldots, 9\} \\ B &= \{2, 3, \ldots, 9\} \\ C &= \{0, 1\} \end{aligned}$$

The old format is $(BCA)\ BBA - AAAA$ and the new format is $(BAA)\ BAA - AAAA$. This gives:

$$\begin{aligned} \text{old numbers} &= |B| \cdot |C| \cdot |A| \ \cdot \ |B| \cdot |B| \cdot |A| \ \cdot \ |A| \cdot |A| \cdot |A| \cdot |A| \\ &= 2 \cdot 8^3 \cdot 10^6 \\ &= 1.024 \text{ billion} \\ \text{new numbers} &= |B| \cdot |A| \cdot |A| \ \cdot \ |B| \cdot |A| \cdot |A| \ \cdot \ |A| \cdot |A| \cdot |A| \cdot |A| \\ &= 8^2 \cdot 10^8 \\ &= 6.4 \text{ billion} \end{aligned}$$

The reason for the new format is that there are more numbers. In particular, the number of area codes is increased from $8 \cdot 2 \cdot 10 = 160$ to $8 \cdot 10 \cdot 10 = 800$.

Passwords

Suppose that a password consists of 8 characters where each character is either a number or a lowercase letter. A password is *legal* if there is at least one number and at least one letter. How many legal passwords are there?

We can solve this problem by “counting the complement”. We used this method earlier to find the number of primes in the range 1 to 100 by counting the composites in this range. In this case, we can find the number of legal passwords by counting the illegal passwords. An illegal password has either all numbers or all letters. By the Product Rule, the number of passwords with all numbers is 10^8, and the number of passwords with all letters is 26^8. Also by the Product Rule, the total number of passwords (both legal and illegal) is 36^8. Therefore, the total number of legal passwords is $36^8 - 26^8 - 10^8$.

6 Tree Diagrams

The Sum and Product Rules are both useful in counting the number of ways that something can be done. In more complicated problems, we need to combine these rules. In such cases, *tree diagrams* are helpful.

6.1 A Problem Getting Dressed

Suppose we have 3 blue shirts, 2 red shirts, and 1 green shirt. We also have 2 gray pants and 3 brown pants. How many outfits are possible? (Two pieces of clothing with the same color are still considered distinct; assume that they have slightly different shades.)

This problem is easier than it sounds. Let S be the set of shirts and P be the set of pants. We form an outfit by picking one shirt and one pair of pants. By the Product Rule, there are $6 \cdot 5 = 30$ outfits. All the color information is irrelevant.

However, suppose that gray pants "go with" only blue and red shirts and brown pants "go with" only green and red shirts. How many *matching* outfits are there?

One approach to this problem is to count the complement set. That is, we count the number of matching outfits by counting the number of mismatching outfits. By the Product Rule, there are $2 \cdot 1 = 2$ mismatching gray-green outfits and $3 \cdot 3 = 9$ mismatching brown-blue outfits. Therefore, there are $30 - 2 - 9 = 19$ matching outfits.

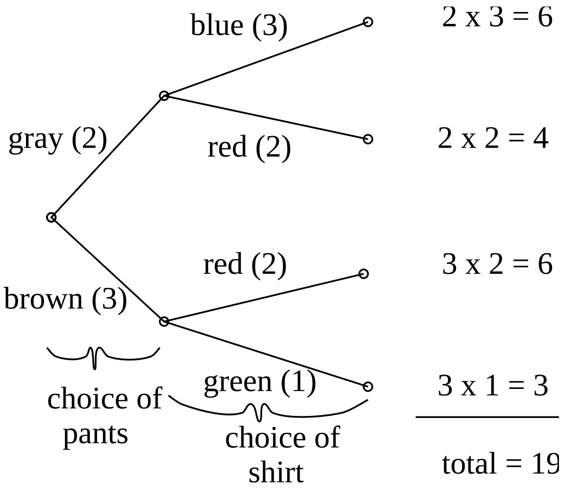


Figure 3: *This is a tree diagram for the matching outfits problem. Edge weights are shown in parentheses.*

A second approach is to use a tree diagram like the one shown in Figure 3. Tree diagrams are useful in counting the number of possible outcomes arising from a sequence of decisions. In the simplest case, each internal node of a tree diagram represents a decision, each child corresponds to an available choice, and each leaf is associated with an outcome. The choice selected at one internal node determines what decisions must be made subsequently. For example, if we choose gray pants, then we must subsequently decide on a blue or red shirt; however, if we choose brown pants, then we must subsequently decide on a green or red shirt. The tree diagram in Figure 3 is a little more complicated because edges have weights; these correspond to distinct choices that lead to the same set of future decisions. For example, we can choose either of the two pairs of gray pants without affecting our subsequent choice of a shirt.

The number of matching outfits can be computed from the tree diagram as follows. A root-to-leaf path corresponds to a sequence of choices such as "gray pants, blue shirt". The number of distinct outcomes of this type can be computed by the Product Rule; we just multiply the weights of the edges on the path. For example, the number of ways to choose a pair of gray pants and a blue shirt is $2 \cdot 3 = 6$. Then we can use the Sum Rule to compute the number of outcomes of all types;

we just add up all the products. This gives $6 + 4 + 3 + 6 = 19$ matching outfits. This is the same number we found by the method of counting the complement. (Good thing!)

Tree diagrams are a more sophisticated example of the case analysis we did for phone numbers. They use cases to break up the top level set, then use more cases to break up each subset, and so on.

6.2 Playoff Outcomes

In how many ways can a 5-game playoff series be decided? That is, how many different sequences of wins and losses are there? A first answer might use the product rule and say $2^5 = 32$, but this count includes certain outcomes that can never happen. For example, one possibility is that the home team wins a game, then loses one, and then wins two more. In this case, there would be no need for a fifth game.

We can solve this problem with the tree diagram shown in Figure 4. In this case, there are no edge weights. (Equivalently, one can say that the edges all have weight 1.) The number of outcomes is equal to the number of leaves, which is 20.

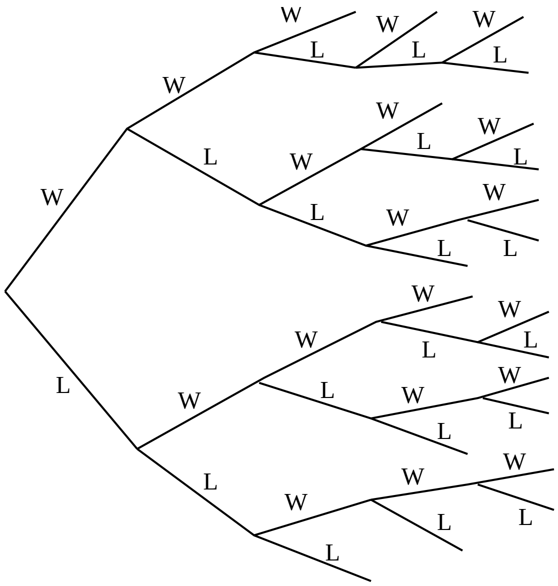


Figure 4: *This is a tree diagram for the number of ways to decide a 5-game playoff. W and L indicates wins and losses for the home team.*

In a larger problem, like the 7-game World Series, the tree diagram would be unmanageable. As we will soon see, however, sometimes we can observe a pattern and count the number of leaves without drawing out the complete tree.

7 Permutations

7.1 Simple Permutations

In how many ways can n items be ordered in a line? Each such ordering is called a *permutation*. So the question can be restated as, "How many permutations are there of n items?"

It is unmanageable to solve this problem by drawing a complete tree diagram. However, the partial diagram shown in Figure 5, suggests that there are n choices for the first item, $n-1$ choices for the second item given the first, $n-2$ choices for the third item given the first two, etc. Therefore, the total number of leaves in the tree is $n(n-1)(n-2)\ldots 2 \cdot 1 = n!$. This gives an important fact; never forget it!

We can formalize this by observing that if $P(n)$ is the number of permutations of n elements, then $P(n) = n \cdot P(n-1)$ because there are n ways to pick the first element and then $P(n-1)$ ways to permute the remaining $n-1$ items. This gives $P(n) = n!$ by induction.

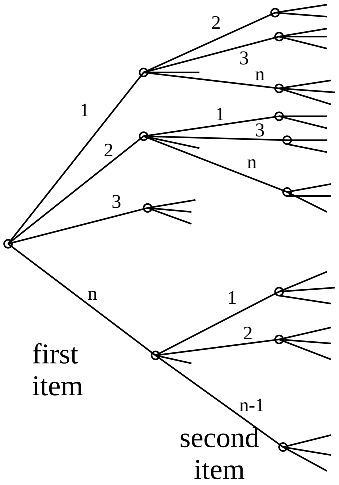


Figure 5: *This is a partial tree diagram for counting the number of ways n items can be ordered in a line. The diagram suggests that there are n choices for the first item, $n-1$ choices for the second, $n-2$ for the third, and so forth.*

Fact 7.1. The number of permutations of n items is $n!$.

For example, the number of ways to order a deck of cards is $52!$. This is clearly a very big number; we can estimate how big with Stirling's Formula:

$$\begin{aligned} 52! &\geq \sqrt{2\pi 52}\left(\frac{52}{e}\right)^{52} e^{\dfrac{1}{12 \cdot 52 + 1}} \\ &= 8.05 \cdots \cdot 10^{67} \end{aligned}$$

This is more than the number of atoms in the universe! This is more than the number of induction proofs in 6.042! Such rapid growth in the number of possibilities in combinatorial problems gives rise to the term "combinatorial explosion". There is no way a computer could ever try all these arrangements!

7.2 A Lower Bound for Sorting

The fact that there are $n!$ ways to order n items can be combined with the Pigeonhole Principle to prove a nice lower bound on the number of comparisons needed to sort a list of n items.

Theorem 7.2. *Any* comparison-based *sorting algorithm must make at least the following number of comparisons to sort n items in the worst case:*

$$n \log_2 n - n \log_2 e$$

In a comparison-based sorting algorithm, we can compare two items and move items around in memory. All other operations on items, such as looking at certain bits or performing arithmetic on items, are ruled out. For example, there is a sorting algorithm called "bucket sort" that is not comparison-based and for which this theorem does not apply.

The idea behind the proof is the same as in the twenty-questions game in Section 3.3 above. There we compared the number of possible locations of the counterfeit coin with the number of leaves in a decision tree defined by our weighing scheme. Here we compare the number of initial permutations of items with the number of leaves in a computation tree defined by the sorting algorithm.

Proof. Let $x_1, \ldots, x_n$ be the items to be sorted. Since we are proving a lower bound, we can assume that the items are all distinct.

Let $\mathcal{A}$ be any sorting algorithm. Algorithm $\mathcal{A}$ must permute the items $x_1, \ldots, x_n$ to put them in the correct order. Since a different permutation is required for each different initial ordering of the items, the number of different permutations that might be required is $n!$.

The computations performed by algorithm $\mathcal{A}$ can be modeled with a tree diagram as shown in Figure 6. Each internal node corresponds to a comparison of two items. The two subtrees beneath each internal node define the computations performed based on the result of the comparison. A run of the algorithm corresponds to a root-to-leaf path. Associated with each leaf is the permutation of $x_1, \ldots, x_n$ that places the items in sorted order.

We can assume that every node in the tree diagram is reached for some input. If a node were unreachable (say, because both $x_1 < x_2$ and $x_1 > x_2$ must hold to reach it), then we could remove that node from the tree.

Every one of the $n!$ permutations of $x_1, \ldots, x_n$ must appear at some leaf. Otherwise, if items are ordered according to some missing permutation, then the algorithm cannot possibly output the correct answer.

Let L be the number of leaves in the computation tree. We will obtain both an upper bound and a lower bound on L.

Permutations are assigned in some way to leaves. Regard the permutations as pigeons and the leaves as holes. If $L < n!$, then by the Pigeonhole Principle, two permutations are assigned to the same leaf. This is a contradiction, since the algorithm always permutes the items one particular way. Therefore $L \geq n!$.

On the other hand, the number of leaves is limited by the depth of the computation tree. Let D be the depth of the tree; that is, D is the length of the longest root-to-leaf path. Early in the course, we proved that the number of leaves in a binary tree with depth D is at most 2^D. Therefore, $L \leq 2^D$.

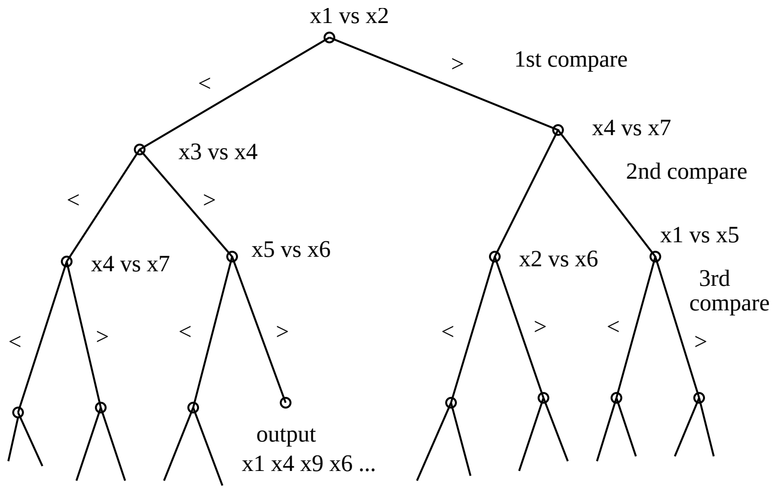


Figure 6: *This is how the computation tree of a sorting algorithm might look.*

Since D is the length of the longest root-to-leaf path in the computation tree, D is the number of comparisons used by algorithm $\mathcal{A}$ in the worst case. We can get a lower bound on D by putting together the upper and lower bounds on the number of leaves L.

$$\begin{aligned} 2^D \geq L &\geq n! \\ D &\geq \log_2 n! \end{aligned}$$

Now we can use Stirling's Formula to make sense of $\log_2 n!$:

$$\begin{aligned} D &\geq \log_2\left(\sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\dfrac{1}{12n+1}}\right) \\ &= n\log_2 n - n\log_2 e + \frac{1}{2}\log_2(2\pi n) + \frac{\log_2 e}{12n+1} \\ &\geq n\log_2 n - n\log_2 e \end{aligned}$$

□

How tight is this lower bound? The Merge Sort algorithm is a familiar, comparison-based sorting algorithm covered in most algorithms texts that is known to take at most $n\log_2 n - n + 1$ comparisons.[1] Our lower bound implies that at least $n\log_2 n - 1.45n$ comparisons are necessary in the worst case. The difference between this achievable number of comparisons and out lower bound is only about $n/2$. So our bound cannot be improved by more than the low-order term $n/2$. This also follows that Merge Sort cannot be improved by more than this low-order term—Merge Sort is very nearly optimal!

[1] Rosen, p 569.

8 *r*-Permutations

In a horse race, the first-place horse is said to "win", the second-place horse is said to "place", and the third-place horse is said to "show". A bet in which one guesses exactly which horses will win, place, and show is called a Trifecta. How many possible Trifecta bets are there in a race with 10 horses? This problem is easy enough that we do not need to draw out the tree diagram; there are 10 choices to win, 9 choices to place given the winner, and 8 choices to show given the first two finishers. This gives $10 \cdot 9 \cdot 8 = 720$ possible Trifecta bets.

This is a special case of a standard problem: counting r-permutations of a set.

Definition 8.1. An *r-permutation* of a set is an ordering of r distinct items from the set.

For example, the 2-permutations of the set $\{A, B, C\}$ are:

$$\begin{array}{ccc} (A,B) & (A,C) & (B,C) \\ (B,A) & (C,A) & (C,B) \end{array}$$

The number of r-permutations of an n-element set comes up so often in combinatorial problems that there is a special notation.

Definition 8.2. $P(n,r)$ denotes the number of r-permutations of an n-element set. In other words,

$$P(n,r) = n(n-1)\dots(n-r+1) = \frac{n!}{(n-r)!}$$

For example, the number of Trifecta bets in a 10 horse race is $P(10,3) = \frac{10!}{7!} = 720$.

Example 8.3. How many strings of 5 letters are there with no repetitions? For example, ZYGML, is such a string. In general, these strings are exactly the 5-permutations of the set of letters. Therefore, there are $P(26,5) = \frac{26!}{21!}$, which is 7,893,600.

Example 8.4. How many ways can 5 students be chosen from a class of 180 to be given 5 fabulous (different) prizes, say \$1000, \$500, \$250, \$100, \$50? Answer: $P(180,5)$, which is $180 \times 179 \times 178 \times 177 \times 176$.

Example 8.5. How many *injective* functions are there from A to B, if $|A| = n$ and $|B| = m$ (assume $n \leq m$). Answer: Order A arbitrarily. The first element maps somewhere in B. The second element maps somewhere else (injective), etc. The number of possibilities is $P(m,n) = m(m-1)(m-2)\dots(m-n+1) = \frac{m!}{(m-n)!}$. Essentially this is picking a sequence of n of the m elements of B.

Massachusetts Institute of Technology
6.042J/18.062J, Fall '02: Mathematics for Computer Science
Prof. Albert Meyer and Dr. Radhika Nagpal

Permutations and Combinations

In Notes 8, we saw a variety of techniques for counting elements in a finite set: the Sum Rule, Inclusion-Exclusion, the Product Rule, tree diagrams, and permutations. We will now introduce yet another rule, the *Division Rule*, and one more concept, *combinations*. We will also learn techniques for counting elements of a finite set when limited repetition is allowed.

1 The Division Rule

The division rule is a common way to ignore "unimportant" differences when you are counting things. You can count distinct objects, and then use the division rule to "merge" the ones that are not significantly different.

We will state the Division Rule twice, once informally and then again with more precise notation.

Theorem 1.1 (Division Rule). *If B is a finite set and $f : A \mapsto B$ maps precisely k items of A to every item of B, then A has k times as many items as B.*

For example, suppose A is a set of students, B is a set of tutorials, and f defines the assignment of students to tutorials. If 12 students are assigned to every tutorial, then the Division Rule says that there are 12 times as many students as tutorials.

The following two definitions permit a more precise statement of the Division Rule.

Definition 1.2. If $f : A \mapsto B$ is a function, then $f^{-1}(b) = \{a \in A \mid f(a) = b\}$.

That is, $f^{-1}(b)$ is the set of items in A that are mapped to the item $b \in B$. In the preceding example, $f^{-1}(b)$ is the set of students assigned to tutorial b.

This notation can be confusing, since f^{-1} normally denotes the inverse of the function f. With our definition, however, $f^{-1}(b)$ can be a set, not just a single value. For example, if f assigns no items in A to some element $b \in B$, then $f^{-1}(b)$ is the empty set. In the special case where f is a bijection, $f^{-1}(b)$ is always a single value, and so f^{-1} by our definition is just the ordinary inverse of f.

Definition 1.3. A function $f : A \mapsto B$ is *k-to-1* if for all $b \in B$, $|f^{-1}(b)| = k$.

For example, if f assigns exactly 12 students to each recitation, then f is 12-to-1. Assuming k is non-zero, a k-to-1 function is always a surjection; every element of the range is mapped to by $k > 0$ elements of the domain.

We can now restate the Division Rule more precisely:

Theorem (Division Rule, restatement). *If B is a finite set and $f : A \mapsto B$ is k-to-1, then $|A| = k|B|$.*

Proof. Since B is finite, we can let $n = |B|$ and let $B = \{b_1, b_2, \ldots, b_n\}$. Then we have:

$$A = f^{-1}(b_1) \cup f^{-1}(b_2) \cup \cdots \cup f^{-1}(b_n)$$

Equality holds because each side is contained in the other. The right side is contained in the left side because every set $f^{-1}(b_i)$ is contained in A by the definition of $f^{-1}(b_i)$. The left side is contained in the right, since every item in A maps to some b_i, and therefore is contained in the set $f^{-1}(b_i)$.

Furthermore, all of the sets $f^{-1}(b_i)$ are disjoint. The proof is by contradiction. Assume for the purpose of contradiction that $a \in f^{-1}(b_i)$ and $a \in f^{-1}(b_j)$ for some $i \neq j$. The first inclusion implies $f(a) = b_i$, but the second inclusion implies that $f(a) = b_j$. This is a contradiction since $f(a)$ denotes a unique item.

Since the sets $f^{-1}(b_i)$ are disjoint, we can compute the cardinality of their union with the Sum Rule. Note that each set $f^{-1}(b_i)$ has size k, since f is k-to-1.

$$\begin{aligned}
|A| &= |f^{-1}(b_1)| + |f^{-1}(b_2)| + \cdots + |f^{-1}(b_n)| \\
&= \underbrace{k + k + \cdots + k}_{n=|B| \text{ terms}} \\
&= k|B|
\end{aligned}$$

□

Example: Seating at a Round Table. In how many ways can King Arthur seat n knights at his round table? Two seatings are considered equivalent if one can be obtained from the other by rotation. For example, if Arthur has only four knights, then there are six possibilities as shown in Figure 1. Hereafter, we denote a seating arrangement in text by listing knights in square brackets in clockwise order, starting at an arbitrary point. For example, [1234] and [4123] are equivalent.

As a sign that we should bring in the division rule, the question points out that certain distinct seatings are equivalent, so should only be counted once—that is, we want to count equivalence classes instead of individual objects. The division rule is a good way to do this.

Claim 1.4. *Arthur can seat n knights at his round table in $(n-1)!$ ways.*

In particular, the claim says that for $n = 4$ knights there are $(4-1)! = 3! = 6$ orderings, which is consistent with the example in Figure 1.

Proof. The proof uses the Division Rule. Let A be the set of orderings of n knights in a line. Let B be the set of orderings of n knights in a ring. Define $f : A \mapsto B$ by $f((x_1, x_2, \ldots, x_n)) = [x_1 x_2 \ldots x_n]$.

The function f is n-to-1. In particular:

$$\begin{aligned}
f^{-1}([x_1 x_2 \ldots x_n]) = \{ \quad & (x_1, x_2, x_3, \ldots, x_n), \\
& (x_n, x_1, x_2, \ldots, x_{n-1}) \\
& (x_{n-1}, x_n, x_1, \ldots, x_{n-2}), \\
& \ldots \\
& (x_2, x_3, x_4, \ldots, x_1)\}
\end{aligned}$$

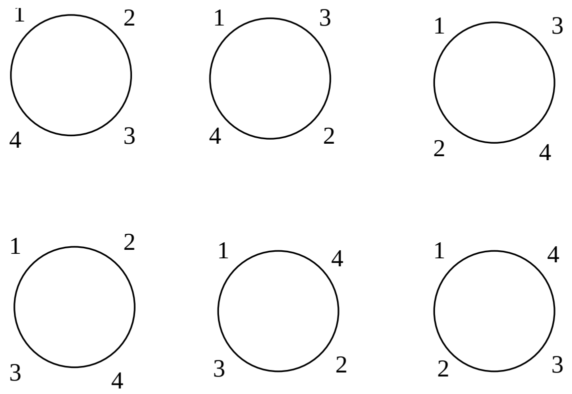


Figure 1: *These are the 6 different ways that King Arthur can seat 4 knights at his round table. Two seatings differing only by rotation are considered equivalent. We denote a seating arrangement in text by listing knights in square brackets in clockwise order, starting at an arbitrary point. For example, the seatings in the top row can be denoted* [1234], [2413], *and* [3421].

There are n tuples in the list of tuples, because x_1 can appears in n different places in a tuple. By the Division Rule, $|A| = n|B|$. This gives:

$$\begin{aligned} |B| &= \frac{|A|}{n} \\ &= \frac{n!}{n} \\ &= (n-1)! \end{aligned}$$

The second equality holds, because the number of orderings of n items in a line is $n!$, as we showed previously. □

2 Combinations

We now use the Division Rule to find a formula for the number of *combinations* of elements from a set. One common pitfall in counting problems is mixing up combinations with permutations; try to keep track of which is which!

Example: Students and prizes. How many groups of 5 students can be chosen from a 180-student class to collaborate on a problem set?

Let A be the set of 5-permutations of the set of students. This number is $P(180, 5) = 180 \times 179 \times 178 \times 177 \times 176$. But that's not really what we want—order of the 5 students in a group doesn't matter. So let B be the set of 5-student groups. This is what we want. Let's define a map $f : A \Rightarrow B$ by ignoring the order. We claim that f is $5!$ to 1, in other words, 120 to 1. This is based on the number of ways that 5 students could be lined up. So, by the Division Rule,

$$|B| = \frac{|A|}{120} = \frac{180 \times 179 \times 178 \times 177 \times 176}{120}.$$

2.1 The *r*-Combinations of a Set

Generalizing the example that we just did, the Division Rule can be used to count the number of r-combinations of a set.

Definition 2.1. An r*-combination* of a set is a subset of size r.

For example, for the set $\{a, b, c\}$, we have the following three 2-combinations:

$$\{a, b\} \quad \{a, c\} \quad \{b, c\}$$

Understanding the distinction between r-combinations and r-permutations is important. The r-combinations of a set are different from r-permutations in that order does not matter. For example, the above set $\{a, b, c\}$ has six 2-permutations:

$$(a, b) \quad (a, c) \quad (b, c) \quad (b, a) \quad (c, a) \quad (c, b)$$

The collaboration groups of 5 students in previous example are the 5-combinations of the set of students in the class.

2.2 The number of *r*-Combinations (Binomial Coefficients)

The number of r-combinations of an n-element set comes up in many problems. The quantity is important enough to merit two notations. The first is $C(n, r)$; this is analogous to the notation $P(n, r)$ from before for the number of r permutations of an n-element set. The second notation is $\binom{n}{r}$, read "n choose r". The values $\binom{n}{r}$ or $C(n, r)$ are often called *binomial coefficients* because of their prominent role in the Binomial Theorem, which we will cover shortly.

The following theorem gives a closed form for $\binom{n}{r}$. This theorem is used all the time; remember it!

Theorem 2.2.

$$\binom{n}{r} = \frac{n!}{(n-r)!\, r!}$$

The theorem says, for example, that the number of 2-combinations of the three element set $\{a, b, c\}$ is $\frac{3!}{(3-2)!\, 2!} = 3$ as we saw above.

Also, for example, the number of 5-combinations of the 180 element set of students is $\frac{180!}{175!5!}$.

Proof. The proof uses the Division Rule. Let X be a set with n elements. Let A be the set of r-permutations of X, and let B be the set of r-combinations of X. Our goal is to compute $|B|$. To this end, let $f : A \mapsto B$ be the function mapping r-permutations to r-combinations that is defined by:

$$f(\ \underbrace{(x_1, x_2, \ldots, x_r)}_{\substack{\text{order matters}\\ \text{(permutation)}}}\ ) = \underbrace{\{x_1, x_2, \ldots, x_r\}}_{\substack{\text{order does not matter}\\ \text{(combination)}}}$$

The function f is $r!$-to-1. In particular:

$$\begin{aligned} f^{-1}(\{x_1, x_2, \ldots, x_r\}) &= \{(x_1, x_2, x_3, x_4, \ldots, x_r), \\ &\quad (x_2, x_1, x_3, x_4, \ldots, x_r), \\ &\quad \ldots \text{all } r! \text{ permutations} \ldots\} \end{aligned}$$

By the Division Rule, $|A| = r!\,|B|$. Now we can compute the number of r-combinations of an n-element set, B, as follows:

$$\begin{aligned} |B| &= \frac{|A|}{r!} \\ &= \frac{P(n,r)}{r!} \\ &= \frac{n!}{(n-r)!\,r!} \end{aligned}$$

□

2.3 Some interesting special cases

Binomial coefficients are important enough that some special values of $\binom{n}{r}$ are worth remembering:

$$\begin{aligned} \binom{n}{0} &= \frac{n!}{n!\,0!} &= 1 &\quad \text{There is a single size 0 subset of an } n\text{-set.} \\ \binom{n}{n} &= \frac{n!}{0!\,n!} &= 1 &\quad \text{There is a single size } n \text{ subset of an } n\text{-set.} \\ \binom{n}{1} &= \frac{n!}{(n-1)!\,1!} &= n &\quad \text{There are } n \text{ size 1 subsets of an } n\text{-set.} \\ \binom{n}{n-1} &= \frac{n!}{1!\,(n-1)!} &= n &\quad \text{There are } n \text{ size } (n-1) \text{ subsets of an } n\text{-set.} \end{aligned}$$

Recall that $0!$ is defined to be 1, not zero.

Theorem 2.3. *For all $n \geq 1$, $0 \leq r \leq n$,*

$$\binom{n}{r} = \binom{n}{n-r}.$$

Theorem 2.3 can be read as saying that the number of ways to choose r elements to form a subset of a set of size n is the same as the number of ways to choose the $n - r$ elements to exclude from the subset. Put that way, it's obvious. This is what is called a *combinatorial proof* of an identity: we write expressions that reflect different ways of counting or constructing the same set of objects and conclude that the expressions are equal. All sorts of complicated-looking and apparently obscure identities can be proved—and made clear—in this way.

Of course we can also prove Theorem 2.3 by simple algebraic manipulation:

$$\binom{n}{r} = \frac{n!}{(n-r)!\,r!} = \frac{n!}{r!\,(n-r)!} = \frac{n!}{(n-(n-r))!\,(n-r)!} = \binom{n}{n-r}.$$

3 Counting Poker Hands

In the poker game Five-Card Draw, each player is dealt a hand consisting of 5 cards from a deck of 52 cards. Each card in the deck has a suit (clubs $\clubsuit$, hearts $\heartsuit$, diamonds $\diamondsuit$, or spades $\spadesuit$) and a value $(A, 2, \ldots, 10, J, Q, K)$. The order in which cards are dealt does not matter. We will count various types of hands in Five-Card Draw to show how to use the formula for r-combinations given in Theorem 2.2.

It may not be clear why we want to count the number of types of hands. The number of hands of a particular type tells us how "likely" we are to encounter such a hand in a "random" deal of the cards. Hands that are less likely are ranked higher than those that are more likely, and so will win over the more likely ones. Knowing the likelihood (or the number) of various hands can help us to figure out how to bet on the game.

3.1 All Five-Card Hands

How many different hands are possible in Five-Card Draw? A hand is just a 5-card subset of the 52-card deck. Therefore, the possible hands in Five-Card Draw are exactly the 5-combinations of a 52-element set. By Theorem 2.2, there are $\binom{52}{5} = \frac{52!}{47!\,5!} = 2,598,960$ possible hands.

Confusion between combinations and permutations is a common source of error in counting problems. For example, one might erroneously count the number of 5-card hands as follows. There are 52 choices for the first card, 51 choice for the second card given the first, ..., and 48 choices for the fifth card given the first four. This totals $52 \cdot 51 \cdot 50 \cdot 49 \cdot 48$ possible hands. The problem with this way of counting is that a hand is counted once for every ordering of the five cards; that is, each hand is counted $5! = 120$ times. We accidentally counted r-permutations instead of r-combinations! To get the number of 5-combinations, we would have to divide by 120.

3.2 Hands with Four-of-a-Kind

How many hands are there with four-of-a-kind? For example, $9\spadesuit\ 4\diamondsuit\ 9\clubsuit\ 9\heartsuit\ 9\diamondsuit$ has a four-of-a-kind, because there are four 9's.

First we must choose one value to appear in all four suits from the set of 13 possible values (this follows the "tree diagram" approach to exploring distinct cases). There are 13 choices of this value, and if we pick one we can't have any others (that would require 8 cards). So the choice of value gives 13 disjoint cases to count. After this choice, we have to choose one more card from the remaining set of 48. This can be done in $\binom{48}{1} = 48$ ways. In total, there are $13 \cdot 48 = 624$ hands with four-of-a-kind. The situation is illustrated by the tree diagram in Figure 2.

Here is a way to interpret the result that uses probability. If all $\binom{52}{5}$ hands are equally likely, then the probability that we are dealt a four-of-a-kind is

$$\frac{624}{\binom{52}{5}} = \frac{1}{4165}.$$

In other words, we can only expect to get four-of-a-kind only once in every 4165 games of Five-Card Draw! (This is just a passing observation; we will talk about probability "for real" in a few weeks.)

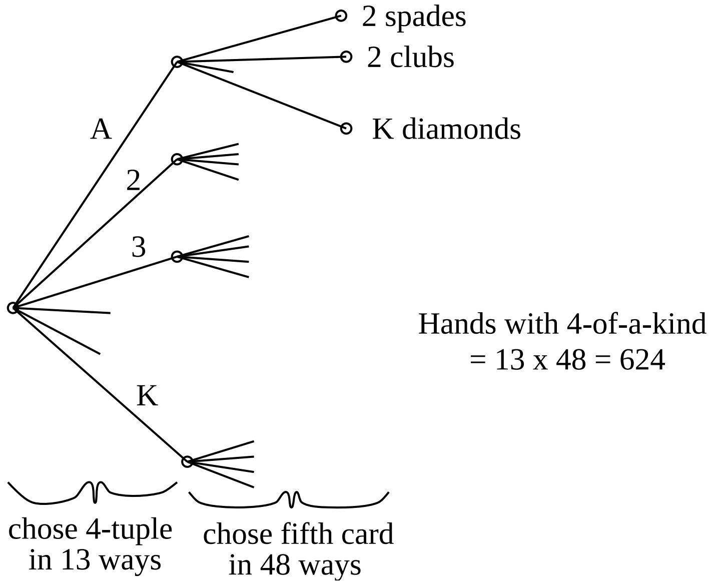


Figure 2: *This (incomplete) tree diagram counts the number of hands with four-of-a-kind. There are 13 choices for the value appearing in all four suits. The fifth card in the hand can be any of the 48 cards remaining in the deck. This gives a total of* $13 \cdot 48 = 624$ *possible hands.*

Mistakes are easy to make in counting problems. It is a good idea to check a result by counting the same thing in another way. For example, we could also count the number of hands with a four-of-a-kind as follows. First, there are 52 ways to choose the "extra" card. Given this, there are 12 ways to choose the value that appears in all four suits. (We cannot choose the value of the first card.) The possibilities are illustrated by the tree diagram in Figure 3. By this method, we find that there are $52 \cdot 12 = 624$ four-of-a-kind hands. This is consistent with our first answer (which is a good thing).

3.3 Hands with a Full House

A full house is a hand with both a three-of-a-kind and a two-of-a-kind. For example, the hand $7\spadesuit\ 7\diamondsuit\ J\clubsuit\ J\heartsuit\ J\diamondsuit$ is a full house because there are three jacks and two sevens. How many full house hands are there?

We can choose the value that appears three times in 13 ways. Then we can pick any three of the four cards in the deck with this value; this can be done in $\binom{4}{3} = 4$ ways. There are then 12 remaining choices for the value that appears two times. We can pick any two of the four cards with this value; this can be done in $\binom{4}{2} = 6$ ways.

The total number of full house hands is therefore $13 \cdot 4 \cdot 12 \cdot 6 = 3744$. This is 6 times greater than the number of hands with a four-of-a-kind. Since a four-of-a-kind is rarer, it is worth more in poker!

3.4 Hands with Two Pairs

How many hands are there with two pairs, but no three- or four-of-a-kind?

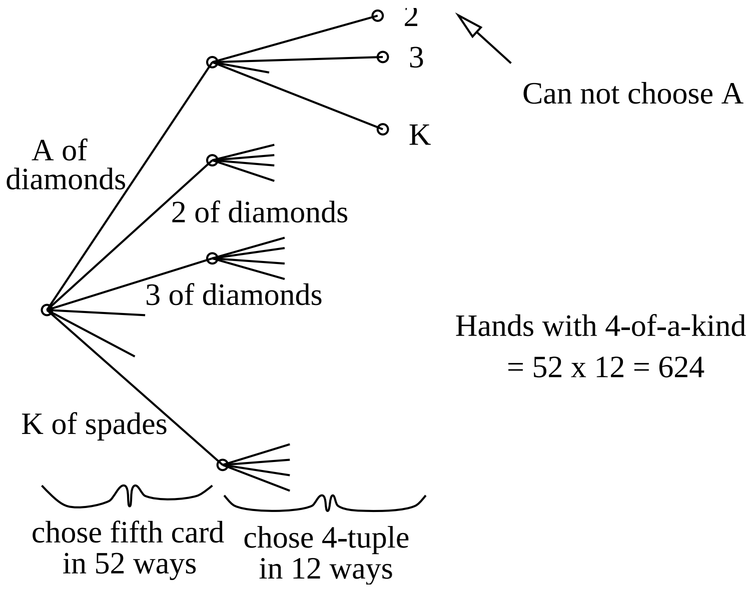


Figure 3: *This tree diagram shows another way to count the number of hands with a four-of-a-kind. There are 52 choices for the "extra" card and then 12 choices for the value appearing in all four suits. This gives* $52 \cdot 12 = 624$ *hands, the same answer as before.*

False proof. There are 13 choices for the value of the cards in the first pair. There are then $\binom{4}{2} = 6$ ways to choose two of the four cards in the deck that have this value. Next, there are 12 choices for the value of the cards in the second pair, and $\binom{4}{2} = 6$ ways to choose two of the four cards with this value. Finally, the fifth card can be any one of the 48 cards remaining in the deck. Altogether there are $13 \cdot 6 \cdot 12 \cdot 6 \cdot 48$ hand with just two pairs. □

There are actually *two* bugs in this argument. The first is that the fifth card *cannot* be any one of the 48 remaining cards; in particular, it cannot have the same value as a card already selected. (Otherwise, there would be three of a kind.) This rules out 4 of the 48 cards, leaving only 44 choices for the fifth card.

The second bug is that every hand has been counted twice! For example, a pair of kings and a pairs of queens is counted once with the kings as the first pair and a second time with the queens as the first pair. The references in the argument to a "first pair" and a "second pair" signal danger; these terms imply an ordering that is not part of the problem. To fix the second bug, we can apply the Division Rule. There is a 2-to-1 mapping from the set of hands we counted to the set of hands with two pairs. Therefore, the set of hands with two pairs is half as large as our initial count:

$$\begin{aligned}\text{hands with two pairs} &= \frac{13 \cdot 6 \cdot 12 \cdot 6 \cdot 44}{2} \\ &= 123,552\end{aligned}$$

Why did this factor of 2 arise in counting pairs, but not in counting full houses? The reason is that pairs are interchangeable, but a pair and a triple are not. For example, a pair of 2's and a pair of

9's is the same as a pair of 9's and a pair of 2's. On the other hand, a pair of 2's and a triple of 9's is different from a pair of 9's and a triple of 2's!

To check our result, we can count the number of hands with two pairs in a completely different way. The number of ways to choose the values for the two pairs is $\binom{13}{2}$. We can choose suits for each pair in $\binom{4}{2} = 6$ ways, and we can choose the remaining card in 44 ways as before. This gives:

$$\binom{13}{2} \cdot 6 \cdot 6 \cdot 44 = 123,552$$

This is the same answer as before. Two pairs turn out to be 33 times more likely than a full house; as one would expect, a full house beats two pairs in poker!

The following optional sections contain similar examples of poker hand counting.

3.5 Three of a kind, two different

[Optional]

There are $\binom{13}{1}$ ways to choose the rank for the three-of-a-kind, and once that is chosen, there are $\binom{4}{3}$ ways to choose the three cards. Once we have the three-of-a-kind, there are $\binom{12}{2}$ ways to choose the two ranks for the other two cards. And then for each of these ranks, there are $\binom{4}{1}$ ways to choose the single card of that rank. The total is:

$$\binom{13}{1}\binom{4}{3}\binom{12}{2}\binom{4}{1}\binom{4}{1} = 54,912.$$

Since this is fewer than the hands with two pairs, this is ranked higher.

3.6 One pair, others all different

[Optional]

There are $\binom{13}{1}$ ways to choose the rank for the pair, and then $\binom{4}{2}$ ways to choose the two cards of that rank. Then there are $\binom{12}{3}$ ways to choose the ranks of the remaining three cards, and for each of these ranks, $\binom{4}{1}$ ways to choose the single card of that rank. The total is:

$$\binom{13}{1}\binom{4}{2}\binom{12}{3}\binom{4}{1}\binom{4}{1}\binom{4}{1} = 1,098,240$$

3.7 Four different ranks

[Optional]

We have analyzed all the actual poker hands based on the numbers of cards of different ranks (4-1, 3-2, 3-1-1, 2-2-1, 2-1-1-1).

How many hands have all 5 cards of different ranks? That's the total number of hands minus the sum of the numbers of hands of these five kinds, or

$$2,598,960 - 1,098,240 - 123,552 - 54,912 - 3,744 - 624 = 1,317,888.$$

What is wrong with the following counting argument?

False counting argument. There are 52 choices for the first card, 48 choices for the second card given the first, 44 choices for the third given the first two, etc. After each card, the number of remaining choices drops by four, since we cannot subsequently pick the card just taken or any of the other three cards with the same value. This gives $52 \cdot 48 \cdot 44 \cdot 40 \cdot 36$ hands with no pair. Is this answer right?

The references to "first card", "second card", ... are a warning signal that we may be inadvertently counting an ordered set. To be sure, we can check how many times a particular hand is counted. Consider the hand $A\heartsuit\ K\spadesuit\ Q\clubsuit\ J\clubsuit\ 10\heartsuit$. We could have obtained this hand by drawing cards in the order listed, but we could also have drawn them in another order. In fact, we counted this hand once for every possible ordering of the five cards, and in fact, every hand is counted $5! = 120$ times! We can correct our initial answer by applying the Division Rule. This gives:

$$\begin{aligned}\text{hands with no pairs} &= \frac{52 \cdot 48 \cdot 44 \cdot 40 \cdot 36}{120} \\ &= 1,317,888\end{aligned}$$

And still another way: There is another way to count the hands with no pairs that avoids this confusion. Since there are no pairs, no two cards have the same value. This means that we could first choose 5 distinct values from the set of 13 possibilities. Then we could choose one of the four suits for each of the five cards. This gives:

$$\begin{aligned}\text{hands with no pairs} &= \binom{13}{5} \cdot 4^5 \\ &= \frac{13 \cdot 12 \cdot 11 \cdot 10 \cdot 9}{5 \cdot 4 \cdot 3 \cdot 2 \cdot 1} \cdot 4^5 \\ &= 1,317,888\end{aligned}$$

This is the same answer as before. Notice that the number of hands with no pairs is more than half of all the hands. This means that in more than half our games of Five-Card Draw, we will not have even one pair!

3.8 Hands with Every Suit

[Optional]

How many hands contain one card of every suit? Such a hand must contain two cards from one suit and one card from each of the other three suits. Again, we can count in two ways.

First count. Pick one card from each suit. This can be done in 13^4 ways. Then pick any one of the remaining 48 cards. This gives a total of $13^4 \cdot 48$ hands. However, we have overcounted by two, since the two cards with the same suit could be picked in either order. Accounting for this, there are $\frac{13^4 \cdot 48}{2} = 685,464$ hands.

Second count First, we can pick the suit that contains two cards in four ways. Then we can pick values for these two cards in $\binom{13}{2}$ ways. Finally, we can pick a card from each of the other three suits in 13 ways. This gives:

$$\begin{aligned}\text{hands with all suits} &= 4\binom{13}{2}\binom{13}{1}^3 \\ &= 4 \cdot \frac{13 \cdot 12}{2} \cdot 13 \cdot 13 \cdot 13 \\ &= 685,464\end{aligned}$$

More than a quarter of all Five-Card Draw hands contain all suits.

3.9 Reminders

To make sure that you get the right answer when counting:

1. Look at a sample 5-card hand (or a sample of whatever you are counting) and make sure that it is *covered once and only once* in your count.

2. Try counting by two different ways and check that you get the same answer.

For example, when we first counted the number of hands with no pairs, we discovered that we had massively overcounted; a sample hand was actually covered 120 times by our count! We corrected this answer by applying the Division Rule, *and* we counted the same thing again in a different way as a double-check.

4 The Magic Trick

There is a Magician and an Assistant. The Assistant goes into the audience with a deck of 52 cards while the Magician looks away. Five audience members are asked to select one card each from the deck. The Assistant then gathers up the five cards and announces four of them to the Magician. The Magician thinks for a time and then correctly names the secret, fifth card! Only elementary combinatorics underlies this trick, but it's not obvious how. (The reader should take a moment to think about this; the Assistant doesn't cheat.)

4.1 How the Trick Works

The Assistant has somehow communicated the secret card to the Magician just by naming the other four cards. In particular, the Assistant has two ways to communicate. First, he can announce the four cards in any order. The number of orderings of four cards is $4! = 24$, so this alone is insufficient to identify which of the remaining 48 cards is the secret one. Second, the Assistant can choose which four of the five cards to reveal. The amount of information that can be conveyed this way is harder to pin down. The Assistant has $\binom{5}{4} = 5$ ways to choose the four cards revealed, but the Magician can not determine which of these five possibilities the Assistant selected, since he does not know the secret card! Nevertheless, these two forms of communication allow the Assistant to covertly reveal the secret card to the Magician.

Here is an overview of how the trick works. The secret card has the same suit as the first card announced. Furthermore, the value of the secret card is offset from the value of the first card announced by between 1 and 6. See Figure 4. This offset is communicated by the order of the last three cards.

Here are the details. The audience selects five cards and there are only 4 suits in the deck. Therefore, by the Pigeonhole Principle, the Assistant can always pick out two cards with the same suit. One of these will become the secret card, and the other will be the first card that he announces.

Here is how he decides which is which. Note that for any two card numbers, one is at most six clockwise steps away from the other in Figure 4. For example, if the card numbers are 2 and Q, then 2 is three clockwise steps away from Q. The Assistant ensures that the value of the secret card is offset between 1 and 6 clockwise steps from the first card he announces.

The offset is communicated by the order that the Assistant announces the last three cards. The Magician and Assistant agree in advance on an order for all 52 cards. For example, they might use:

$$A\clubsuit, 2\clubsuit, \ldots, K\clubsuit, A\diamondsuit, \ldots, K\diamondsuit, A\heartsuit, \ldots, K\heartsuit, A\spadesuit, \ldots, K\spadesuit,$$

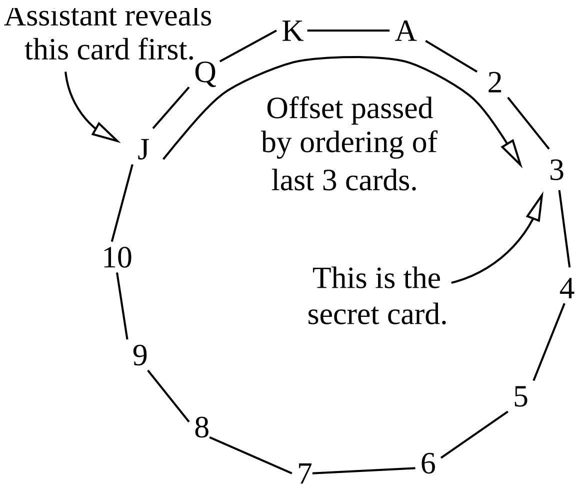

Assistant reveals this card first.
K
A
Q
2
J
3
10
4
9
5
8
7
6
Offset passed by ordering of last 3 cards.
This is the secret card.

Figure 4: *The 13 possible card values can be ordered in a cycle. For any two distinct values, one is offset between 1 and 6 clockwise steps from the other. In this diagram, 3 is offset five steps from J. In the card trick, the value of the secret card is offset between 1 and 6 steps from the first card announced. This offset is communicated by the order of the last three cards named by the Assistant.*

though any other order will do as well—so long as the Magician and his Assistant use the same one `:-)`. With this order, one of the last three cards announced is the smallest (S), one is largest (L), and the other is medium (M). The offset is encoded by the order of these three:

$$\begin{array}{lll} SML = 1 & SLM = 2 & MSL = 3 \\ MLS = 4 & LSM = 5 & LMS = 6 \end{array}$$

For example, suppose that the audience selects $3\heartsuit$ $8\diamondsuit$ $A\spadesuit$ $J\heartsuit$ $6\diamondsuit$. The Assistant picks out two cards with the same suit, say $3\heartsuit$ and $J\heartsuit$. The $3\heartsuit$ is five clockwise steps from $J\heartsuit$ in Figure 4. So the Assistant makes $3\heartsuit$ the secret card, announces $J\heartsuit$ first, and then encodes the number 5 by the order in which he announces the last three cards: $A\spadesuit$ $6\diamondsuit$ $8\diamondsuit = LSM = 5$.

4.2 Same Trick with 4 Cards?

Could the same magic trick work with just 4 cards? That is, if the audience picks four cards, and the Assistant reveals three, then can the Magician determine the fourth card? The answer turns out to be "no". The proof relies on the Pigeonhole Principle.

Theorem 4.1. *The magic trick is not possible with 4 cards.*

Proof. The audience can select any 4-combination of cards; let A be the set of all such 4-combinations. The Assistant can announce any 3-permutation of cards; let B be the set of all such 3-permutations. The formulas for r-combinations and r-permutations give the following sizes for A and B:

$$\begin{aligned}
|A| &= C(n,r)\\
&= \frac{52!}{48!\,4!}\\
&= 270,725\\
|B| &= P(n,r)\\
&= \frac{52!}{49!}\\
&= 132,600
\end{aligned}$$

The Assistant sees a 4-combination of cards selected by the audience and must announce a 3-permutation to the Magician. Let $f : A \mapsto B$ be the function that the Assistant uses in mapping one to the other. Since $|A| > |B|$, the Pigeonhole Principle implies that f maps at least two 4-combinations to the same 3-permutation. That is, there are two different sets of four cards that the audience can pick for which the Assistant says exactly the same thing to the Magician.

For these two sets of four cards, three cards must be the same (since the Assistant announces the same three cards in both cases) and one card must be different (since the two sets are different). For example, these might be the two sets of four cards for which the Assistant says exactly the same thing to the Magician:

$$\begin{gathered}
3\heartsuit\ 8\diamondsuit\ A\spadesuit\ J\heartsuit\\
3\heartsuit\ 8\diamondsuit\ A\spadesuit\ K\diamondsuit
\end{gathered}$$

In this case, the Assistant announces $3\heartsuit\ 8\diamondsuit\ A\spadesuit$ in some order. The magician is now stuck; he can not determine whether the remaining card is $J\heartsuit$ or $K\diamondsuit$. □

There are many variants of the magic trick. For example, if the audience picks 8 cards, then revealing 6 to the Magician is enough to let him determine the other two. This sort of subtle transmission of information is important in the security business where one wants to prevent information from leaking out in undetected ways. This is actually a whole field of study.

4.3 Hall's Theorem applied to the Magic Trick

We know how the "Magic" trick works in which an Assistant reads four cards from a five card hand and the Magician predicts the fifth card. We also know the trick cannot be made to work if the Assistant only shows three cards from a four card hand, because there are fewer sequences of three out of 52 cards than there are 4-card hands out of 52 cards.

So the question is, when can the trick be made to work? For example, what is the largest size deck for which our trick of reading 4 cards from a 5-card hand remains possible?

An elegant result known as Hall's Marriage Theorem allows us to answer the general question of when the trick can be made to work. Namely, the trick is possible using h-card hands chosen from an n-card deck, with the Assistant reading r of the h cards iff

$$P(h,r) \geq C(n-r, h-r). \tag{1}$$

In particular, using $h = 5$-card hands and revealing $4 = r$ cards, we can do the trick with an n-card deck as long as $120 = P(5, 4) \leq C(n - 4, 1) = n - 4$. So we can do the trick with a deck of up to size 124.

For example, we could still do the trick, with room to spare, if we combined a red deck with a blue deck to obtain a 104-card deck. Of course it's not clear whether with the double-size deck there will be a *simple* rule for determining the hidden card as there is with the 52-card deck, but Hall's Theorem guarantees there will be *some* rule.

4.4 Hall's Marriage Theorem

The Magician gets to see a sequence of 4 cards and has to determine the 5th card. He can do this if and only if there is a way to map every 5-card hand into a sequence of 4 cards from that hand so that no two hands map to the same sequence. That is, there needs to be an *injection* from the set of 5-card hands into the 4-card sequences, subject to the constraint that each 5-card hand maps to a 4-card sequence of cards from that hand.

This is an example of a *Marriage Problem*. The traditional way to describe such a problem involves having a set of women and another set of men. Each women has a list of men she is willing to marry and who are also willing to marry her. The Marriage Problem is to find, for each woman, a husband she is willing to marry. Bigamy is not allowed: each husband can have only one wife. When such a matching of wives to husbands is possible, that particular Marriage Problem is *solvable*.

For our card trick, each 5-card hand is a "woman" and each 4-card sequence as a "man." A man (4-card sequence) and woman (5-card hand) are willing to marry iff the cards in the sequence all occur in the hand.

Hall's Theorem gives a simple necessary and sufficient condition for a Marriage Problem to be solvable.

Theorem 4.2 (Hall's Marriage Theorem[1]). *Suppose a group of women each have a list of the men they would be willing to marry. Say that a subset of these women* has enough willing men *if the total number of distinct men on their lists is at least as large as the number of women in the subset. Then there is a way to select distinct husbands for each of the women so that every husband is acceptable to his wife iff every subset of the women has enough willing men.*

Hall's Theorem can also be stated more formally in terms of bipartite graphs.

Definition. A *bipartite graph*, $G = (V_1, V_2, E)$, is a simple graph whose vertices are the disjoint union of V_1 and V_2 and whose edges go between V_1 and V_2, *viz.*,

$$E \subseteq \{\{v_1, v_2\} \mid v_1 \in V_1 \text{ and } v_2 \in V_2\}.$$

A *perfect matching* in G is an injection $f : V_1 \to V_2$ such that $\{v, f(v)\} \in E$ for all $v \in V_1$.

For any set, A, of vertices, define the neighbor set,

$$N(A) ::= \{v \mid \exists a \in A \ \{a, v\} \in E\}.$$

A set $A \subseteq V_1$ is called a *bottleneck* if $|A| > |N(A)|$.

Theorem (Hall). *A bipartite graph has a perfect matching iff it has no bottlenecks.*

A simple condition ensuring that a bipartite graph has no bottleneck is given in the in-class problems for Wednesday, Oct. 30, 2002. This condition is easy to verify for the graph describing the card trick, and it implies that inequality (1) is necessary and sufficient for the trick to be possible.

4.4.1 Other Examples [Optional]

[Optional]

Hall's Marriage Theorem guarantees the possibility of selecting four cards of distinct suits, one each from any four separate piles of 13 cards from a standard deck of 52.

Likewise, there will always be 13 cards of distinct denominations, one from each of 13 piles of four cards. In this case, think of each pile as the set of one to four the distinct card denominations that appear in that pile. Notice that among any k piles of cards at least k distinct denominations appear—by the Generalized Pigeonholing Principle, there are at least that many distinct denominations among any 4k cards. Thus the existence of one card of each denomination spread out across 13 piles is assured by the Theorem.

Pulling in this big theorem is comforting—but not completely satisfactory. Although it assures us that these feats can always be accomplished, it gives absolutely no indication of how to perform them! Trial and error works well enough for the simple four-pile problem, but not with 13 piles. Is there a general procedure for pulling out a selection of distinct denominations from 13 piles of four cards?

Fortunately there is: choose an arbitrary pile containing an Ace, then one containing a two, and so on, doing this for as long as possible until you get stuck. Place these selected piles in a row. Suppose you have selected a total of ten piles, containing an Ace, and two through ten, respectively. This scenario leaves three untouched piles. If any of the 12 cards in those three piles is a Jack, Queen, or King then you are not really stuck; you can continue a little further (but perhaps not in the usual sequential order). If you are truly stuck, select any card in, say, the 11th pile, and go to one of the ten piles that corresponds to the number of that card. Thus if you select a three you go to the third pile. Turn a card in that third pile over (to make it conspicuous) and move to the pile that corresponds to the number of that turned card. Keep doing this to create a chain of piles and turned cards, until you eventually hit upon a pile that contains a card not in the initial list; in our case, until we come upon a Jack, Queen, or King. (Note: One can prove that you will not fall into a closed loop of choices.) Now shift all the piles one place back along the chain of piles to obtain a configuration that allows you to add one more pile to the initial list of ten. Repeat this process until you solve the puzzle completely.

4.4.2 Proof of Hall's Theorem by Induction [Optional]

[Optional]

This method of chain shifting is known in the literature as the technique of *augmenting paths* and is the basis for *efficient* ways to find perfect matchings and to solve related "network flow" problems. This is described in many combinatorics texts[2], but we shall not develop this method here.

Instead, we'll go back to the man-woman marriage terminology and give a simple proof by:

Proof. Strong induction on the number, n, of women.

Base case ($n = 1$). If there is just a single woman with at least one name on her list, then she can simply marry the first man on her list.

Induction step ($n > 1$).

Case I. Suppose these women's lists satisfy the stronger condition that among any r lists, $1 \leq r < n$, at least $r + 1$ distinct names are mentioned. Select one woman. She has at least two names on her list ($r = 1$ case). Have her marry one of these men (call him "Poindexter"). This leaves $n - 1$ women to marry.

[2] For example, Ian Anderson presents a procedural proof using augmenting paths in his book *A First Course in Combinatorial Mathematics*, Oxford Applied Mathematics and Computer Science Series, Clarendon Press, Oxford, 1974.

The lists possessed by these $n - 1$ women have the property that among any r of them $(1 \leq r < n)$ at least $r + 1$ distinct names are mentioned. One of these names could be Poindexter's who is no longer available for marriage. But we can still say that among any r lists, at least r distinct names of available men are mentioned. This is all we need to invoke the induction hypothesis for these remaining $n-1$ women. Thus we have a means to marry all n women in this scenario.

Case II. Suppose this stronger condition does not hold. Thus there is subgroup of r_0 women $(1 \leq r_0 < n)$ such that among their lists precisely r_0 distinct names are mentioned. By the strong induction hypothesis for $r_0 < n$, we can marry these women. This leaves $m = n - r_0 < n$ women to consider.

Is it true that among these m women's lists any r of them $(1 \leq r \leq m)$ mention at least r distinct names of available men? The answer is yes! If not, say some r of these women mention fewer than r available men. Then these r women plus the r_0 women above have mentioned fewer in $r_0 + r$ men in total. This contradicts the property satisfied by these lists.

Thus we can invoke the strong induction hypothesis for the remaining m women and successfully have them marry as well. This completes the proof by induction. □

Problem 1. The induction proof can read as a recursive procedure to construct a marriages for n woman in terms of constructing marriages for smaller sets of women. Why isn't this procedure very efficient?

Problem 2. A deck of cards is shuffled and dealt into 26 piles of two cards. Is it possible to select a black Ace from one pile, a Red ace from another, a black two from a third, a red two from a fourth, and so on all the way down to a black King and a red King from the two remaining piles?

5 Properties of Binomial Coefficients

There are infinitely many interesting identities involving binomial coefficients. We already seen a simple example in Theorem 2.3. These identities are important because of how often binomial coefficients arise in analysis of algorithms (6.046) and probability (next week). You can't learn them all, but you can learn the basic ones and look others up/derive them as needed.

Let's begin with a formula for the ratio between successive binomial coefficients—same n, successive r:

Theorem 5.1. *Suppose* $1 \leq r \leq n$. *Then*

$$\binom{n}{r} \Big/ \binom{n}{r-1} = \frac{n-r+1}{r}.$$

Algebraic proof.

$$\binom{n}{r} \Big/ \binom{n}{r-1} = \frac{n!}{r!\,(n-r)!} \times \frac{(r-1)!\,(n-(r-1))!}{n!} = \frac{n-r+1}{r}.$$

□

Combinatorial proof. Consider choosing a committee of size r and a leader, from n people. One way is to first pick the $r-1$ commons and then their leader; this can be done in $\binom{n}{r-1}(n-(r-1))$ ways. Another way is to pick the r committee members first and then pick a leader from among them; this can be done in $\binom{n}{r}r$ ways. Thus,

$$\binom{n}{r-1}(n-r+1) = \binom{n}{r}r.$$

□

This ratio is greater than 1 if $r < \frac{n+1}{2}$ and less than 1 if $r > \frac{n+1}{2}$. This says that the successive coefficients increase until r reaches $\frac{n+1}{2}$ and then decrease, i.e., they are *unimodal* like the curve in Figure 5.

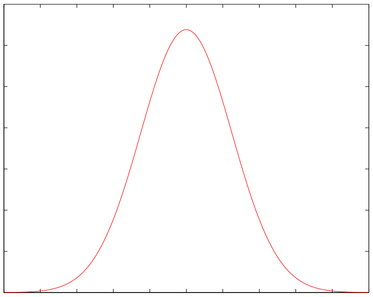


Figure 5: Bell Curve

In fact, the binomial coefficients form a real *bell curve*. These arise on examinations in a natural way. Suppose that a test has n questions. Then the bell curve describes, for every r, the number of ways the student can get exactly r of the questions right, namely $\binom{n}{r}$. If we suppose that each student taking the test is doing random guessing, so is equally likely to get each of the n questions right or wrong, then these coefficients turn out to be proportional to the number of students that get those numbers of questions right. (We'll see this when we do probability.)

Another identity:

Theorem 5.2. *Suppose* $1 \leq r \leq n$. *Then*

$$\binom{n}{r} = \frac{n}{r}\binom{n-1}{r-1}.$$

Algebraic proof.

$$\begin{aligned}\binom{n}{r} &= \frac{n!}{r!(n-r)!} = \frac{n}{r}\frac{(n-1)!}{(r-1)!(n-r)!} \\ &= \frac{n}{r}\frac{(n-1)!}{(r-1)!\,((n-1)-(r-1))!} = \frac{n}{r}\binom{n-1}{r-1}.\end{aligned}$$

□

Combinatorial proof. Consider choosing a committee of size r and a leader, from n people. One way is to first pick the leader and then his $r-1$ subjects; this can be done in $n\binom{n-1}{r-1}$ ways. Another way is to pick the r committee members first and then pick a leader from among them; this can be done in $\binom{n}{r}r$ ways. Thus,

$$n\binom{n-1}{r-1} = \binom{n}{r}r.$$

□

Now an important theorem due to Pascal:

Theorem 5.3 (Pascal). *Suppose* $1 \leq r \leq n-1$. *Then*

$$\binom{n}{r} = \binom{n-1}{r} + \binom{n-1}{r-1}.$$

This is probably our most important identity, since it gives a kind of recurrence for binomial coefficients. **Memorize it!**

Algebraic proof.

$$\begin{aligned}
\binom{n-1}{r} + \binom{n-1}{r-1} &= \frac{(n-1)!}{r!(n-1-r)!} + \frac{(n-1)!}{(r-1)!(n-r)!} \\
&= (n-r)\frac{(n-1)!}{r!(n-r)!} + r\frac{(n-1)!}{r!(n-r)!} \\
&= n\frac{(n-1)!}{r!(n-r)!} \\
&= \frac{n!}{r!(n-r)!} \\
&= \binom{n}{r}
\end{aligned}$$

□

Combinatorial proof. We use case analysis (a tree diagram) and the sum rule. Let $S ::= \{1, \ldots, n\}$. Let A be the set of r-element subsets of S. Let B be the set of r-element subsets of S that contain n. Let C be the set of r-element subsets of S that don't contain n.

Then $A = B \cup C$, and B and C are disjoint. So $|A| = |B| + |C|$, by the Sum Rule. But now we can get expressions for $|A|$, $|B|$ and $|C|$ as numbers of combinations:

- $$|A| = \binom{n}{r}.$$

- $$|B| = \binom{n-1}{r-1}.$$

  This is because, in addition to n, another $r-1$ elements must be chosen from $\{1, \ldots, n-1\}$.

- $$|C| = \binom{n-1}{r}.$$

  This is because r elements must be chosen from $\{1, \ldots, n-1\}$.

So (by the Sum Rule)

$$\binom{n}{r} = |A| = |B| + |C| = \binom{n-1}{r-1} + \binom{n-1}{r}.$$

□

Pascal's theorem has a nice pictorial representation: The row represents n, starting with 0 in the top row. Successive elements in the row represent r, starting with 0 at the left.

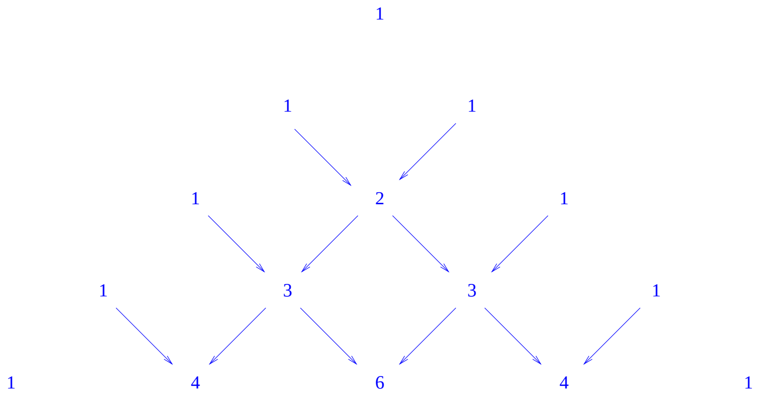


Figure 6: Pascal's triangle.

(Notice that it's just the double-induction matrix "reshaped".)

This triangle has lots of nice properties. Experiment with it. For example, what happens if we add the coefficients in one row?

For the following two theorems we provide their combinatorial proofs, only. The corresponding algebraic proofs are easy inductive arguments.

Theorem 5.4. *Suppose n is any natural number. Then*

$$\sum_{r=0}^{n} \binom{n}{r} = 2^n.$$

Combinatorial proof. Again, we use case analysis and the sum rule for disjoint unions. There are 2^n different subsets of a set of n elements. Decompose this collection based on the sizes of the subsets. That is, let A_r be the collection of subsets of size r. Then the set of all subsets is the disjoint union $\cup_r A_r$. There are $\binom{n}{r}$ subsets of size r, for $r = 0, 1, \ldots, n$. Hence the theorem follows. □

[Optional]

Theorem 5.5 (Vandermonde). *Suppose* $0 \leq r \leq m, n$. *Then*

$$\binom{m+n}{r} = \sum_{k=0}^{r} \binom{m}{r-k}\binom{n}{k}$$

Combinatorial proof. Again, we use case analysis with the sum and product rules. Suppose there are m red balls and n blue balls, all distinct. There are $\binom{m+n}{r}$ ways to choose r balls from the two sets combined. That's the LHS. Now decompose this collection of choices based on how many of the chosen balls are blue. For any k, $0 \leq k \leq r$, there are $\binom{m}{r-k}$ ways to choose $r-k$ red balls, and $\binom{n}{k}$ ways to choose k blue balls. By the product rule, that makes $\binom{m}{r-k}\binom{n}{k}$ ways to choose r balls such that k of them are blue. Adding up these numbers for all k gives the RHS. □

5.1 The Binomial Theorem

An important theorem gives the coefficients in the expansion of powers of binomial expressions. It turns out they are binomial coefficients—thus the name!

Theorem 5.6. *Suppose* $n \in \mathbb{N}$. *Then*

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}.$$

Example 5.7. $(x+y)^4 = 1x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + 1y^4$.

Algebraic proof. By induction, using Pascal's identity. □

Combinatorial proof. Consider the product

$$(x+y)^n = (x+y)(x+y)(x+y)(x+y)\cdots$$

This product has n factors. To expand this product, we generate individual terms of the sums by selecting (in all possible ways) one of the two variables inside each factor to get an n-variable product. If we select k "x" variables and $n-k$ "y" variables, we get a term of the form $x^k y^{n-k}$. Then we gather all those terms together. So how many ways do we get such a term? Each $x^k y^{n-k}$ term selects x from k of the factors and y from the other $n-k$. So there are $\binom{n}{k}$ ways of choosing which factors have x. □

The binomial theorem can be used to give slick proofs of some binomial identities, for example:

Theorem 5.8.

$$\sum_{k=0}^{n} \binom{n}{k} = 2^n.$$

Proof. Plug $x = 1$, $y = 1$ into binomial theorem. □

Theorem 5.9.

$$\sum_{k=0}^{n} (-1)^k \binom{n}{k} = 0.$$

Proof. Plug $x = -1$, $y = 1$ into binomial theorem. □

The combinatorial interpretation is as follows: the number of ways of selecting an even number of elements from a set of n equals the number of ways of selecting an odd number. This works for both even and odd n.

Example 5.10. Consider $n = 5$. We can choose an odd-size set

$$\binom{5}{1} + \binom{5}{3} + \binom{5}{5} = 5 + 10 + 1 = 16$$

ways. We can choose an even-size set

$$\binom{5}{0} + \binom{5}{2} + \binom{5}{4} = 1 + 10 + 5 = 16$$

ways.

Example 5.11. $n = 6$: There are $6 + 20 + 6 = 32$ odd sets. There are $1 + 15 + 15 + 1 = 32$ even sets.

For odd n, this is intuitive (choosing an even set leaves an odd set). For even n, it may be somewhat surprising—there is no obvious bijection between the even and odd sets. Theorem 5.9 can also be proven using inclusion/exclusion in a tricky way.

6 Principle of Inclusion and Exclusion

Now we return to the Principle of Inclusion-Exclusion from last week. Our binomial coefficient identities give us a way to prove this theorem, via our alternating sum identity (proved by binomial theorem). Recall the Inclusion-Exclusion theorem:

Theorem 6.1. $|\bigcup_i A_i| = \sum_i |A_i| - \sum_{i<j} |A_i \cap A_j| + \sum_{i<j<k} |A_i \cap A_j \cap A_k| - \cdots$

The theorem can be proved by induction on the number of sets, but this approach is messy . Instead we will give a combinatorial proof using binomial coefficients.

Proof. Each term in the summation counts certain elements in $\bigcup_i A_i$. We prove that every element of this union is counted exactly once. So, consider any particular element a, and suppose it is in exactly r of the sets A_i. We see how many times a is counted by each term.

In the first term, a is counted $r = \binom{r}{1}$ times, once for each of the sets that contains it. In the second term, a is counted $\binom{r}{2}$ times, once for each of the pairs of sets that contain it. In the third term, a is counted $\binom{r}{3}$ times, ..., and in the rth term, a is counted $\binom{r}{r} = 1$ times. The terms alternate signs, so the total number of times a is counted is:

$$\binom{r}{1} - \binom{r}{2} + \binom{r}{3} - \cdots - (-1)^r \binom{r}{r}.$$

But Theorem 5.9 implies that

$$\sum_{k=0}^{r} (-1)^k \binom{n}{k} = 0.$$

This implies that the sum above is equal to $\binom{r}{0} = 1$. That is, a is counted exactly once, as needed. □

7 *r*-Permutations with Repetition

We now turn to permutations and combinations where elements are allowed to repeat. An r-*permutation with repetition* of a set S is the number of ways to choose r elements from S with repetition allowed and where order matters. For example, there are nine 2-permutations with repetition of the set $S = \{A, B, C\}$, which are listed below.

$$\begin{array}{ccc} (A, A) & (A, B) & (A, C) \\ (B, A) & (B, B) & (B, C) \\ (C, A) & (C, B) & (C, C) \end{array}$$

7.1 Counting *r*-Permutations with Repetition

Fortunately, r-permutations with repetition are easy to count. It is the same as using the product rule to count the number of strings of length r from an alphabet with n letters.

Theorem 7.1. *The number of r-permutations with repetition of an n-element set is n^r.*

For example, the theorem says that the number of 2-permutations with repetition of the 3-element set $S = \{A, B, C\}$ is $3^2 = 9$, which checks with our previous answer.

Proof. Let S be a set with n elements. The r-permutations with repetition of S are precisely the elements of:

$$\underbrace{S \times S \times \cdots \times S}_{r \text{ terms}}$$

By the Product Rule, this set has cardinality $|S|^r = n^r$. □

7.2 Permutations with Limited Repetition

We might want to count the number of r-permutations where each element can be repeated a limited number of times. In general, this leads to some hairy analysis and no closed-form answer. Therefore, we will consider only a special case, the number of r-permutations where each element is repeated a precisely specified number of times.

For example, in how many ways can we arrange the letters in the word $PEPPER$? This is equal to the number of 6-permutations of the set $\{P, E, R\}$ where P is repeated 3 times, E is repeated 2 times, and R is repeated 1 time. (Since the total number of repetitions defines r, we will use the term "permutations" in place of "r-permutations" for the remainder of the section.)

Initially, suppose that we make all the letters distinct by adding subscripts. That is, we want the number of ways to arrange the letters $P_1E_1P_2P_3E_2R$. In this case, there are $6! = 720$ arrangements because there are six choices for the first letter, five choices for the second letter, etc.

Next, suppose that we erase the subscripts on the E's. This maps each arrangement of the letters $P_1E_1P_2P_3E_2R$ to an arrangement of the letters $P_1EP_2P_3ER$. Since E_1 and E_2 could be ordered in 2! ways before the erasure, the mapping is 2!-to-1. For example, we have:

$$\begin{array}{ccc} P_1E_1P_2P_3E_2R & \rightarrow & P_1EP_2P_3ER \\ & \text{and} & \\ P_1E_2P_2P_3E_1R & \rightarrow & P_1EP_2P_3ER \end{array}$$

Therefore, by the Division Rule there are $6!/2! = 360$ arrangement of the letters in $P_1EP_2P_3ER$.

Finally, suppose that we erase the subscripts on the P's. This maps each arrangement of the letters $P_1EP_2P_3ER$ to an arrangement of the letters $PEPPER$. Since P_1, P_2, and P_3 could be ordered in 3! ways before the erasure, this mapping is 3!-to-1. Therefore, by the Division Rule, the number of arrangements of the letters $PEPPER$ is

$$\frac{6!}{2!\,3!} = 60.$$

We can prove a general theorem using the same argument as in the $PEPPER$ problem.

Theorem 7.2. *Let A be the set $\{a_1, a_2, \ldots, a_n\}$, and let $r_1, r_2, \ldots, r_n$ be non-negative integers. The number of permutations of the set A where each element a_i is repeated exactly r_i times is:*

$$\frac{(r_1 + r_2 + \cdots + r_n)!}{r_1!\, r_2!\, \ldots\, r_n!}$$

For example, the theorem says that the number of permutations of $\{P, E, R\}$ where P is repeated 3 times, E is repeated 2 times, and R is repeated 1 time is

$$\frac{(3 + 2 + 1)!}{3!\, 2!\, 1!} = 60.$$

This is the answer we found before.

Proof sketch. We initially make r_i distinct copies of each element a_i. Then the number of permutations where each element is repeated exactly once is $(r_1 + r_2 + \cdots + r_n)!$. We then "erase the subscripts" on the distinct copies of element a_1. This defines an r_1!-to-1 mapping from old permutations to permutations where a_1 is repeated r_1 times and all other elements are repeated once. Therefore the number of new permutations is

$$\frac{(r_1 + r_2 + \cdots + r_n)!}{r_1!}.$$

Continuing this way with $a_2, a_3, \ldots$, we find that the number of permutations where each element a_i is repeated exactly r_i times is

$$\frac{(r_1 + r_2 + \cdots + r_n)!}{r_1!\, r_2!\, \ldots\, r_n!}$$

. □

7.3 Multinomial Coefficients

What is the number of permutations of the set $A = \{a_1, a_2\}$ where a_1 is repeated r_1 times and a_2 is repeated r_2 times? According to Theorem 7.2, the answer is:

$$\frac{(r_1 + r_2)!}{r_1!\, r_2!} = \binom{r_1 + r_2}{r_1}$$

We can restate the question as follows. How many strings contain r_1 copies of the symbol a_1 and r_2 copies of the symbol a_2? This is equal to the number of ways to choose r_1 distinct positions for the a_1's from the set of all $r_1 + r_2$ positions, which is $\binom{r_1+r_2}{r_1}$. This is the same as the previous answer.

By shifting from "permutations" to "strings" in this way, we can give an alternative proof of Theorem 7.2.

Alternative proof of Theorem 7.2. There is a bijection between permutations of the set $A = \{a_1, a_2, \ldots, a_n\}$ where each element a_i appears exactly r_i times and strings where each symbol a_i appears exactly r_i times. Therefore, we can count permutations by counting strings as follows.

The number of ways to choose r_1 distinct positions for the a_1's from the set of all $r_1 + r_2 + \cdots + r_n$ positions is $\binom{r_1+r_2+\cdots+r_n}{r_1}$. Then the number of ways to choose r_2 distinct positions for the a_2's from the set of all $r_2 + \cdots + r_n$ remaining positions is $\binom{r_2+\cdots+r_n}{r_2}$, and so forth. The total number of strings is therefore:

$$\begin{aligned}
&\binom{r_1 + r_2 + \cdots + r_n}{r_1} \cdot \binom{r_2 + \cdots + r_n}{r_2} \cdot \binom{r_3 + \cdots + r_n}{r_3} \cdots \cdot \binom{r_n}{r_n} \\
=\ & \frac{(r_1 + r_2 + \cdots + r_n)!}{r_1!\,(r_2 + \cdots + r_n)!} \cdot \frac{(r_2 + \cdots + r_n)!}{r_2!\,(r_3 + \cdots + r_n)!} \cdot \frac{(r_3 + \cdots + r_n)!}{r_3!\,(r_4 + \cdots + r_n)!} \cdots \cdot \frac{r_n!}{r_n!\,0!} \\
=\ & \frac{(r_1 + r_2 + \cdots + r_n)!}{r_1!\, r_2!\, \ldots\, r_n!}
\end{aligned}$$

The first equality uses the definition of binomial coefficients, and the second follows by cancelling terms. □

The quantity

$$\frac{(r_1 + r_2 + \cdots + r_n)!}{r_1!\, r_2!\, \ldots\, r_n!}$$

is called a *multinomial coefficient* and is denoted

$$\binom{r_1 + r_2 + \cdots + r_n}{r_1, r_2, \ldots, r_n}.$$

Given a set with $r_1 + r_2 + \cdots + r_n$ elements, the multinomial coefficient $\binom{r_1+r_2+\cdots+r_n}{r_1,r_2,\ldots,r_n}$ represents the number of ways to choose r_1 elements, then r_2 of the remaining elements, and so forth.

Multinomial coefficients also arise in the Multinomial Theorem, a generalization of the Binomial Theorem. The result is stated below, but not proved.

Theorem 7.3 (Multinomial Theorem).

$$(x_1 + x_2 + \cdots + x_n)^r = \sum_{r_1+r_2+\cdots+r_n=r} \binom{r}{r_1, r_2, \ldots, r_n} x^{r_1} x^{r_2} \ldots x^{r_n}$$

8 Combinations with Repetition

Now we move from permutations with repetition to combinations with repetition. Let S be the set $\{A, B, C\}$. As we saw in the previous lecture, this set has three 2-combinations. That is, there are three ways to choose two distinct elements of S where order does not matter. The three 2-combinations of S are shown below.

$$\{A, B\} \quad \{A, C\} \quad \{B, C\}$$

Suppose that we are not required to choose distinct elements of S, but rather can choose the same element repeatedly. The resulting sets are called the *r-combinations with repetition* of the set S. Listed below are the six 2-combinations with repetition of S.

$$\{A, B\} \quad \{A, C\} \quad \{B, C\} \quad \{A, A\} \quad \{B, B\} \quad \{C, C\}$$

Strictly speaking, these are multisets (bags), not sets, since an element may appear multiple times.

8.1 Counting *r*-Combinations with Repetition

The following theorem gives a nice formula for the number of r-combinations with repetition of an n-element set.

Theorem 8.1. *The number of r-combinations with repetition of an n-element set is*

$$\binom{n + r - 1}{r}.$$

In the example above, we found six ways to choose two elements from the set $S = \{A, B, C\}$ with repetition allowed. Sure enough, the theorem says that the number of 2-combinations of a 3-element set is $\binom{3+2-1}{2} = 6$.

For comparison, recall that the number of ordinary r-combinations of an n-element set is $\binom{n}{r}$. Every ordinary r-combination is also a valid r-combination with repetition. So, as one would expect, the number of r-combinations with repetition is greater if $r > 1$.

The proof of this theorem uses an important trick called "stars and bars".

Proof. Let S be a set with n elements that are ordered in some way. We will establish a bijection between r-combinations with repetition of the set S and strings of stars and bars.

Let R be a particular r-combination with repetition of S. Write down $n-1$ bars. These $n-1$ bars divide the line into n regions.

$$\underbrace{\quad | \quad | \quad | \quad \cdots \quad | \quad}_{n-1 \text{ bars define } n \text{ regions}}$$

Put one star in the i-th region for each time that the i-th element of S appears in R. This procedure maps each r-combination with repetition to a string with r stars and $n-1$ bars.

(For example, let S be the set $\{A, B, C, D, E\}$, with elements ordered alphabetically. Let R be the 7-combination with repetition $\{A, B, B, B, D, E, E\}$. The stars-and-bars string corresponding to R is shown below.

$$\underbrace{\;\star\;}_{A} \mid \underbrace{\star\ \star\ \star}_{B,B,B} \mid \mid \underbrace{\;\star\;}_{D} \mid \underbrace{\star\,\star}_{E,E}$$

The two bars with no stars between indicate that element C never appears in R.)

This mapping is a bijection because it has an inverse. That is, given any stars-and-bars string, we can construct the corresponding r-combination with repetition. The number of stars in the first region determines the number of times that the first element of S appears in the r-combination, the stars in the second region determine the number of times that the second element appears, etc.

Since the mapping is a bijection, the number of r-combinations with repetition of an n-element set is equal to the number of strings containing $n-1$ bars and r stars. The number of such strings is equal to the number of ways to choose r distinct positions for the stars in a string of $n+r-1$ stars and bars. This is the number of ordinary r-combinations of a set with $n+r-1$ elements, which is $\binom{n+r-1}{r}$. □

8.2 Triple-Scoop Ice Cream Cones

Baskin-Robbins is an ice cream store that has 31 different flavors. How many different triple-scoop ice cream cones are possible at Baskin-Robbins? Two ice cream cones are considered the same if one can be obtained from the other by reordering the scoops. Of course, we are permitted to have two or even three scoops of the same flavor.

Of course, the best solution to this problem is to go to the Baskin-Robbins in Harvard Square and explicitly construct all possible combinations. This is called the *consumption method*. However, there is also a purely mathematical approach.

The number of triple-scoop ice cream cones is precisely the number of 3-combinations with repetition of the set of 31 flavors. Therefore, we can count the number of ice creams cones with the formula in Theorem 8.1 where $n = 31$ and $r = 3$. This gives:

$$\binom{31+3-1}{3} = \binom{33}{3} = \frac{33 \cdot 32 \cdot 31}{3 \cdot 2 \cdot 1} = 5456$$

On Thursdays the Harvard Square Baskin-Robbins is run by an irritating woman who refuses to serve a cone with two or three scoops of the same flavor. How many different triple-scoop ice cream cones are possible on Thursday?

Now we must select an ice cream cone by choosing 3 *distinct* flavors from the complete set of 31. Therefore, the number of different cones is the number of ordinary 3-combinations of a 31-element set. This is:

$$\binom{31}{3} = \frac{31 \cdot 30 \cdot 29}{3 \cdot 2 \cdot 1} = 4495$$

As we would expect, the number of combinations with repetition is greater than number of combinations without repetition. Permitting scoops of the same flavor gives an extra $5456 - 4495 = 961$ options.

8.3 Dozens of Dunkin Donuts

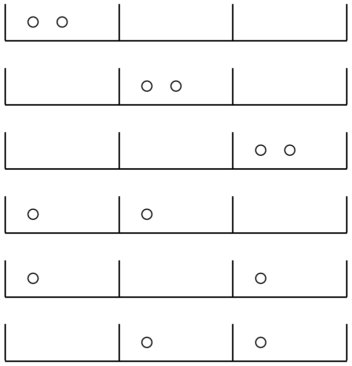
Suppose we next stagger down to the Dunkin Donuts in Central Square. We want a box of a dozen doughnuts and there are 21 varieties available. How many options do we have?

We select our box of doughnuts by pointing out 12 varieties from the complete set of 21 where repetition is allowed. Therefore, the number of options is the number of 12-combinations with repetition of a 21 element set. Applying Theorem 8.1 with $n = 21$ and $r = 12$, we find that number of different boxes of doughnuts is:

$$\binom{21+12-1}{12} = \binom{32}{12} = 225,792,840$$

8.4 Balls and Bins

Suppose that we have r identical balls and n distinct bins. In how many ways can we arrange the balls in the bins? There are six possibilities for the case of two balls and three bins, as shown below.



Claim 8.2. *There are $\binom{n+r-1}{r}$ ways to arrange r identical balls in n distinct bins.*

Proof. Each arrangement of the r balls corresponds to an r-combination with repetition of the set of n bins. Specifically, if the i-th bin contains a_i balls, then the corresponding r-combination with repetition contains the i-th bin a_i times. Therefore, the number of arrangement of balls is equal to the number of r-combinations of an n-element set, which is $\binom{n+r-1}{r}$ by Theorem 8.1. □

There is another way to prove the claim that uses a cute bijection between balls-and-bins arrangements and stars-and-bars strings. The six arrangements in the preceding example are redrawn below with some lines erased and the balls replaced by stars.

$$\begin{array}{ccccc}
\star\ \star & | & & | & \\
 & | & \star\ \star & | & \\
 & | & & | & \star\ \star \\
\star & | & \star & | & \\
\star & | & & | & \star \\
 & | & \star & | & \star
\end{array}$$

Now each balls-and-bin diagram has become a stars-and-bars string. The number of ways to place r balls in n bins is therefore equal to the number of strings with r stars and $n-1$ bars, which is

$$\binom{n+r-1}{r}.$$

8.5 *r*-Combinations with at Least One of Each Item

Suppose that Kaybee Toys carries balls in three delightful colors: red, blue, and green.[3] In how many ways can we choose five balls so that we get at least one ball of each color?

We might as well start by choosing one ball of each color. Then we can choose the last two balls however we like. Remember that we are counting combinations, so the order in which we choose the balls does not matter.

Under this interpretation, the number of options is equal to the number of ways to choose two balls from the set of three colors with repetition allowed. Therefore, by Theorem 8.1 there are $\binom{3+2-1}{2} = 6$ possibilities. Here they are:

$$\begin{array}{ccc}
\{R,G,B,\ R,R\} & \{R,G,B,\ R,G\} & \{R,G,B,\ R,B\} \\
\{R,G,B,\ G,G\} & \{R,G,B,\ G,B\} & \{R,G,B,\ B,B\}
\end{array}$$

This argument generalizes to give the theorem below.

[3] I bet you're wondering if the 6.042 staff has discovered a lucrative scam involving product promotion in lecture notes. No comment.

Theorem 8.3. *The number of r-combinations with repetition of an n-element set that contain every element in the set at least once is:*

$$\binom{r-1}{n-1}$$

For example, in the colored balls problem, we have $r = 5$ and $n = 3$. The theorem says that there are $\binom{5-1}{3-1} = 6$ possibilities, which is consistent with our previous answer.

Proof. Every such r-combination consists of the entire n-element set together with an $(r - n)$-combination with repetition of the n-element set. By Theorem 8.1 the number of such combinations is:

$$\binom{n + (r-n) - 1}{r-n} = \binom{r-1}{r-n} = \binom{r-1}{n-1}$$

The last equality uses the identity $\binom{m}{k} = \binom{m}{m-k}$. □

How many n-combinations with repetition of an n-element set are there that contain every element in the set? In this case, Theorem 8.3 gives $\binom{n-1}{n-1} = 1$. This makes sense; the only such combination is the set S itself!

In how many ways can we arrange r identical balls in n distinct bins so that no bin is empty? If we first put one ball in each bin, then we can arrange the remaining $r - n$ balls in $\binom{n+(r-n)-1}{r-n} = \binom{n-1}{r-1}$ ways.

8.6 *r*-Combinations with Limited Repetition

Suppose that we want to count r-combinations where an element can be repeated only a limited number of times. For example, in how many ways can we arrange 10 identical balls in 4 distinct bins such no bin gets more than 7 balls? A good way to solve such problems is first to count the number of arrangements without a limit on repetition and then to subtract off the illegal arrangements.

The number of ways to arrange 10 identical balls in 4 distinct bins without a limit on repetition is $\binom{4+10-1}{10} = 286$.

Now we must count the illegal arrangements; that is, arrangements in which some bin contains 8 or more balls. Since there are only 10 balls in total, at most one bin can be overloaded with 8 or more balls. We can count the number of ways to overload the first bin by putting 8 balls into the first bin and then observing that the last two balls can be placed in $\binom{4+2-1}{2} = 10$ ways. Since any one of the four bins could be overloaded, the total number of illegal arrangement is $4 \cdot 10 = 40$.

Overall, there are $286 - 40 = 246$ ways to arrange 10 balls in 4 bins so that no bin gets more than 7 balls.

Sometimes when there are limits on repetition, the number of illegal arrangements can itself be difficult to compute. In such cases, a messy inclusion-exclusion calculation may be necessary.

8.7 Data Compression [Optional]

[Optional]

Stars and bars can help solve a data compression problem. How many bits are needed to specify a multiset of n arbitrary integers in the range $[0, 2n]$? For example, how much disk space do we need to store one million numbers in the range zero to two million?

A Simple Scheme

One simple scheme is store each number in binary. With this approach, storing each number requires $\lceil \log_2(2n+1) \rceil$ bits. Storing all n numbers requires $n \lceil \log_2(2n+1) \rceil$ bits. In the case of a million numbers, we need 21 bits to store each number and therefore 21 million bits to store all of the numbers.

A Better Approach Using Stars and Bars

There is a more clever scheme that uses stars and bars. We can regard the multiset of numbers that we want to store as an n-combination with repetition of the $(2n+1)$-element set $[0, 2n]$. The proof of Theorem 8.1 shows that such n-combinations with repetition correspond to strings of n stars and $2n$ bars. If we represent a star with a 0 and a bar with a 1, then we can store such a stars-and-bars string using $n + 2n = 3n$ bits.

To make the scheme clear, suppose that we are storing a multiset of $n = 5$ numbers in the range $[0, 10]$. In particular, suppose that the numbers are $\{2, 4, 4, 6, 7\}$. We can represent this multiset as a stars-and-bars string and then with $3n = 15$ bits as follows.

$$\begin{array}{rcl} \{2,4,4,6,7\} & \rightarrow & \mid \mid \star \mid \mid \star\star \mid \mid \star \mid \star \mid \mid \mid \\ & \rightarrow & 1\,1\,0\,1\,1\,0\,0\,1\,1\,0\,1\,0\,1\,1\,1 \end{array}$$

As another example, we could store a million numbers in the range zero to two million using only 3 million bits, a factor of 7 improvement over the simple scheme. It is rather surprising that we need only 3 bits per number on average, no matter how large n becomes!

An Optimal Scheme

The stars-and-bars approach is simple and efficient, but not quite optimal.

In any scheme, we must associate every multiset of n numbers with a distinct binary string. From the stars-and-bars interpretation, we know that there are $\binom{3n}{n}$ possible sets of n numbers. Therefore, we must be prepared to store any one of $\binom{3n}{n}$ distinct binary strings. Using k bits, we can store at most 2^k different binary strings. This means that we need at least k bits where $2^k \geq \binom{3n}{n}$. This inequality is satisfied provided $k \geq \left\lceil \log_2 \binom{3n}{n} \right\rceil$. Therefore, as a lower bound, we need at least $\left\lceil \log_2 \binom{3n}{n} \right\rceil$ bits.

How far from optimal is the stars-and-bars scheme? To answer this question, we must rewrite the lower bound in a more familiar form. To this end, we can use Stirling's Formula to approximate $\binom{3n}{n}$ as follows.

$$\begin{aligned} \binom{3n}{n} &= \frac{(3n)!}{(2n)!\, n!} \\ &\sim \frac{\sqrt{2\pi(3n)}\left(\frac{3n}{e}\right)^{3n}}{\sqrt{2\pi(2n)}\left(\frac{2n}{e}\right)^{2n}\sqrt{2\pi n}\left(\frac{n}{e}\right)^{n}} \\ &= \sqrt{\frac{3}{4\pi n}}\left(\frac{3^{3n}}{2^{2n}}\right) \\ &= \sqrt{\frac{3}{4\pi n}}\left(\frac{27}{4}\right)^{n} \end{aligned}$$

With this formula, we can approximate the lower bound.

$$\begin{aligned}\left\lceil \log_2 \binom{3n}{n} \right\rceil &\sim n \log_2 \frac{27}{4} - \underbrace{\frac{1}{2}\log_2 \frac{4\pi n}{3}}_{\text{low-order term}} \\ &\sim 2.755\cdots n\end{aligned}$$

The stars-and-bars approach uses $3n$ bits, which is slightly more than the lower bound of about $2.755n$ bits.

This lower bound is actually not hard to achieve, though we will not cover the details. The idea is to order all of the multisets of n numbers in the range $[0, 2n]$ and then index them $0, 1, 2, \ldots$. We then store a multiset by storing its index. The index is a number in the range 0 to $\binom{3n}{n} - 1$, which we can store in exactly $\lceil \log_2 \binom{3n}{n} \rceil$ bits. Of course, the hard part is efficiently mapping a multiset to an index and vice versa. Nevertheless, we can store a million numbers in the range zero to two million with only 2.755 million bits.

Introduction to Probability

1 Probability

Probability will be the topic for the rest of the term. Probability is one of the most important subjects in Mathematics and Computer Science. Most upper level Computer Science courses require probability in some form, especially in analysis of algorithms and data structures, but also in information theory, cryptography, control and systems theory, network design, artificial intelligence, and game theory. Probability also plays a key role in fields such as Physics, Biology, Economics and Medicine.

There is a close relationship between Counting/Combinatorics and Probability. In many cases, the probability of an event is simply the fraction of possible outcomes that make up the event. So many of the rules we developed for finding the cardinality of finite sets carry over to Probability Theory. For example, we'll apply an Inclusion-Exclusion principle for *probabilities* in some examples below.

In principle, probability boils down to a few simple rules, but it remains a tricky subject because these rules often lead unintuitive conclusions. Using "common sense" reasoning about probabilistic questions is notoriously unreliable, as we'll illustrate with many real-life examples.

This reading is longer than usual . To keep things in bounds, several sections with illustrative examples that do not introduce new concepts are marked "[Optional]." You should read these sections selectively, choosing those where you're unsure about some idea and think another example would be helpful.

2 Modelling Experimental Events

One intuition about probability is that we want to predict how likely it is for a given experiment to have a certain kind of outcome. Asking this question invariably involves four distinct steps:

Find the sample space. Determine all the possible outcomes of the experiment.

Define the event of interest. Determine which of those possible outcomes is "interesting."

Determine the individual outcome probabilities. Decide how likely each individual outcome is to occur.

Determine the probability of the event. Combine the probabilities of "interesting" outcomes to find the overall probability of the event we care about.

In order to understand these four steps, we will begin with a toy problem. We consider rolling three dice, and try to determine the probability that we roll exactly two sixes.

Step 1: Find the Sample Space

Every probability problem involves some experiment or game. The key to most probability problems is to look carefully at the *sample space* of the experiment. Informally, this is the set of all possible experimental *outcomes*. An outcome consists of the total information about the experiment after it has been performed. An outcome is also called a "sample point" or an "atomic event".

In our die rolling experiment, a particular outcome can be expressed as a triple of numbers from 1 to 6. For example, the triple $(3, 5, 6)$ indicates that the first die rolled 3, the second rolled 5, and the third rolled 6.[1]

Step 2: Define Events of Interest

We usually declare some subset of the possible outcomes in the sample space to be "good" or "interesting." Any subset of the sample space is called an *event*.

For example, the event that all dice are the same consists of six possible outcomes

$$\{(1,1,1),(2,2,2),(3,3,3),(4,4,4),(5,5,5),(6,6,6)\}.$$

Let T be the event that we roll exactly two sixes. T has $3 \cdot 5 = 15$ possible outcomes: we need to choose which die is not a six, and then we need to choose a value for that die. Namely,

$$\begin{aligned} T ::= \{&(1,6,6),(2,6,6),(3,6,6),(4,6,6),(5,6,6),\\ &(6,1,6),(6,2,6),(6,3,6),(6,4,6),(6,5,6),\\ &(6,6,1),(6,6,2),(6,6,3),(6,6,4),(6,6,5)\} \end{aligned}$$

Our goal is to determine the probability that our experiment yields one of the outcomes in this set T.

Step 3: Specify Outcome Probabilities

Assign a real number between zero and one, called a *probability*, to each outcome of an experiment so that the sum of the probabilities of all the outcomes is one. This is called specifying a *probability space* appropriate to the experiment. We use the notation, $\Pr\{w\}$, to denote the probability of an outcome w.

Assigning probabilities to the atomic outcomes is an *axiomatic* action. One of the philosophical bases for probability says that the probability for an outcome should be the fraction of times that we expect to see that outcome when we carry out a large number of experiments. Thinking of the probabilities as fractions of one whole set of outcomes makes it plausible that probabilities should be nonnegative and sum to one.

In our experiment (and in many others), it seems quite plausible to say that all the possible outcomes are equally likely. Probability spaces of this kind are called *uniform*:

[1] Notice that we're assuming the dice are distinguishable—say they are different colors—so we know which is which. We would need a different sample space of outcomes if we regarded the dice as *in*distinguishable.

Definition 2.1. A *uniform* probability space is a finite space in which all the outcomes have the same probability. That is, if $\mathcal{S}$ is the sample space, then

$$\Pr\{w\} = \frac{1}{|\mathcal{S}|}$$

for every outcome $w \in \mathcal{S}$.

Since there are $6^3 = 216$ possible outcomes, we axiomatically declare that each occurs with probability $1/216$.

Step 4: Compute Event Probabilities

We now have a probability for each outcome. To compute the probability of the event, T, that we get exactly two sixes, we add up the probabilities of all the outcomes that yield exactly two sixes. In our example, since there are 15 outcomes in T, each with probability $1/216$, we can deduce that $\Pr\{T\} = 15/216$.

Probability on a uniform sample space such as this one is pretty much the same as counting. Another example where it's reasonable to use a uniform space is for poker hands. Instead of asking how many distinct full houses there are in poker, we can ask about the probability that a "random" poker hand is a full house. For example, of the $\binom{52}{5}$ possible poker hands, we saw that

- There are 624 "four of a kind" hands, so the probability of 4 of a kind is $624/\binom{52}{5} = 1/4165$.
- There are 3744 "full house" hands, so the probability of a full house is $6/4165 \approx 1/694$.
- There are 123,552 "two pair" hands, so the probability of two pair $\approx 1/21$.

3 The Monty Hall Problem

In the 1970's, there was a game show called Let's Make a Deal, hosted by Monty Hall and his assistant Carol Merrill. At one stage of the game, a contestant is shown three doors. The contestant knows there is a prize behind one door and that there are goats behind the other two. The contestant picks a door. To build suspense, Carol always opens a *different* door, revealing a goat. The contestant can then stick with his original door or switch to the other unopened door. He wins the prize only if he now picks the correct door. Should the contestant "stick" with his original door, "switch" to the other door, or does it not matter?

This was the subject of an "Ask Marilyn" column in *Parade* Magazine a few years ago. Marilyn wrote that your chances of winning were $2/3$ if you switched — because if you switch, then you win if the prize was originally behind either of the two doors you didn't pick. Now, Marilyn has been listed in the *Guiness Book of World Records* as having the world's highest IQ, but for this answer she got a tidal wave of critical mail, some of it from people with Ph.D.'s in mathematics, telling her she was wrong. Most of her critics insisted that the answer was $1/2$, on the grounds that the prize was equally likely to be behind each of the doors, and since the contestant knew he was going to see a goat, it remains equally likely which the two remaining doors has the prize behind it. The pros and cons of these arguments still stimulate debate.

It turned out that Marilyn was right, but given the debate, it is clearly not apparent which of the intuitive arguments for 2/3 or 1/2 is reliable. Rather than try to come up with our own explanation in words, let's use our standard approach to finding probabilities. In particular, we will analyze the probability that the contestant wins with the "switch" strategy; that is, the contestant chooses a random door initially and then always switches after Carol reveals a goat behind one door. We break the down into the standard four steps.

Step 1: Find the Sample Space

In the Monty Hall problem, an outcome is a triple of door numbers:

1. The number of the door concealing the prize.
2. The number of the door initially chosen by the contestant.
3. The number of the door Carol opens to reveal a goat.

For example, the outcome $(2, 1, 3)$ represents the case where the prize is behind door 2, the contestant initially chooses door 1, and Carol reveals the goat behind door 3. In this case, a contestant using the "switch" strategy wins the prize.

Not every triple of numbers is an outcome; for example, $(1, 2, 1)$ is not an outcome, because Carol never opens the door with the prize. Similarly, $(1, 2, 2)$ is not an outcome, because Carol does not open the door initially selected by the contestant, either.

As with counting, a tree diagram is a standard tool for studying the sample space of an experiment. The tree diagram for the Monty Hall problem is shown in Figure 1. Each vertex in the tree corresponds to a state of the experiment. In particular, the root represents the initial state, before the prize is even placed. Internal nodes represent intermediate states of the experiment, such as after the prize is placed, but before the contestant picks a door. Each leaf represents a final state, an outcome of the experiment. One can think of the experiment as a walk from the root (initial state) to a leaf (outcome). In the figure, each leaf of the tree is labeled with an outcome (a triple of numbers) and a "W" or "L" to indicate whether the contestant wins or loses.

Step 2: Define Events of Interest

For the Monty Hall problem, let $\mathcal{S}$ denote the sample space, the set of all 12 outcomes shown in Figure 1. The event $W \subset \mathcal{S}$ that the contestant wins with the "switch" strategy consists of six outcomes:

$$W ::= \{(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1)\}.$$

The event $L \subset \mathcal{S}$ that the contestant loses is the complementary set:

$$L ::= \{(1, 1, 2), (1, 1, 3), (2, 2, 1), (2, 2, 3), (3, 3, 1), (3, 3, 2)\}.$$

Our goal is to determine the probability of the event W; that is, the probability that the contestant wins with the "switch" strategy.

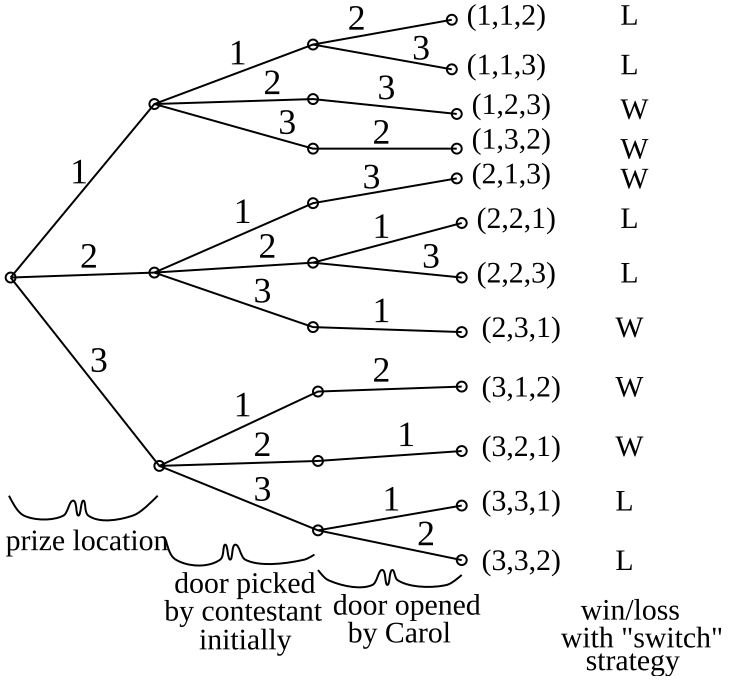


Figure 1: This is a tree diagram for the Monty Hall problem. Each of the 12 leaves of the tree represents an outcome. A “W” next to an outcome indicates that the contestant wins, and an “L” indicates that he loses.

Well, the contestant wins in 6 outcomes and loses in 6 outcomes. Does this not imply that the contestant has a $6/12 = 1/2$ chance of winning? No! Under our natural assumptions, this sample space is not uniform! Some outcomes may be more likely than others. We must compute the probability of each outcome.

Step 3: Compute Outcome Probabilities

3.1 Assumptions

To assign a meaningful probability to each outcome in the Monty Hall problem, we must make some assumptions. The following three are sufficient:

1. The prize is placed behind each door with probability $1/3$.
2. No matter where the prize is placed, the contestant picks each door with probability $1/3$.
3. No matter where the prize is placed, if Carol has a choice of which door to open, then she opens each possible door with equal probability.

The first two assumptions capture the idea that the contestant initially has no idea where the prize is placed. The third assumption eliminates the possibility that Carol somehow secretly communicates the location of the prize by which door she opens. Assumptions of this sort almost always arise in probability problems; making them explicit is a good idea, although in fact not all of these assumptions are absolutely necessary. For example, it doesn't matter how Carol chooses a door to open in the cases when she has a choice, though we won't prove this.

3.2 Assigning Probabilities to Outcomes

With these assumptions, we can assign probabilities to outcomes in the Monty Hall problem by a calculation illustrated in Figure 2 and described below. There are two steps.

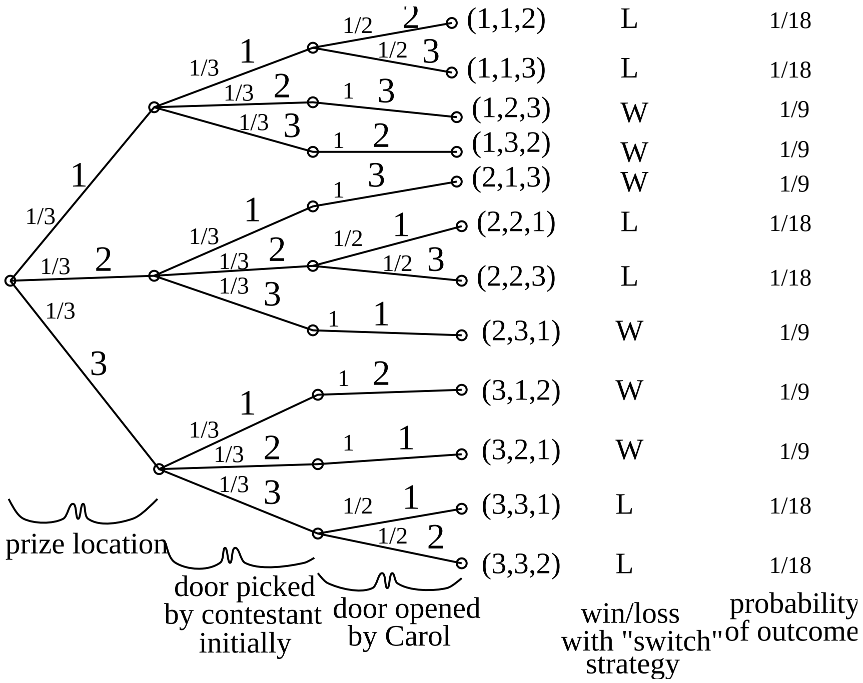


Figure 2: This is the tree diagram for the Monty Hall problem, annotated with probabilities for each outcome.

The first step is to record a probability on each edge in the tree diagram. Recall that each node represents a state of the experiment, and the whole experiment can be regarded as a walk from the root (initial state) to a leaf (outcome). The probability recorded on an edge is the probability of moving from the state corresponding to the parent node to the state corresponding to the child node. These edge probabilities follow from our three assumptions about the Monty Hall problem.

Specifically, the first assumption says that there is a $1/3$ chance that the prize is placed behind each of the three doors. This gives the $1/3$ probabilities on the three edges from the root. The second assumption says that no matter how the prize is placed, the contestant opens each door with probability $1/3$. This gives the $1/3$ probabilities on edges leaving the second layer of nodes. Finally, the third assumption is that if Carol has a choice of what door to open, then she opens each with equal probability. In cases where Carol has no choice, edges from the third layer of nodes are labeled with probability 1. In cases where Carol has two choices, edges are labeled with probability $1/2$.

The second step is to use the edge weights to compute a probability for each outcome by multiplying the probabilities along the edges leading to the outcome. This way of assigning probabilities reflects our idea that probability measures the fraction of times that a given outcome should happen over the course of many experiments. Suppose we want the probability of outcome $(2, 1, 3)$. In $1/3$ of the experiments, the prize is behind the second door. Then, in $1/3$ of these experiments when the prize is behind the second door, and the contestant opens the first door. After that, Carol has no choice but to open the third door. Therefore, the probability of the outcome is the product of the edge probabilities, which is

$$\frac{1}{3} \cdot \frac{1}{3} \cdot 1 = \frac{1}{9}.$$

For example, the probability of outcome $(2, 2, 3)$ is the product of the edge probabilities on the path from the root to the leaf labeled $(2, 2, 3)$. Therefore, the probability of the outcome is

$$\frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{2} = \frac{1}{18}.$$

Similarly, the probability of outcome $(3, 1, 2)$ is

$$\frac{1}{3} \cdot \frac{1}{3} \cdot 1 = \frac{1}{9}.$$

The other outcome probabilities are worked out in Figure 2.

Step 4: Compute Event Probabilities

We now have a probability for each outcome. All that remains is to compute the probability of W, the event that the contestant wins with the "switch" strategy. The probabilility of an event is simply the sum of the probabilities of all the outcomes in it. So the probability of the contestant winning with the "switch" strategy is the sum of the probabilities of the six outcomes in event W, namely, $2/3$:

$$\begin{aligned}
\Pr\{W\} \quad &::= \quad \Pr\{(1,2,3)\} + \Pr\{(1,3,2)\} + \Pr\{(2,1,3)\} + \Pr\{(2,3,1)\} + \Pr\{(3,1,2)\} + \Pr\{(3,2,1)\} \\
&= \quad \frac{1}{9} + \frac{1}{9} + \frac{1}{9} + \frac{1}{9} + \frac{1}{9} + \frac{1}{9} \\
&= \quad \frac{2}{3}.
\end{aligned}$$

In the same way, we can compute the probability that a contestant loses with the "switch" strategy. This is the probability of event L:

$$\begin{aligned}
\Pr\{L\} \quad &::= \quad \Pr\{(1,1,2)\} + \Pr\{(1,1,3)\} + \Pr\{(2,2,1)\} + \Pr\{(2,2,3)\} + \Pr\{(3,3,1)\} + \Pr\{(3,3,2)\} \\
&= \quad \frac{1}{18} + \frac{1}{18} + \frac{1}{18} + \frac{1}{18} + \frac{1}{18} + \frac{1}{18} \\
&= \quad \frac{1}{3}.
\end{aligned}$$

The probability of the contestant losing with the switch strategy is $1/3$. This makes sense; the probability of winning and the probability of losing ought to sum to 1!

We can determine the probability of winning with the "stick" strategy without further calculations. In every case where the "switch" strategy wins, the "stick" strategy loses, and vice versa. Therefore, the probability of winning with the stick strategy is $1 - 2/3 = 1/3$.

Solving the Monty Hall problem formally requires only simple addition and multiplication. But trying to solve the problem with "common sense" leaves us running in circles!

4 Intransitive Dice

There is a game involving three dice and two players. The dice are not normal; rather, they are numbered as shown in Figure 3. Each hidden face has the same number as the opposite, exposed face. As a result, each die has only three distinct numbers, and each number comes up $1/3$ of the time.

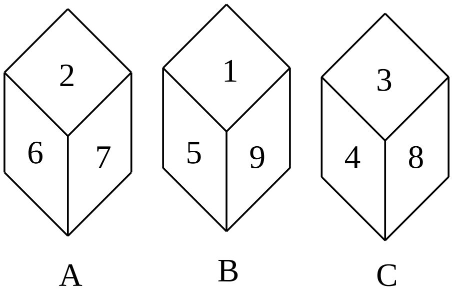


Figure 3: This figure shows the strange numbering of the three dice "intransitive" dice. The number on each concealed face is the same as the number on the exposed, opposite face.

In the game, the first player can choose any one of the three dice. Then the second player chooses one of the two remaining dice. They both roll and the player with the higher number wins. Which of the three dice should player one choose? That is, which of the three dice is best?

For example, die B is attractive, because it has a 9, the highest number overall; on the other hand, it also has a 1, the lowest number. Intuition gives no clear answer! We can solve the problem with our standard four-step method.

Claim 4.1. *Die A beats die B more than half of the time.*

Proof. The claim concerns the experiment of throwing dice A and B.

Step 1: Find the Sample Space. The sample space for this experiment is indicated by the tree diagram in Figure 4.

Step 2: Define Events of Interest. We are interested in the event that die A comes up greater than die B. The outcomes in this event are marked "A" in the figure.

Step 3: Compute Outcome Probabilities. To find outcome probabilities, we first assign probabilities to edges in the tree diagram. Each number comes up with probability $1/3$, regardless of the value of the other die. Therefore, we assign all edges probability $1/3$. The probability of an outcome is the product of probabilities on the corresponding root-to-leaf path; this means that every outcome has probability $1/9$.

Step 4: Compute Event Probabilities. The probability of an event is the sum of the probabilities of the outcomes in the event. Therefore, the probability that die A comes up greater than die "B" is

$$\frac{1}{9}+\frac{1}{9}+\frac{1}{9}+\frac{1}{9}+\frac{1}{9}=\frac{5}{9}.$$

As claimed, the probability that die A beats die B is greater than half.

□

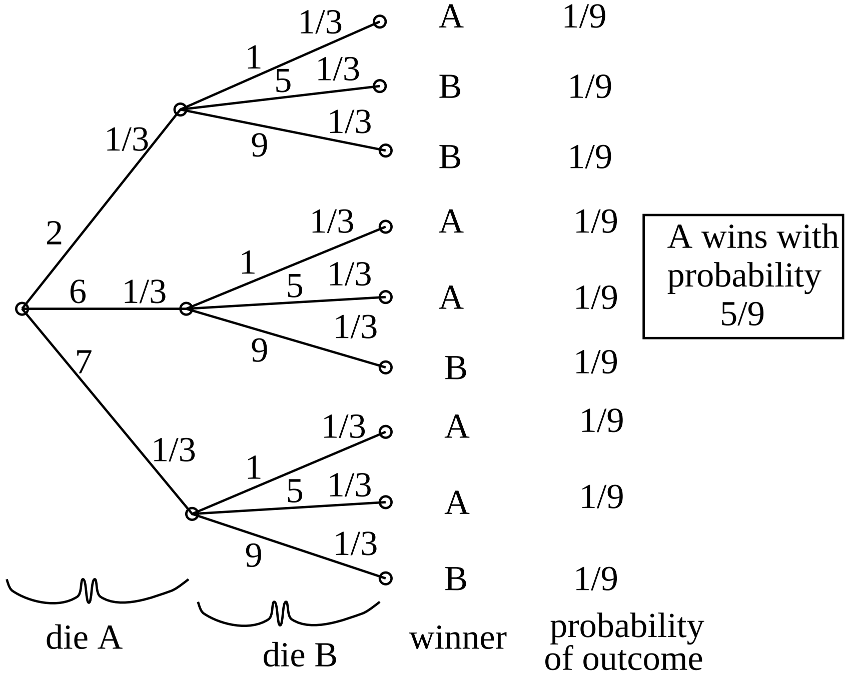


Figure 4: This is the tree diagram arising when die A is played against die B. Die A beats die B with probability $5/9$.

The analysis may be even clearer by giving the outcomes in a table:

Winner		B roll		
		1	5	9
A roll	2	A	B	B
	6	A	A	B
	7	A	A	B

All the outcomes are equally likely, and we see that A wins 5 of them. This table works because our probability space is based on 2 pieces of information, A's roll and B's roll. For more complex probability spaces, the tree diagram is necessary.

Claim 4.2. *Die B beats die C more than half of the time.*

Proof. The proof is by the same case analysis as for the preceding claim, summarized in the table:

Winner		C roll		
		3	4	8
B roll	1	C	C	C
	5	B	B	C
	9	B	B	B

□

We have shown that A beats B and that B beats C. From these results, we might conclude that A is the best die, B is second best, and C is worst. But this is totally wrong!

Claim 4.3. *Die C beats die A more than half of the time!*

Proof. See the tree diagram in Figure 5. Again, we can present this analysis in a tabular form:

Winner		A roll		
		2	6	7
C roll	3	C	A	A
	4	C	A	A
	8	C	C	C

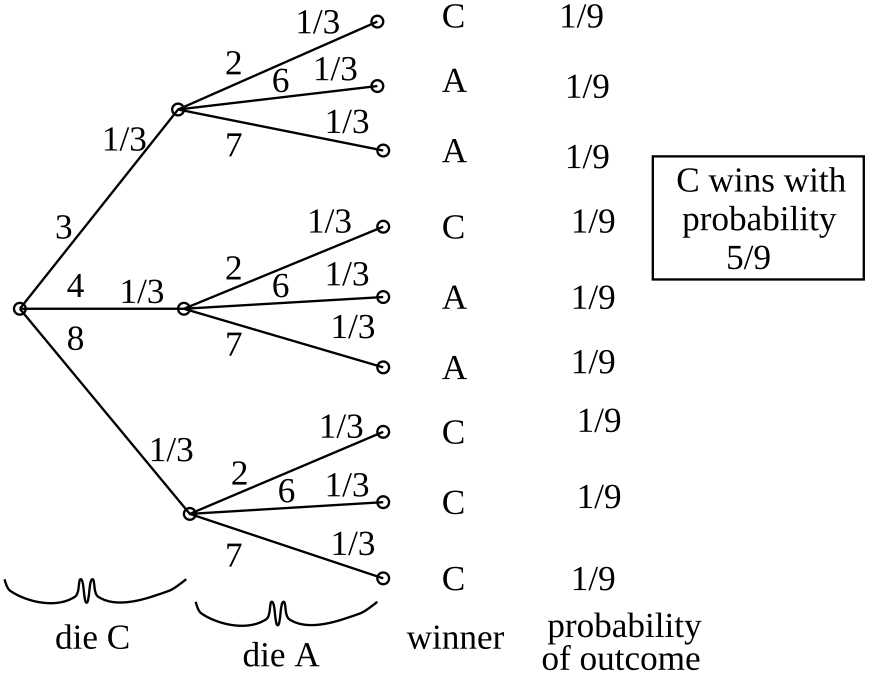


Figure 5: Die C beats die A with probability $5/9$. Amazing!

□

Die A beats B, B beats C, and C beats A! Apparently, there is no "transitive law" here! This means that no matter what die the first player chooses, the second player can choose a die that beats it with probability $5/9$. The player who picks first is always at a disadvantage!

[Optional]

The same effect can arise with three dice numbered the ordinary way, but "loaded" so that some numbers turn up more often. For example, suppose:

A	rolls	3 with probability 1
B	rolls	2 with probability $p ::= (\sqrt{5}-1)/2 = 0.618\ldots$
	rolls	5 with probability $1-p$
C	rolls	1 with probability $1-p$
	rolls	4 with probability p

It's clear that A beats B, and C beats A, each with probability p. But note that $1 - p^2 = p$. Now the probability that B beats C is

$$\Pr\{\text{B rolls to 5}\} + \Pr\{\text{B rolls to 2 and C rolls to 1}\} = (1-p) + p(1-p) = 1 - p^2 = p.$$

So A beats B, B beats C, and C beats A, all with probability $p = 0.618\cdots > 5/9$.

5 Set Theory and Probability

Having gone through these examples, we should be ready to make sense of the formal definitions of basic probability theory.

5.1 Basic Laws of Probability

Definition 5.1. A *sample space*, $\mathcal{S}$, is a nonempty set whose elements are called *outcomes*. The *events* are subsets of $\mathcal{S}$.[2]

Definition. A family, $\mathcal{F}$, of sets is *pairwise disjoint* if the intersection of every pair of distinct sets in the family is empty, *i.e.*, if $A, B \in \mathcal{F}$ and $A \neq B$, then $A \cap B = \emptyset$. In this case, if $\mathcal{S} = \bigcup \mathcal{F}$, then $\mathcal{S}$ is said to be the *disjoint union* of the sets in $\mathcal{F}$.

Definition 5.2. A *probability space* consists of a sample space, $\mathcal{S}$, and a *probability function*, $\Pr\{\}$, mapping the events of $\mathcal{S}$ to real numbers between zero and one, such that:

1. $\Pr\{S\} = 1$, and

[2] For all the examples in 6.042, we let every subset of $\mathcal{S}$ be an event. However, when $\mathcal{S}$ is a set such as the unit interval of real numbers, there can be problems. In this case, we typically want subintervals of the unit interval to be events with probability equal to their length. For example, we'd say that if a dart hit "at random" in the unit interval, then the probability that it landed within the subinterval from 1/3 to 3/4 was equal to the length of the interval, namely 5/12.

Now it turns out to be inconsistent with the axioms of Set Theory to insist that *all* subsets of the unit interval be events. Instead, the class of events must be limited to rule out certain pathological subsets which do not have a well-defined length. An example of such a pathological set is the real numbers between zero and one with an infinite number of fives in the even-numbered positions of their decimal expansions. Fortunately, such pathological subsets are not relevant in applications of Probability Theory.

The results of the Probability Theory hold as long as we have some set of events with a few basic properties: every finite set of outcomes is an event, the whole space is an event, the complement of an event is an event, and if $A_0, A_1, \ldots$ are events, so is $\bigcup_{i \in \mathbb{N}} A_i$. It is easy to come up with such a class of events that includes all the events we care about and leaves out all the pathological cases.

2. if $A_0, A_1, \ldots$ is a sequence of disjoint events, then

$$\Pr\left\{\bigcup_{i\in\mathbb{N}} A_i\right\} = \sum_{i\in\mathbb{N}} \Pr\{A_i\}. \qquad \text{(Sum Rule)}$$

The Sum Rule[3] lets us analyze a complicated event by breaking it down into simpler cases. For example, if the probability that a randomly chosen MIT student is native to the United States is 60%, to Canada is 5%, and to Mexico is 5%, then the probability that a random MIT student is native to North America is 70%.

One immediate consequence of Definition 5.2 is that $\Pr\{A\} + \Pr\{\overline{A}\} = 1$ because $\mathcal{S}$ is the disjoint union of A and $\overline{A}$. This equation often comes up in the form

$$\Pr\{\overline{A}\} = 1 - \Pr\{A\}. \qquad \text{(Complement Rule)}$$

Some further basic facts about probability parallel facts about cardinalities of finite sets. In particular:

$$\Pr\{B - A\} = \Pr\{B\} - \Pr\{A \cap B\} \qquad \text{(Difference Rule)}$$
$$\Pr\{A \cup B\} = \Pr\{A\} + \Pr\{B\} - \Pr\{A \cap B\} \qquad \text{(Inclusion-Exclusion)}$$

The Difference Rule follows from the Sum Rule because B is the disjoint union of $B - A$ and $A \cap B$. The (Inclusion-Exclusion) equation then follows from the Sum and Difference Rules, because $A \cup B$ is the disjoint union of A and $B - A$, so

$$\Pr\{A \cup B\} = \Pr\{A\} + \Pr\{B - A\} = \Pr\{A\} + (\Pr\{B\} - \Pr\{A \cap B\}).$$

This (Inclusion-Exclusion) equation is the Probability Theory version of the Inclusion-Exclusion Principle for the size of the union of two finite sets. It generalizes to n events in a corresponding way. An immediate consequence of (Inclusion-Exclusion) is

$$\Pr\{A \cup B\} \leq \Pr\{A\} + \Pr\{B\}. \qquad \text{(Boole's Inequality)}$$

Similarly, the Difference Rule implies that

$$\text{If } A \subseteq B, \text{ then } \Pr\{A\} \leq \Pr\{B\}. \qquad \text{(Monotonicity)}$$

In the examples we considered above, we used the fact that the probability of an event was the sum of the probabilities of its outcomes. This follows as a trivial special case of the Sum Rule with one quibble: according to the official definition, the probability function is defined on *events* not outcomes. But we can always treat an outcome as the event whose only element is that outcome, that is, define $\Pr\{w\}$ to be $\Pr\{\{w\}\}$. Then, for the record, we can say

Corollary 5.3. *If $A = \{w_0, w_1, \ldots\}$ is an event, then*

$$\Pr\{A\} = \sum_{i\in\mathbb{N}} \Pr\{w_i\}.$$

[3] If you think like a Mathematician, you should be wondering if the infinite sum is really necessary. Namely, suppose we had only used finite sums in Definition 5.2 instead of sums over all natural numbers. Would this imply the result for infinite sums? It's hard to find counterexamples, but there are some: it is possible to find a pathological "probability" measure on a sample space satisfying the Sum Rule for finite unions, in which the outcomes $w_0, w_1, \ldots$ each have probability zero, and the probability assigned to any event is either zero or one! So the infinite Sum Rule fails dramatically, since the whole space is of measure one, but it is a union of the outcomes of measure zero.

The construction of such weird examples is beyond the scope of 6.042. You can learn more about this by taking a course in Set Theory and Logic that covers the topic of "ultrafilters."

5.2 Circuit Failure

Suppose you are wiring up a circuit containing a total of n connections. From past experience we assume that any particular connection is made *incorrectly* with probability p, for some $0 \leq p \leq 1$. That is, for $1 \leq i \leq n$,

$$\Pr\{i\text{th connection is wrong}\} = p.$$

What can we say about the probability that the circuit is wired correctly, *i.e.*, that it contains no incorrect connections?

Let A_i denote the event that connection i is made *correctly*. Then $\overline{A_i}$ is the event that connection i is made incorrectly, so $\Pr\{\overline{A_i}\} = p$. Now

$$\Pr\{\text{all connections are OK}\} = \Pr\left\{\bigcap_{i=1}^{n} A_i\right\}.$$

Without any additional assumptions, we can't get an exact answer. However, we can give reasonable upper and lower bounds. For an upper bound, we can see that

$$\Pr\left\{\bigcap_{i=1}^{n} A_i\right\} = \Pr\left\{A_1 \cap (\bigcap_{i=2}^{n} A_i)\right\} \leq \Pr\{A_1\} = 1 - p$$

by Monotonicity. For a lower bound, we can see that

$$\Pr\left\{\bigcap_{i=1}^{n} A_i\right\} = 1 - \Pr\left\{\overline{\bigcap_{i=1}^{n} A_i}\right\} = 1 - \Pr\left\{\bigcup_{i=1}^{n} \overline{A_i}\right\} \geq 1 - \sum_{i=1}^{n} \Pr\{\overline{A_i}\} = 1 - np,$$

where the $\geq$-inequality follows from Boole's Law.

So for example, if $n = 10$ and $p = 0.01$, we get the following bounds:

$$0.9 = 1 - 10 \cdot 0.01 \leq \Pr\{\text{all connections are OK}\} \leq 1 - 0.01 = 0.99.$$

So we have concluded that the chance that all connections are okay is somewhere between 90% and 99%. Could it actually be as high as 99%? Yes, if the errors occur in such a way that all connection errors always occur at the same time.

Could it be 90%? Yes, suppose the errors are such that we never make two wrong connections. In other words, the events $\overline{A_i}$ are all disjoint and the probability of getting it right is

$$\Pr\left\{\bigcap A_i\right\} = 1 - \Pr\left\{\bigcup \overline{A_i}\right\} = 1 - \sum_{i=1}^{10} \Pr\{\overline{A_i}\} = 1 - 10 \cdot 0.01 = 0.9.$$

6 Combinations of Events

6.1 Carnival Dice

There is a gambling game called Carnival Dice. A player picks a number between 1 and 6 and then rolls three *fair* dice—"fair" means each number is equally likely to show up on a die. The player wins if his number comes up on at least one die. The player loses if his number does not appear on any of the dice. What is the probability that the player wins? This problem sounds simple enough that we might try an intuitive lunge for the solution.

False Claim 6.1. *The player wins with probability* $1/2$.

False proof. Let A_i be the event that the ith die matches the player's guess.

$$\begin{aligned}
\Pr\{win\} &= \Pr\{A_1 \cup A_2 \cup A_3\} && (1)\\
&= \Pr\{A_1\} + \Pr\{A_2\} + \Pr\{A_3\} && (2)\\
&= \frac{1}{6} + \frac{1}{6} + \frac{1}{6} && (3)\\
&= \frac{1}{2} && (4)
\end{aligned}$$

□

The justification for the equality (2) is that the union is disjoint. This may seem reasonable in a vague way, but in a precise way it's not. To see that this is a silly argument, note that it would also imply that with six dice, our probability of getting a match is 1, *i.e.*, it is sure to happen. This is clearly false—there is some chance that none of the dice match.[4]

To compute the actual chance of winning at Carnival Dice, we can use Inclusion-Exclusion for three sets. The probability that one die matches the player's guess is $1/6$. The probability that two particular dice both match the player's guess is $1/36$: there are 36 possible outcomes of the two dice and exactly one of them has both equal to the player's guess. The probability that all three dice match is $1/216$. Inclusion-Exclusion gives:

$$\Pr\{\text{win}\} = \frac{1}{6} + \frac{1}{6} + \frac{1}{6} - \frac{1}{36} - \frac{1}{36} - \frac{1}{36} + \frac{1}{216} = \frac{91}{216} \approx 42\%.$$

These are terrible odds in a gambling game; it is much better to play roulette, craps, or blackjack!

6.2 More Intransitive Dice [Optional]

[Optional]

In Section 4, we described three dice A, B and C such that the probabilities of A beating B, B beating C, C beating A are each $p ::= (\sqrt{5} - 1)/2 \approx 0.618$. Can we increase this probability? For example, can we design dice so that each of these probabilities are, say, at least 3/4? The answer is "No." In fact, using the elementary rules of probability, it's easy to show that these "beating" probabilities cannot all exceed 2/3.

In particular, we consider the experiment of rolling all three dice, and define $[A]$ to be the event that A beats B, $[B]$ the event that B beats C, and $[C]$ the event that C beats A.

Claim.

$$\min\{\Pr\{[A]\}, \Pr\{[B]\}, \Pr\{[C]\}\} \leq \frac{2}{3}. \qquad (5)$$

[4] On the other hand, the idea of adding these probabilities is not completely absurd. We will see in Course Notes 11 that adding would work to compute the *average* number of matching dice: $1/2$ a match per game with three dice and one match per game in the game with six dice.

Proof. Suppose dice A, B, C roll numbers a, b, c. Events $[A], [B], [C]$ all occur on this roll iff $a > b, b > c, c > a$, so in fact they cannot occur simultaneously. That is,

$$[A] \cap [B] \cap [C] = \emptyset. \tag{6}$$

Therefore,

$$\begin{aligned}
0 &= \Pr\{[A] \cap [B] \cap [C]\} && \text{(by (6))}\\
&= 1 - \Pr\left\{\overline{[A]} \cup \overline{[B]} \cup \overline{[C]}\right\} && \text{(Complement Rule and DeMorgan)}\\
&\geq 1 - (\Pr\left\{\overline{[A]}\right\} + \Pr\left\{\overline{[B]}\right\} + \Pr\left\{\overline{[C]}\right\} && \text{(Boole's Inequality)}\\
&= \Pr\{[A]\} + \Pr\{[B]\} + \Pr\{[C]\}) - 2 && \text{(Complement Rule)}\\
&\geq 3\min\{\Pr\{[A]\}, \Pr\{[B]\}, \Pr\{[C]\}\} - 2. && \text{(def of min)}
\end{aligned}$$

Hence

$$2 \geq 3\min\{\Pr\{[A]\}, \Pr\{[B]\}, \Pr\{[C]\}\},$$

proving (5). □

6.3 Derangements [Optional]

[Optional]

Suppose we line up two randomly ordered decks of n cards against each other. What is the probability that at least one pair of cards "matches"? Let A_i be the event that card i is in the same place in both arrangements. We are interested in $\Pr\{\bigcup A_i\}$. To apply the Inclusion-Exclusion formula, we need to compute the probabilities of individual intersection events—namely, to determine the probability $\Pr\{A_{i_1} \cap A_{i_2} \cap \cdots \cap A_{i_k}\}$ that a particular set of k cards matches. To do so we apply our standard four steps.

The sample space. The sample space involves a permutation of the first card deck and a permutation of the second deck. We can think of this as a tree diagram: first we permute the first deck ($n!$ ways) and then, for each first deck arrangement, we permute the second deck ($n!$ ways). By the product rule for sets, we get $(n!)^2$ arrangements.

Determine atomic event probabilities. We assume a uniform sample space, so each event has probability $1/(n!)^2$.

Determine the event of interest. These are the arrangements where cards $i_1, \ldots, i_k$ are all in the same place in both permutations.

Find the event probability. Since the sample space is uniform, this is equivalent to determining the *number* atomic events in our event of interest. Again we use a tree diagram. There are $n!$ permutations of the first deck. Given the first deck permutation, how many second deck permutations line up the specified cards? Well, those k cards must go in specific locations, while the remaining $n-k$ cards can be permuted arbitrarily in the remaining $n-k$ locations in $(n-k)!$ ways. Thus, the total number of atomic events of this type is $n!(n-k)!$, and the probability of the event in question is

$$\frac{n!(n-k)!}{n!n!} = \frac{(n-k)!}{n!}.$$

We have found that the probability a specific set of k cards matches is $(n-k)!/n!$. There are $\binom{n}{k}$ such sets of k cards. So the k^{th} Inclusion-Exclusion term is

$$\binom{n}{k}\frac{(n-k)!}{n!} = 1/k!.$$

Thus, the probability of at least one match is

$$1 - 1/2! + 1/3! - \cdots \pm 1/n!$$

We can understand this expression by thinking about the Taylor expansion of

$$e^{-x} = 1 - x + x^2/2! - x^3/3! + \cdots.$$

In particular,

$$e^{-1} = 1 - 1 + 1/2! - 1/3! + \cdots.$$

Our expression takes the first n terms of the Taylor expansion; the remainder is negligible—it is in fact less than $1/(n+1)!$—so our probability is approximately $1 - 1/e$.

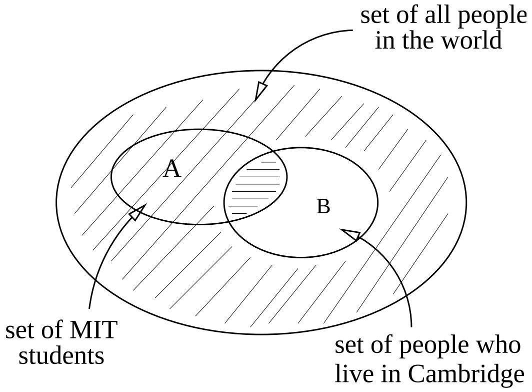


Figure 6: What is the probability that a random person in the world is an MIT student, given that the person is a Cambridge resident?

7 Conditional Probability

Suppose that we pick a random person in the world. Everyone has an equal chance of being picked. Let A be the event that the person is an MIT student, and let B be the event that the person lives in Cambridge. The situation is shown in Figure 6. Clearly, both events A and B have low probability. But what is the probability that a person is an MIT student, *given* that the person lives in Cambridge? This is a conditional probability question. It can be concisely expressed in a special notation. In general, $\Pr\{A \mid B\}$ denotes the probability of event A, given event B. In this example, $\Pr\{A \mid B\}$ is the probability that the person is an MIT student, given that he or she is a Cambridge resident.

How do we compute $\Pr\{A \mid B\}$? Since we are *given* that the person lives in Cambridge, all outcomes outside of event B are irrelevant; these irrelevant outcomes are diagonally shaded in the figure. Intuitively, $\Pr\{A \mid B\}$ should be the fraction of Cambridge residents that are also MIT students. That is, the answer should be the probability that the person is in set $A \cap B$ (horizontally shaded) divided by the probability that the person is in set B. This leads us to

Definition 7.1.

$$\Pr\{A \mid B\} ::= \frac{\Pr\{A \cap B\}}{\Pr\{B\}}$$

providing $\Pr\{B\} \neq 0$.

Rearranging terms gives the following

Rule 7.2 (Product Rule, base case). *Let A and B be events, with* $\Pr\{B\} \neq 0$. *Then*

$$\Pr\{A \cap B\} = \Pr\{B\} \cdot \Pr\{A \mid B\}.$$

Note that we are now using the term "Product Rule" for two separate ideas. One is the rule above, and the other is the formula for the cardinality of a product of sets. In the rest of this lecture, the phrase always refers to the rule above. We will see the connection between these two product rules shortly, when we study independent events.

As an example, what is $\Pr\{B \mid B\}$? That is, what is the probability of event B, given that event B happens? Intuitively, this ought to be 1! The Product Rule gives exactly this result if $\Pr\{B\} \neq 0$:

$$\begin{aligned}\Pr\{B \mid B\} &= \frac{\Pr\{B \cap B\}}{\Pr\{B\}} \\ &= \frac{\Pr\{B\}}{\Pr\{B\}} \\ &= 1\end{aligned}$$

A routine induction proof based on the special case leads to The Product Rule for n events.

Rule 7.3 (Product Rule, general case). *Let $A_1, A_2, \ldots, A_n$ be events.*

$$\Pr\{A_1 \cap A_2 \cap \cdots \cap A_n\} = \Pr\{A_1\}\Pr\{A_2 \mid A_1\}\Pr\{A_3 \mid A_1 \cap A_2\} \cdots \Pr\{A_n \mid A_1 \cap \cdots \cap A_{n-1}\}$$

7.1 Conditional Probability Identities

All our probability identities continue to hold when all probabilities are conditioned on the same event. For example,

$$\Pr\{A \cup B \mid C\} = \Pr\{A \mid C\} + \Pr\{B \mid C\} - \Pr\{A \cap B \mid C\} \quad \text{(Conditional Inclusion-Exclusion)}$$

The identities carry over because for any event C, we can define a new probability measure, $\Pr_C\{\}$ on the same sample space by the rule that

$$\Pr_C\{A\} ::= \Pr\{A \mid C\}.$$

Now the conditional-probability version of an identity is just an instance of the original identity using the new probability measure.

Problem 1. Prove that for any probability space, $\mathcal{S}$, and event $C \subseteq \mathcal{S}$, the function $\Pr_C\{\}$ is a probability measure on $\mathcal{S}$.

In carrying over identities to conditional versions, a common blunder is mixing up events before and after the conditioning bar. For example, the following is *not* a consequence of the Sum Rule:

False Claim 7.4.

$$\Pr\{A \mid B \cup C\} = \Pr\{A \mid B\} + \Pr\{A \mid C\} \qquad (B \cap C = \emptyset)$$

A counterexample is shown in Figure 7. In this case, $\Pr\{A \mid B\} = 1$, $\Pr\{A \mid C\} = 1$, and $\Pr\{A \mid B \cup C\} = 1$. However, since $1 \neq 1 + 1$, the equation above does not hold.

7.2 Conditional Probability Examples

This section contains as series of examples of conditional probability problems. Trying to solve conditional problems by intuition can be very difficult. On the other hand, we can chew through these problems with our standard four-step method along with the Product Rule.

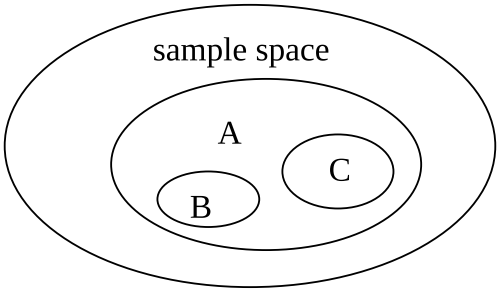


Figure 7: This figure illustrates a case where the equation $\Pr\{A \mid B \cup C\} = \Pr\{A \mid B\} + \Pr\{A \mid C\}$ does not hold.

7.2.1 A Two-out-of-Three Series

The MIT EECS department's famed D-league hockey team, The Halting Problem, is playing a 2-out-of-3 series. That is, they play games until one team wins a total of two games. The probability that The Halting Problem wins the first game is $1/2$. For subsequent games, the probability of winning depends on the outcome of the preceding game; the team is energized by victory and demoralized by defeat. Specifically, if The Halting Problem wins a game, then they have a $2/3$ chance of winning the next game. On the other hand, if the team loses, then they have only a $1/3$ chance of winning the following game. What is the probability that The Halting Problem wins the 2-out-of-3 series, given that they win the first game?

This problem involves two types of conditioning. First, we are told that the probability of the team winning a game is $2/3$, *given* that they won the preceding game. Second, we are asked the odds of The Halting Problem winning the series, *given* that they win the first game.

Step 1: Find the Sample Space

The sample space for the hockey series is worked out with a tree diagram in Figure 8. Each internal node has two children, one corresponding to a win for The Halting Problem (labeled W) and one corresponding to a loss (labeled L). The sample space consists of six outcomes, since there are six leaves in the tree diagram.

Step 2: Define Events of Interest

The goal is to find the probability that The Halting Problem wins the series given that they win the first game. This suggests that we define two events. Let A be the event that The Halting Problem wins the series, and let B be the event that they win the first game. The outcomes in each event are checked in Figure 8. Our problem is then to determine $\Pr\{A \mid B\}$.

Step 3: Compute Outcome Probabilities

Next, we must assign a probability to each outcome. We begin by assigning probabilities to edges in the tree diagram. These probabilities are given explicitly in the problem statement. Specifically,

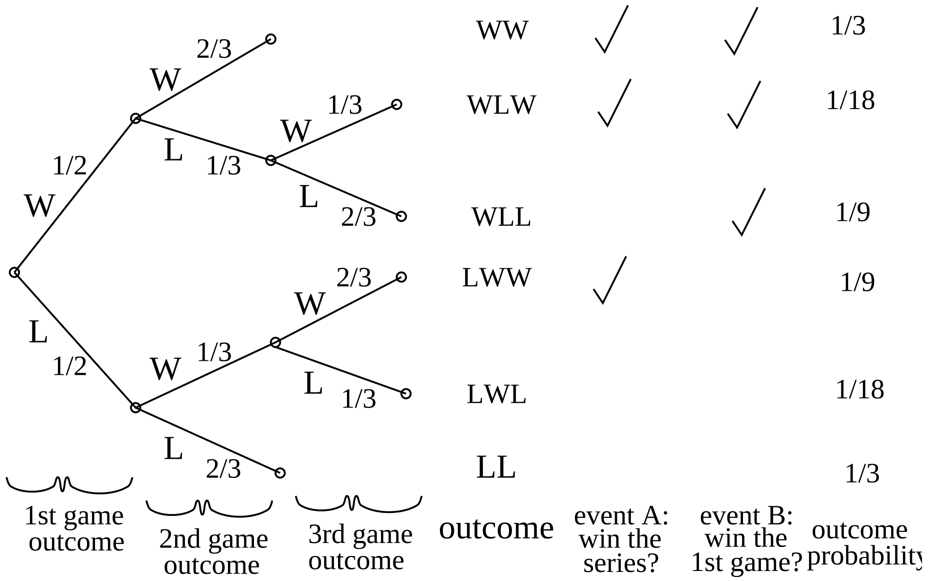


Figure 8: What is the probability that The Halting Problem wins the 2-out-of-3 series, given that they win the first game?

The Halting Problem has a $1/2$ chance of winning the first game, so the two edges leaving the root are both assigned probability $1/2$. Other edges are labeled $1/3$ or $2/3$ based on the outcome of the preceding game. We find the probability of an outcome by multiplying all probabilities along the corresponding root-to-leaf path. The results are shown in Figure 8.

This method of computing outcome probabilities by multiplying edge probabilities was introduced in our discussion of Monty Hall and Carnival Dice, but was not really justified. In fact, the justification is actually the Product Rule! For example, by multiplying edge weights, we conclude that the probability of outcome WW is

$$\frac{1}{2} \cdot \frac{2}{3} = \frac{1}{3}.$$

We can justify this rigorously with the Product Rule as follows.

$$\begin{aligned}
\Pr\{WW\} &= \Pr\{\text{win 1st game} \cap \text{win 2nd game}\} \\
&= \underbrace{\Pr\{\text{win 1st game}\} \cdot \Pr\{\text{win 2nd game} \mid \text{win 1st game}\}}_{\substack{\text{product of edge weights on} \\ \text{root-to-leaf path}}} \\
&= \frac{1}{2} \cdot \frac{2}{3} \\
&= \frac{1}{3}
\end{aligned}$$

The first equation states that WW is the outcome in which we win the first game and win the second game. The second equation is an application of the Product Rule. In the third step, we substitute probabilities from the problem statement, and the fourth step is simplification. The heart of this calculation is equivalent to multiplying edge weights in the tree diagram!

Here is a second example. By multiplying edge weights in the tree diagram, we conclude that the probability of outcome WLL is

$$\frac{1}{2}\cdot\frac{1}{3}\cdot\frac{2}{3}=\frac{1}{9}.$$

We can formally justify this with the Product Rule as follows:

$$\begin{aligned}
\Pr\{WLL\} &= \Pr\{\text{win 1st} \cap \text{lose 2nd} \cap \text{lose 3rd}\}\\
&= \underbrace{\Pr\{\text{win 1st}\}\cdot\Pr\{\text{lose 2nd} \mid \text{win 1st}\}\Pr\{\text{lose 3nd} \mid \text{win 1st} \ \cap \ \text{lose 2nd}\}}_{\substack{\text{product of edge weights on}\\ \text{root-to-leaf path}}}\\
&= \frac{1}{2}\cdot\frac{1}{3}\cdot\frac{2}{3}\\
&= \frac{1}{9}
\end{aligned}$$

Step 4: Compute Event Probabilities

We can now compute the probability that The Halting Problem wins the tournament given that they win the first game:

$$\begin{aligned}
\Pr\{A \mid B\} &= \frac{\Pr\{A\cap B\}}{\Pr\{B\}} && \text{(Product Rule)}\\
&= \frac{1/3+1/18}{1/3+1/18+1/9} && \text{(Sum Rule for } \Pr\{B\}\text{)}\\
&= \frac{7}{9}.
\end{aligned}$$

The Halting Problem has a $7/9$ chance of winning the tournament, given that they win the first game.

7.2.2 An *a posteriori* Probability

In the preceding example, we wanted the probability of an event A, given an *earlier* event B. In particular, we wanted the probability that The Halting Problem won the series, given that they won the first game. It can be harder to think about the probability of an event A, given a *later* event B. For example, what is the probability that The Halting Problem wins its first game, given that the team wins the series? This is called an *a posteriori* probability.

An *a posteriori* probability question can be interpreted in two ways. By one interpretation, we reason that since we are given the series outcome, the first game is already either won or lost; we do not know which. The issue of who won the first game is a question of fact, not a question of probability. Though this interpretation may have philosophical merit, we will never use it.

We will always prefer a second interpretation. Namely, we suppose that the experiment is run over and over and ask in what fraction of the experiments did event A occur when event B occurred?

For example, if we run many hockey series, in what fraction of the series did the Halting Problem win the first game when they won the whole series? Under this interpretation, whether A precedes B in time is irrelevant. In fact, we will solve *a posteriori* problems exactly the same way as other conditional probability problems. The only trick is to avoid being confused by the wording of the problem!

We can now compute the probability that The Halting Problem wins its first game, given that the team wins the series. The sample space is unchanged; see Figure 8. As before, let A be the event that The Halting Problem wins the series, and let B be the event that they win the first game. We already computed the probability of each outcome; all that remains is to compute the probability of event $\Pr\{B \mid A\}$:

$$\begin{aligned}\Pr\{B \mid A\} &= \frac{\Pr\{B \cap A\}}{\Pr\{A\}} \\ &= \frac{1/3 + 1/18}{1/3 + 1/18 + 1/9} \\ &= \frac{7}{9}\end{aligned}$$

The probability of The Halting Problem winning the first game, given that they won the series is $7/9$.

This answer is suspicious! In the preceding section, we showed that $\Pr\{A \mid B\} = 7/9$. Could it be true that $\Pr\{A \mid B\} = \Pr\{B \mid A\}$ in general? We can determine the conditions under which this equality holds by writing $\Pr\{A \cap B\} = \Pr\{B \cap A\}$ in two different ways as follows:

$$\Pr\{A \mid B\}\Pr\{B\} = \Pr\{A \cap B\} = \Pr\{B \cap A\} = \Pr\{B \mid A\}\Pr\{A\}.$$

Evidently, $\Pr\{A \mid B\} = \Pr\{B \mid A\}$ only when $\Pr\{A\} = \Pr\{B\} \neq 0$. This is true for the hockey problem, but only by coincidence. In general, $\Pr\{A \mid B\}$ and $\Pr\{B \mid A\}$ are *not* equal!

7.2.3 A Problem with Two Coins [Optional]

[Optional]

We have two coins. One coin is fair; that is, comes up heads with probability $1/2$ and tails with probability $1/2$. The other is a trick coin; it has heads on both sides, and so *always* comes up heads. Now suppose we randomly choose one of the coins, without knowing one we're picking and with each coin equally likely. If we flip this coin and get heads, then what is the probability that we flipped the fair coin?

This is one of those tricky *a posteriori* problems, since we want the probability of an event (the fair coin was chosen) given the outcome of a later event (heads came up). Intuition may fail us, but the standard four-step method works perfectly well.

Step 1: Find the Sample Space

The sample space is worked out with the tree diagram in Figure 9.

Step 2: Define Events of Interest

Let A be the event that the fair coin was chosen. Let B the event that the result of the flip was heads. The outcomes in each event are marked in the figure. We want to compute $\Pr\{A \mid B\}$, the probability that the fair coin was chosen, given that the result of the flip was heads.

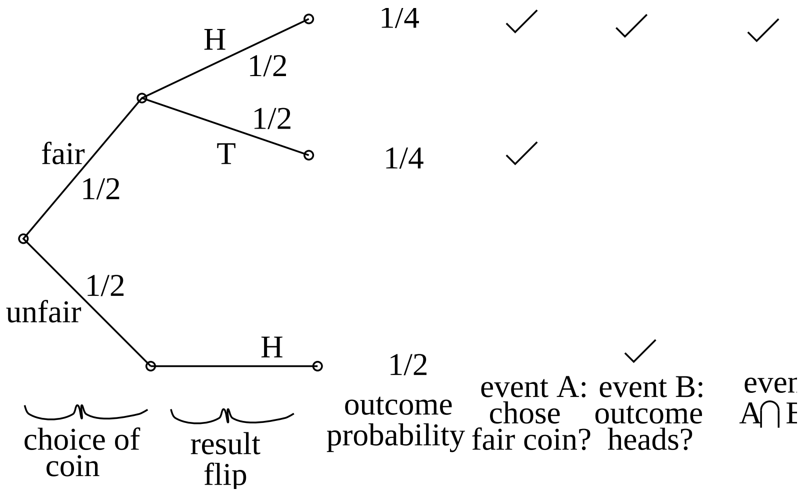


Figure 9: What is the probability that we flipped the fair coin, given that the result was heads?

Step 3: Compute Outcome Probabilities

First, we assign probabilities to edges in the tree diagram. Each coin is chosen with probability $1/2$. If we choose the fair coin, then head and tails each come up with probability $1/2$. If we choose the trick coin, then heads comes up with probability 1. By the Product Rule, the probability of an outcome is the product of the probabilities on the corresponding root-to-leaf path. All of these probabilities are shown in Figure 9.

Step 4: Compute Event Probabilities

$$\begin{aligned}
\Pr\{A \mid B\} &= \frac{\Pr\{A \cap B\}}{\Pr\{B\}} && \text{(Product Rule)} \\
&= \frac{1/4}{1/4 + 1/2} && \text{(Sum Rule for } \Pr\{B\}\text{)} \\
&= \frac{1}{3}
\end{aligned}$$

So the probability that the fair coin was chosen, given that the result of the flip was heads, is $1/3$.

7.2.4 A Variant of the Two Coins Problem [Optional]

[Optional] Here is a variant of the two coins problem. Someone hands us either the fair coin or the trick coin, but we do not know which. We flip the coin 100 times and see heads every time. What can we say about the probability that we flipped the fair coin? Remarkably, nothing! That's because we have no idea with what probability, if any, the fair coin was chosen.

In fact, maybe we were intentionally handed the fair coin. If we try to capture this fact with a probability model, we would have to say that the probability that we have the fair coin is one. Then the conditional probability that we have the fair coin given that we flipped 100 heads remains one, because we do have it.

A similar problem arises in polls around election time. A pollster picks a random American and ask his or her party affiliation. Suppose he repeats this experiment several hundred times and 60% of respondents say that they are Democrats. What can be said about the probability that a majority of Americans are Democrats? Nothing!

To make the analogy clear, suppose the country contains only two people. There is either one Democrat and one Republican (like the fair coin), or there are two Democrats (like the trick coin). The pollster picks a random citizen 100 times; this is analogous to flipping the coin 100 times. Even if he always picks a Democrat (flips heads), he can not determine the probability that the country is all Democrat!

Of course, if we have the fair coin, it is very unlikely that we would flip 100 heads. So in practice, if we got 100 heads, we would bet *with confidence* that we did not have the fair coin. This distinction between the probability of an event—which may be undefined—and the confidence we may have in its occurrence is central to statistical reasoning about real data. We'll return to this important issue in the coming weeks.

7.2.5 Medical Testing

There is a degenerative disease called Zostritis that 10% of men in a certain population may suffer in old age. However, if treatments are started before symptoms appear, the degenerative effects can largely be controlled.

Fortunately, there is a test that can detect latent Zostritis before any degenerative symptoms appear. The test is not perfect, however:

- If a man has latent Zostritis, there is a 10% chance that the test will say he does not. (These are called "false negatives".)
- If a man does not have latent Zostritis, there is a 30% chance that the test will say he does. (These are "false positives".)

A random man is tested for latent Zostritis. If the test is positive, then what is the probability that the man has latent Zostritis?

Step 1: Find the Sample Space

The sample space is found with a tree diagram in Figure 10.

Step 2: Define Events of Interest

Let A be the event that the man has Zostritis. Let B be the event that the test was positive. The outcomes in each event are marked in Figure 10. We want to find $\Pr\{A \mid B\}$, the probability that a man has Zostritis, given that the test was positive.

Step 3: Find Outcome Probabilities

First, we assign probabilities to edges. These probabilities are drawn directly from the problem statement. By the Product Rule, the probability of an outcome is the product of the probabilities on the corresponding root-to-leaf path. All probabilities are shown in the figure.

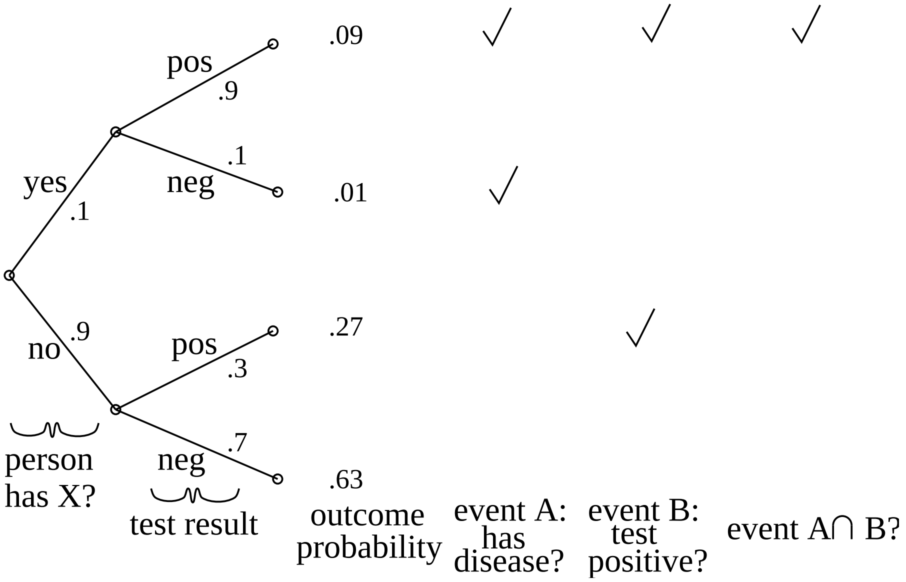


Figure 10: What is the probability that a man has Zostritis, given that the test is positive?

Step 4: Compute Event Probabilities

$$\begin{aligned}\Pr\{A \mid B\} &= \frac{\Pr\{A \cap B\}}{\Pr\{B\}} \\ &= \frac{0.09}{0.09 + 0.27} \\ &= \frac{1}{4}\end{aligned}$$

If a man tests positive, then there is only a 25% chance that he has Zostritis!

This answer is initially surprising, but makes sense on reflection. There are two ways a man could test positive. First, he could be sick and the test correct. Second, could be healthy and the test incorrect. The problem is that most men (90%) are healthy; therefore, most of the positive results arise from incorrect tests of healthy people!

We can also compute the probability that the test is correct for a random man. This event consists of two outcomes. The man could be sick and the test positive (probability 0.09), or the man could be healthy and the test negative (probability 0.63). Therefore, the test is correct with probability $0.09 + 0.63 = 0.72$. This is a relief; the test is correct almost 75% of the time.

But wait! There is a simple way to make the test correct *90%* of the time: always return a negative result! This "test" gives the right answer for all healthy people and the wrong answer only for the 10% that actually have the disease. The best strategy is to completely ignore the test result![5]

[5] In real medical tests, one usually looks at some underlying measurement (e.g., temperature) and uses it to decide whether someone has the disease or not. "Unusual" measurements lead to a conclusion that the disease is present. But just how unusual a measurement should lead to such a conclusion? If we are conservative, and declare the disease present when things are even slightly unusual, we will have a lot of false positives. If we are relaxed, and declare the disease present only when the measurement is very unusual, then we will have a lot of false negatives. So by

There is a similar paradox in weather forecasting. During winter, almost all days in Boston are wet and overcast. Predicting miserable weather every day may be more accurate than really trying to get it right! This phenomenon is the source of many paradoxes; we will see more in coming weeks.

7.3 Confusion about Monty Hall

Using conditional probability we can examine the main argument that confuses people about the Monty Hall example of Section 3.

Let the doors be numbered $1, 2, 3$, and suppose the contestant chooses door 1 and then Carol opens door 2. Now the contestant has to decide whether to stick with door 1 or switch to door 3. To do this, he considers the probability that the prize is behind the remaining unopened door 3, given that he has learned that it is not behind door 2.

To calculate this conditional probability, let W be the event that the contestant chooses door 1, and let R_i be the event that the prize is behind door i, for $i = 1, 2, 3$. The contestant knows that $\Pr\{W\} = 1/3 = \Pr\{R_i\}$, and since his choice has no efffect on the location of the prize, he can say that

$$\Pr\{R_i \cap W\} = \Pr\{R_i\} \cdot \Pr\{W\} = \frac{1}{3} \cdot \frac{1}{3} = \frac{1}{9}$$

and likewise,

$$\Pr\left\{\overline{R_i} \cap W\right\} = (2/3)(1/3) = 2/9,$$

for $i = 1, 2, 3$.

Now the probability that the prize is behind the remaining unopened door 3, given that the contestant has learned that it is not behind door 2 is $\Pr\left\{R_3 \cap W \mid \overline{R_2} \cap W\right\}$. But

$$\Pr\left\{R_3 \cap W \mid \overline{R_2} \cap W\right\} ::= \frac{\Pr\left\{R_3 \cap \overline{R_2} \cap W\right\}}{\Pr\left\{\overline{R_2} \cap W\right\}} = \frac{\Pr\{R_3\} \cap W}{\Pr\left\{\overline{R_2} \cap W\right\}} = \frac{1/9}{2/9} = \frac{1}{2}.$$

Likewise, $\Pr\left\{R_1 \cap W \mid \overline{R_2} \cap W\right\} = 1/2$. So the contestant concludes that the prize is equally likely to be behind door 1 as behind door 3, and therefore there is no advantage to the switch strategy over the stick strategy. But this contradicts our earlier analysis!

Whew, that is confusing! Where did the contestant's reasoning go wrong? (Maybe, like some Ph.D. mathematicians, you are convinced by the contestant's reasoning and now think we must have made a mistake in our earlier conclusion that switching is twice as likely to win than sticking.) Let's try to sort this out.

There is a fallacy in the contestant's reasoning—a subtle one. In fact, his calculation that, given that the prize is not behind door 2, that it's equally likely to be behind door 1 as door 3 is *correct*. His mistake is in not realizing that he knows *more* than that the prize is not behind door 2. He has confused two similar, but distinct, events, namely,

shifting the decision threshold, one can trade off on false positives versus false negatives. It appears that the tester in our example above did not choose the right threshold for their test—they can probably get higher overall accuracy by allowing a few more false negatives to get fewer false positives.

1. the contestant chooses door 1 and the prize is not behind door 2, and,
2. the contestant chooses door 1 and then Carol opens door 2..

These are different events and indeed they have different probabilities. The fact that Carol opens door 2 tells the contestant *more* than that the prize is not behind door 2.

We can precisely demonstrate this with our sample space of triples (i, j, k), where the prize is behind door i, the contestant picks door j, and Carol opens door k. In particular, let C_i be the event that Carol opens door i. Then, event 1. is $\overline{R_2} \cap W$, and event 2. is $W \cap C_2$.

We can confirm the correctness of the contestant's calculation that the prize is behind door 1 given event 1:

$$\begin{aligned}
\overline{R_2} \cap W &::= \{(1,1,2),(3,1,2),(1,1,3)\} \\
\Pr\left\{\overline{R_2} \cap W\right\} &= = \frac{1}{18} + \frac{1}{9} + \frac{1}{18} = \frac{2}{9} \\
\Pr\left\{R_1 \mid \overline{R_2} \cap W\right\} &= \frac{\Pr\{\{(1,1,2),(1,1,3)\}\}}{2/9} = \frac{1}{2}.
\end{aligned}$$

But although the contestant's calculation is correct, his blunder is that he calculated the wrong thing. Specifically, he conditioned his conclusion on the wrong event. The contestant's situation when he must decide to stick or switch is that event 2. has occurred. So he should have calculated:

$$\begin{aligned}
W \cap C_2 &::= \{(1,1,2),(3,1,2)\} \\
\Pr\{W \cap C_2\} &= \frac{1}{18} + \frac{1}{9} = \frac{1}{6} \\
\Pr\{R_1 \mid W \cap C_2\} &= \frac{\Pr\{(1,1,2)\}}{1/6} = \frac{1}{3}.
\end{aligned}$$

In other words, the probability that the prize is behind his chosen door 1 is 1/3, so he should switch because the probability is 2/3 that the prize is behind the other door 3, exactly as we correctly concluded in Section 3.

Once again, we see that mistaken intuition gets resolved by falling back on an examination of outcomes in the probability space.

8 Case Analysis

Combining the sum and product rules provides a natural way to determine the probabilities of complex events via case analysis. As a motivating example, we consider a rather paradoxical true story.

8.1 Discrimination Lawsuit

Several years ago there was a sex discrimination lawsuit against Berkeley. A female professor was denied tenure, allegedly because she was a woman. She argued that in every one of Berkeley's 22 departments, the percentage of male applicants accepted was greater than the percentage of female applicants accepted. This sounds very suspicious, if not paradoxical!

However, Berkeley's lawyers argued that across the whole university the percentage of male applicants accepted was actually *lower* than the percentage of female applicants accepted! This suggests that if there was any sex discrimination, then it was against men! Must one party in the dispute be lying?

8.1.1 A false analysis

Here is a fallacious analysis of the discrimination lawsuit.

To clarify the arguments, let's and express them in terms of conditional probabilities. Suppose that there are only two departments, EE and CS, and consider the experiment where we ignore gender and pick an applicant at random. Define the following events:

- Let A be the event that the applicant is accepted.
- Let F_{EE} the event that the applicant is a female applying to EE.
- Let F_{CS} the event that the applicant is a female applying to CS.
- Let M_{EE} the event that the applicant is a male applying to EE.
- Let M_{CS} the event that the applicant is a male applying to CS.

Assume that all applicants are either male or female, and that no applicant applied to both departments. That is, the events F_{EE}, F_{CS}, M_{EE}, and M_{CS} are all disjoint.

The female plaintiff makes the following argument:

$$\Pr\{A \mid F_{EE}\} < \Pr\{A \mid M_{EE}\} \tag{7}$$

$$\Pr\{A \mid F_{CS}\} < \Pr\{A \mid M_{CS}\} \tag{8}$$

That is, in both departments, the probability that a woman is accepted is less than the probability that a man is accepted. The university retorts that overall a woman applicant is *more* likely to be accepted than a man:

$$\Pr\{A \mid F_{EE} \cup F_{CS}\} > \Pr\{A \mid M_{EE} \cup M_{CS}\} \tag{9}$$

It is easy to believe that these two positions are contradictory.

[Optional] In fact, we might even try to prove this as follows:

$$\Pr\{A \mid F_{EE}\} + \Pr\{A \mid F_{CS}\} < \Pr\{A \mid M_{EE}\} + \Pr\{A \mid M_{CS}\} \qquad \text{(by (7) \& (8)).} \tag{10}$$

Therefore

$$\Pr\{A \mid F_{EE} \cup F_{CS}\} < \Pr\{A \mid M_{EE} \cup M_{CS}\}, \tag{11}$$

which exactly contradicts the university's position!

However, there is a problem with this argument; equation (11) follows (10) only if we accept False Claim 7.4 above! Therefore, this argument is invalid.

In fact, the table below shows a set of application statistics for which the assertions of both the plaintiff and the university hold:

CS	0 females accepted, 1 applied	0%
	50 males accepted, 100 applied	50%
EE	70 females accepted, 100 applied	70%
	1 male accepted, 1 applied	100%
Overall	70 females accepted, 101 applied	$\approx$ 70%
	51 males accepted, 101 applied	$\approx$ 51%

In this case, a higher percentage of males were accepted in both departments, but overall a higher percentage of females were accepted! Bizarre!

Let's think about the reason that this example is counterintuitive. Our intuition tells us that we should be able to analyze an applicant's overall chance of acceptance through case analysis. A female's overall chance of acceptance should be some sort of average of her chance of acceptance within each department, and similarly for males. Since the female's chance in each department is smaller, her overall average chance ought to be smaller as well. What is going on?

A correct analysis of the Discrimination Lawsuit problem rests on a proper rule for doing case analysis. This rule is called the *Law of Total Probability*.

8.2 The Law of Total Probability

Theorem 8.1 (Total Probability). *If a sample space is the disjoint union of events $B_0, B_1, \dots$, then for all events A,*

$$\Pr\{A\} = \sum_{i \in \mathbb{N}} \Pr\{A \cap B_i\}.$$

Theorem 8.1 follows immediately from the Sum Rule, because A is the disjoint union of $A \cap B_0$, $A \cap B_1, \dots$.

A more traditional form of this theorem uses conditional probability.

Corollary 8.2 (Total Probability). *If a sample space is the disjoint union of events $B_0, B_1, \dots$, then for all events A,*

$$\Pr\{A\} = \sum_{i \in \mathbb{N}} \Pr\{A \mid B_i\} \Pr\{B_i\}.$$

Example 8.3. The probability a student comes to class is $1/2$ in rainy weather, but $1/10$ in sunny weather. If the probability that it rains is $1/5$, what is the probability the student comes to class?

We can answer this question using the law of Total Probability. If we let C be the event that the student comes to class, and R the event that it rains, then we have

$$\begin{aligned} \Pr\{C\} &= \Pr\{C \mid R\} \Pr\{R\} + \Pr\{C \mid \overline{R}\} \Pr\{\overline{R}\} \\ &= (1/2) \cdot (1/5) + (1/10) \cdot (4/5) \\ &= 6/50 \end{aligned}$$

8.3 Resolving the Discrimination Lawsuit Paradox

With the law of total probability in hand, we can perform a proper case analysis for our discrimination lawsuit.

Let F_A be the event that a female applicant is accepted.

Assume that no applicant applied to both departments. That is, the events, F_{EE}, that the female applicant is applying to EE, and F_{CS}, that she is applying to CS, are disjoint (and in fact complementary).

Since F_{EE} and F_{CS} partition the sample space, we can apply the law of total probability to analyze acceptance probability:

$$\begin{aligned} \Pr\{F_A\} &= \Pr\{F_A \mid F_{EE}\}\Pr\{F_{EE}\} + \Pr\{F_A \mid F_{CS}\}\Pr\{F_{CS}\} \\ &= (70/100)\cdot(100/101) + (0/1)\cdot(1/101) = 70/101, \end{aligned}$$

which is the correct answer. Notice that as we intuited, $\Pr\{F_A\}$ is a *weighted average* of the conditional probabilities of F_A, where the weights (of 100/101 and 1/101 respectively) are simply the probabilities of being in each condition.

In the same fashion, we can define the events M_A and evaluate a male's overall acceptance probability:

$$\begin{aligned} \Pr\{M_A\} &= \Pr\{M_A \mid M_{EE}\}\Pr\{M_{EE}\} + \Pr\{M_A \mid M_{CS}\}\Pr\{M_{CS}\} \\ &= (1/1)\cdot(1/101) + (50/100)\cdot(100/101) = 51/101, \end{aligned}$$

which is the correct answer. As before, the overall acceptance probability is a weighted average of the conditional acceptance probabilities.

But here we have the source of our paradox: the weights of the weighted averages for males and females are *different*. For the females, the bulk of the weight (common department) falls on the condition (department) in which females do very well (EE); thus the weighted average for females is quite good. For the males, the bulk of the weight falls on the condition in which males do poorly (CS); thus the weighted average for males is poor.

Which brings us back to the allegation in the lawsuit. Having precisely analyzed the arguments of the plaintiff and the defendent, you are in a position to judge how persuasive they are. If you were on the jury, would you find Berkeley guilty of gender bias in its admissions?

8.4 On-Time Airlines

[Optional]

Here is a second example of the same paradox. Newspapers publish on-time statistics for airlines to help travelers choose the best carrier. The on-time rate for an airline is defined as follows:

$$\text{Airline on-time rate} = \frac{\text{\#flights less than 15 minutes late}}{\text{\#flights total}}$$

This seems reasonable, but actually can be completely misleading! Here is some on-time data for two airlines in the late 80's.

	Alaska Air			America West		
Airport	#on-time	#flights	%	#on-time	#flights	%
Los Angeles	500	560	89	700	800	87
Phoenix	220	230	95	4900	5300	92
San Diego	210	230	92	400	450	89
San Francisco	500	600	83	320	450	71
Seattle	1900	2200	86	200	260	77
OVERALL	3330	3020	87	6520	7260	90

This is the same paradox as in the Berkeley lawsuit; America West has a better overall on-time percentage, but Alaska Airlines does a better job at every single airport! The problem is that Alaska Airlines flies proportionally more of its flights to bad weather airports like Seattle; whereas America West is based in fair-weather, low-traffic Phoenix!

9 A Dice Game with an Infinite Sample Space

Suppose two players take turns rolling a fair six-sided die, and whoever first rolls a 1 first is the winner. It's pretty clear that the first player has an advantage since he has the first chance to win. How much of an advantage?

The game is simple and so is its analysis. The only part of the story that turns out to require some attention is the formulation of the probability space.

9.1 Probability that the First Player Wins

Let W be the event that the first player wins. We want to find the probability $\Pr\{W\}$. Now the first player can win in two separate ways: he can win on the first roll or he can win on a later roll. Let F be the event that the first player wins on the first roll. We assume the die is fair; that means $\Pr\{F\} = 1/6$.

So suppose the first player does not win on the first roll, that is, event $\overline{F}$ occurs. But now on the second move, the roles of the first and second player are simply the reverse of what they were on the first move. So the probability that the first player now wins is the same as the probability at the start of the game that the second player would win, namely $1 - \Pr\{W\}$. In other words,

$$\Pr\left\{W \mid \overline{F}\right\} = 1 - \Pr\{W\}. \tag{12}$$

So

$$\Pr\{W\} = \Pr\{F\} + \Pr\left\{W \mid \overline{F}\right\}\Pr\left\{\overline{F}\right\} = \frac{1}{6} + (1 - \Pr\{W\})\frac{5}{6}.$$

Solving for $\Pr\{W\}$ yields

$$\Pr\{W\} = \frac{6}{11} \approx 0.545.$$

We have figured out that the first player has about a 4.5% advantage.

9.2 The Possibility of a Tie

Our calculation that $\Pr\{W\} = 6/11$ is correct, but it rests on an important, hidden assumption. We assumed that the second player *does* win if the first player does *not* win. In other words, there will always be a winner. This seems obvious until we realize that there may be a game in which *neither player wins*—the players might roll forever without rolling a 1. Our assumption is wrong!

But a more careful look at the reasoning above reveals that we didn't actually assume that there always is a winner. All we need to justify is the assumption that the *probability* that the second player wins equals one minus the probability that the first player wins. This is equivalent to assuming, not that there will always be a winner, but only that *the probability is 1* that there is a winner.

How can we justify this? Well, the probability of a winner exactly on the nth roll is the probability, $(5/6)^{n-1}$, that there is no winner on the first $n-1$ rolls, times the probability, $1/6$, that then there is a winner on the nth roll. So the probability that there is a winner is

$$\begin{aligned}
\sum_{n=1}^{\infty} \left(\frac{5}{6}\right)^{n-1} \frac{1}{6} &= \frac{1}{6}\sum_{n=1}^{\infty}\left(\frac{5}{6}\right)^{n-1} \\
&= \frac{1}{6}\sum_{n=0}^{\infty}\left(\frac{5}{6}\right)^{n} \\
&= \frac{1}{6}\cdot\frac{1}{1-5/6} = 1,
\end{aligned}$$

as required.

9.3 The Sample Space

Again, the calculation in the previous subsection was correct: the probability that *some* player wins is indeed 1. But we ought to feel a little uneasy about calculating an infinite sum of probabilities without ever having described the probability space. Notice that in all our previous examples this wasn't much of an issue, because all the sample spaces were finite. But in the dice game, there are an infinite number of outcomes because the game can continue for any finite number of rolls.

Following our recipe for modelling experiments, we should first decide on the sample space, namely, what is an outcome of our dice game? Since a game involves a series of dice rolls until a 1 appears, it's natural to include as outcomes the sequences of rolls which determine a winner. Namely, we include as sample points all sequences of integers between 1 and 6 that end with a first occurrence of 1.

For example, the sequences (1), $(5,4,1)$, $(6,6,6,6,1)$ are sample points describing wins by the first player—after 1, 3 and 5 rolls, respectively. Similarly, $(2,1)$ and $(5,4,3,1)$ are outcomes describing wins by the second player. On the other hand, $(3,2,3)$ is not a sample point because no 1 occurs, and $(3,1,2,1)$ is not a sample point because it continues after the first 1.

Now since we assume the die is fair, each number is equally likely to appear, so it's natural to *define* the probability of any winning sample point of length n to be $(1/6)^n$.

The outcomes in the event that there is a winner on the nth roll are the 5^{n-1} length-n sequences whose first 1 occurs in the nth position. Therefore this event has the probability

$$5^{n-1}\left(\frac{1}{6}\right)^{n} = \left(\frac{5}{6}\right)^{n-1}\frac{1}{6}.$$

This is the probability that we used in the previous subsection to calculate that the probability is 1 that there is a winner.

Besides winning sequences, which are necessarily of finite length, we should consider including sample points corresponding to games with no winner. Now since the winning probabilities already total to one, any sample points we choose to reflect no-winner situations must be assigned probability zero, and moreover the event consisting of all the no-winner points that we include must have probability zero.

A natural choice for the no-winner outcomes would be all the *infinite* sequences of integers between 2 and 6, namely, those with no occurrence of a 1. This leads to a legitimate sample space. But for the analysis we just did of the dice game, *it makes absolutely no difference what no-win outcomes we include*. In fact, it doesn't matter whether we include any no-win points at all.

It does seem a little strange to model the game in a way that denies the logical possibility of an infinite sequence of rolls. On the other hand, we have no need to model the details of the infinite sequences of rolls when there is no winner. So let's define our sample space to include a *single* additional outcome which does represent the possibility of the game continuing forever with no winner; the probability of this "no winner" point is defined to be 0. So this choice of sample space acknowledges the logical possibility of an infinite game.[6]

10 Independence

10.1 The Definition

Definition 10.1. Suppose A and B are events, and B has positive probability. Then A is *independent* of B iff

$$\Pr\{A \mid B\} = \Pr\{A\}.$$

In other words, that fact that event B occurs does not affect the probability that event A occurs.

Figure 11 shows an arrangement of events such that A is independent of B. Assume that the probability of an event is proportional to its area in the diagram. In this example, event A occupies the same fraction of event B as of event $\mathcal{S}$, namely $1/2$. Therefore, the probability of event A is $1/2$ and the probability of event A, given event B, is also $1/2$. This implies that A is independent of B.

[6] Representing the no-winner event by a single outcome has the technical advantage that every set of outcomes is an event—which would not be the case if we explicitly included all the infinite sequences without occurrences of a 1 (*cf.*, footnote 2).

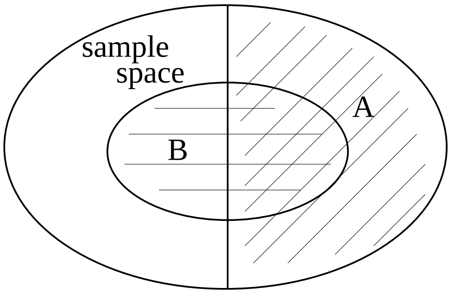


Figure 11: In this diagram, event A is independent of event B.

10.2 An Example with Coins

Suppose we flip two fair coins. Let A be the event that the first coin is heads, and let B be the event that the second coin is heads. Since the coins are fair, we have $\Pr\{A\} = \Pr\{B\} = 1/2$. In fact, the probability that the first coin is heads is still $1/2$, even if we are given that the second coin is heads; the outcome of one toss does not affect the outcome of the other. In symbols, $\Pr\{A \mid B\} = 1/2$. Since $\Pr\{A \mid B\} = \Pr\{A\}$, events A and B are independent.

Now suppose that we glue the coins together, heads to heads. Now each coin still has probability $1/2$ of coming up heads; that is, $\Pr\{A\} = \Pr\{B\} = 1/2$. But if the first coin comes up heads, then the glued on second coin must be tails! That is, $\Pr\{A \mid B\} = 0$. Now, since $\Pr\{A \mid B\} \neq \Pr\{A\}$, the events A and B are not independent.

10.3 The Independent Product Rule

The Definition 10.1 of independence of events A and B does not apply if the probability of B is zero. It's useful to extend the definition to the zero probability case by defining *every* event to be independent of a zero-probability event—even the event itself.

Definition 10.2. If A and B are events and $\Pr\{B\} = 0$, then A is defined to be independent of B.

Now there is an elegant, alternative way to define independence that is used in many texts:

Theorem 10.3. *Events A and B are independent iff*

$$\Pr\{A \cap B\} = \Pr\{A\} \cdot \Pr\{B\}. \qquad \text{(Independent Product Rule)}$$

Proof. If $\Pr\{B\} = 0$, then Theorem 10.3 follows immediately from Definition 10.2, so we may assume that $\Pr\{B\} > 0$. Then

$$\begin{aligned}
A \text{ is independent of } B \text{ iff } \Pr\{A \mid B\} &= \Pr\{A\} && \text{(Definition 10.1)} \\
\text{iff } \frac{\Pr\{A \cap B\}}{\Pr\{B\}} &= \Pr\{A\} && \text{(Definition 7.1)} \\
\text{iff } \Pr\{A \cap B\} &= \Pr\{A\}\Pr\{B\} && \text{(multiplying by } \Pr\{B\} > 0)
\end{aligned}$$

□

The Independent Product Rule is fundamental and worth remembering. In fact, many texts use the Independent Product Rule as the definition of independence.

Notice that because the Rule is symmetric in A and B, it follows immediately that independence is a symmetric relation. For this reason, we do not have to say, "A is independent of B" or vice versa; we can just say "A and B are independent".

10.4 Independence of the Complement

We think of A being independent of B intuitively as meaning that "knowing" whether *or not* B has occurred has no effect on the probability of A. This intuition is supported by an easy, but important property of our formal Definition 10.1 of independence:

Lemma 10.4. *If A is independent of B, then A is independent of $\overline{B}$.*

Proof. If A is independent of B, then

$$\begin{aligned}
\Pr\{A\}\Pr\left\{\overline{B}\right\} &= \Pr\{A\}(1 - \Pr\{B\}) && \text{(Complement Rule)}\\
&= \Pr\{A\} - \Pr\{A\}\Pr\{B\} &&\\
&= \Pr\{A\} - \Pr\{A \cap B\} && \text{(independence)}\\
&= \Pr\{A - B\} && \text{(Difference Rule)}\\
&= \Pr\left\{A \cap \overline{B}\right\} && \text{(Definition of } A - B\text{).}
\end{aligned}$$

That is,

$$\Pr\{A\}\Pr\left\{\overline{B}\right\} = \Pr\left\{A \cap \overline{B}\right\}$$

so A and $\overline{B}$ are independent by Theorem 10.3. □

10.5 Disjoint Events vs. Independent Events

Suppose that events A and B are disjoint, as shown in Figure 12; that is, no outcome is in both events. In the diagram, we see that $\Pr\{A\}$ is non-zero. On the other hand:

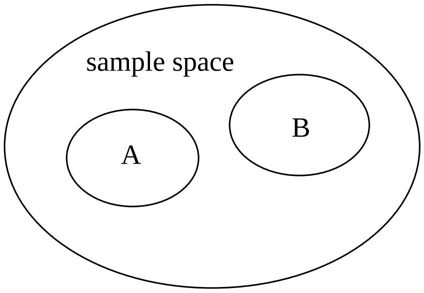


Figure 12: This diagram shows two disjoint events, A and B. Disjoint events are not independent!

$$\Pr\{A \mid B\} = \frac{\Pr\{A \cap B\}}{\Pr\{B\}} = 0.$$

Therefore, $\Pr\{A \mid B\} \neq \Pr\{A\}$, and so event A is not independent of event B. In general, *disjoint* events are not *independent*.

11 Independent Coins and Dice

11.1 An Experiment with Two Coins

Suppose that we flip two independent, fair coins. Let A be the event that the coins match; that is, both are heads or both are tails. Let B the event that the first coin is heads. Are these independent events?

At first, the answer may appear to be "no". After all, whether or not the coins match depends on how the first coin comes up; if we toss HH, then they match, but if we toss TH, then they do not.

The preceding observation is true, but does not imply dependence. Independence is a precise, technical concept, and may hold even if there is a "causal" relationship between two events. In this case, the two events *are* independent, as we prove by the usual procedure.

Claim 11.1. *Events A and B are independent.*

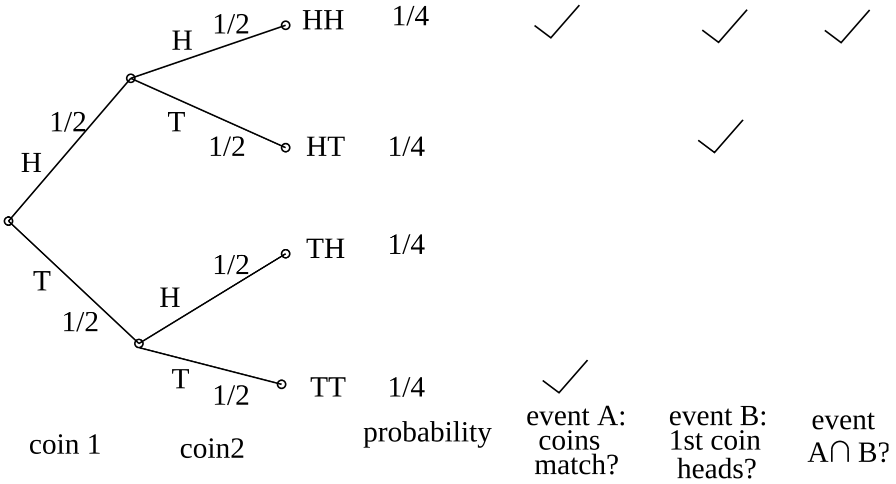


Figure 13: This is a tree diagram for the two coins experiment.

Proof. We must show that $\Pr\{A \mid B\} = \Pr\{A\}$.

Step 1: Find the Sample Space. The tree diagram in Figure 13 shows that there are four outcomes in this experiment, HH, TH, HT, and TT.

Step 2: Define Events of Interest. As previously defined, A is the event that the coins match, and B is the event that the first coin is heads. Outcomes in each event are marked in the tree diagram.

Step 3: Compute Outcome Probabilities. Since the coins are independent and fair, all edge probabilities are $1/2$. We find outcome probabilities by multiplying edge probabilities on each root-to-leaf path. All outcomes have probability $1/4$.

Step 4: Compute Event Probabilities.

$$\Pr\{A \mid B\} = \frac{\Pr\{A \cap B\}}{\Pr\{B\}} = \frac{\Pr\{HH\}}{\Pr\{HH\} + \Pr\{HT\}} = \frac{1/4}{1/4 + 1/4} = \frac{1}{2}$$

$$\Pr\{A\} = \Pr\{HH\} + \Pr\{TT\} = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$$

Therefore, $\Pr\{A \mid B\} = \Pr\{A\}$, and so A and B are independent events as claimed. □

11.2 A Variation of the Two-Coin Experiment

Now suppose that we alter the preceding experiment so that the coins are independent, but not fair. That is each coin is heads with probability p and tails with probability $1 - p$. Again, let A be the event that the coins match, and let B the event that the first coin is heads. Are events A and B independent for all values of p?

The problem is worked out with a tree diagram in Figure 14. The sample space and events are the same as before, so we will not repeat steps 1 and 2 of the probability calculation.

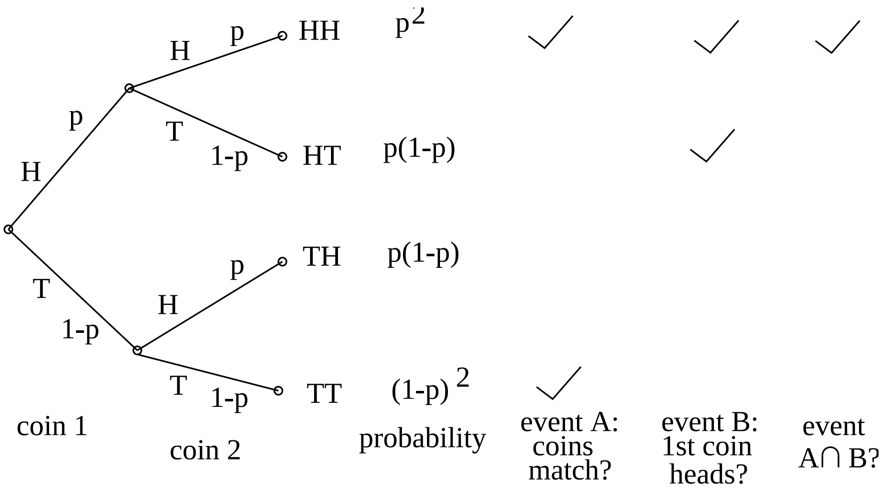


Figure 14: This is a tree diagram for a variant of the two coins experiment. The coins are still independent, but no longer necessarily fair.

Step 3: Compute Outcome Probabilities. Since the coins are independent, all edge probabilities are p or $1 - p$. Outcome probabilities are products of edge probabilities on root-to-leaf paths, as shown in Figure 14.

Step 4: Compute Event Probabilities. We want to determine whether $\Pr\{A \mid B\} = \Pr\{A\}$.

$$\Pr\{A \mid B\} = \frac{\Pr\{A \cap B\}}{\Pr\{B\}} = \frac{\Pr\{HH\}}{\Pr\{HH\} + \Pr\{HT\}} = \frac{p^2}{p^2 + p(1-p)} = p$$
$$\Pr\{A\} = \Pr\{HH\} + \Pr\{TT\} = p^2 + (1-p)^2 = 1 - 2p + 2p^2$$

Events A and B are independent only if these two probabilities are equal:

$$\begin{aligned}
\Pr\{A \mid B\} &= \Pr\{A\} \\
\Leftrightarrow \quad p &= 1 - 2p + 2p^2 \\
\Leftrightarrow \quad 0 &= 1 - 3p + 2p^2 \\
\Leftrightarrow \quad 0 &= (1-2p)(1-p) \\
\Leftrightarrow \quad p &= \frac{1}{2}, 1
\end{aligned}$$

The two events are independent only if the coins are fair or if both always come up heads. Evidently, there was some dependence lurking in the previous problem, but it was cleverly hidden by the unbiased coins!

11.3 Independence of Dice Events [Optional]

[Optional]

Suppose we throw two fair dice. Is the event that the sum is equal to a particular value independent of the event that the first throw yields a particular value? More specifically, let A be the event that the first die turns up 3 and B the event that the sum is 6. Are the two events independent?

No, because

$$\Pr\{B \mid A\} = \frac{\Pr\{B \cap A\}}{\Pr\{A\}} = \frac{1/36}{1/6} = \frac{1}{6},$$

whereas $\Pr\{B\} = 5/36$.

On the other hand, let A be the event that the first die turns up 3 and B the event that the sum is 7. Then

$$\Pr\{B \mid A\} = \frac{\Pr\{B \cap A\}}{\Pr\{A\}} = \frac{1/36}{1/6} = \frac{1}{6},$$

whereas $\Pr\{B\} = 6/36$. So in this case, the two events are independent.

Can you explain the difference between these two results?

12 Mutual Independence

We have defined what it means for two events to be independent. But how can we talk about independence when there are more than two events?

12.1 Example: Blood Evidence

During the O. J. Simpson trial a few years ago, a probability problem involving independence came up. A prosecution witness claimed that only one in 200 Americans has the blood type found at the crime scene. The witness then presented facts something like the following:

- $\frac{1}{10}$ of people have type O blood.
- $\frac{1}{5}$ of people have a positive Rh factor.
- $\frac{1}{4}$ of people have another special marker.

The one in 200 figure came from multiplying these three fractions. Was the witness reasoning correctly?

The answer depends on whether or not the three blood characteristics are independent. This might not be true; maybe most people with O^+ blood have the special marker. When the math-competent defense lawyer asked the witness whether these characteristics were independent, he could not say. He could not justify his claim.

12.2 Definition of Mutual Independence

What sort of independence is needed to justify multiplying probabilities of more than two events? The notion we need is called *mutual independence*.

Definition 12.1. Events $A_1, A_2, \ldots, A_n$ are *mutually independent* if for all i such that $1 \leq i \leq n$ and for all $J \subseteq \{1, \ldots, n\} - \{i\}$, we have:

$$\Pr\left\{ A_i \;\middle|\; \bigcap_{j \in J} A_j \right\} = \Pr\{A_i\}.$$

In other words, a collection of events is mutually independent if each event is independent of the intersection of every subset of the others. An equivalent way to formulate mutual independence is give in the next Lemma, though we will skip the proof. Some texts use this formulation as the definition.

Lemma 12.2. *Events $A_1, A_2, \ldots, A_n$ are* mutually independent *iff for all $J \subseteq \{1, \ldots, n\}$, we have:*

$$\Pr\left\{ \bigcap_{j \in J} A_j \right\} = \prod_{j \in J} \Pr\{A_j\}.$$

For example, for $n = 3$, Lemma 12.2 says that

Corollary. *Events A_1, A_2, A_3 are mutually independent iff all of the following hold:*

$$\begin{aligned}
\Pr\{A_1 \cap A_2\} &= \Pr\{A_1\} \cdot \Pr\{A_2\} \\
\Pr\{A_1 \cap A_3\} &= \Pr\{A_1\} \cdot \Pr\{A_3\} \\
\Pr\{A_2 \cap A_3\} &= \Pr\{A_2\} \cdot \Pr\{A_3\} \\
\Pr\{A_1 \cap A_2 \cap A_3\} &= \Pr\{A_1\} \cdot \Pr\{A_2\} \cdot \Pr\{A_3\} \qquad (13)
\end{aligned}$$

Note that A is independent of B *iff* it is independent of $\overline{B}$. This follows immediately from Lemma 10.4 and the fact that $\overline{\overline{B}} = B$. This result also generalizes to many events and provides yet a third equivalent formulation of mutual independence. Again, we skip the proof:

Theorem 12.3. *For any event, A, let $A^{(1)} ::= A$ and $A^{(-1)} ::= \overline{A}$. Then events $A_1, A_2, \ldots, A_n$ are mutually independent iff*

$$\prod_{i=1}^{n} \Pr\left\{A_i^{(x_i)}\right\} = \Pr\left\{ \bigcap_{i=1}^{n} A_i^{(x_i)} \right\} \qquad (14)$$

for all $x_i \in \{1, -1\}$ where $1 \leq i \leq n$.

12.3 Carnival Dice Revisited

We have already considered the gambling game of Carnival Dice in Section 6.1. Now, using independence we can more easily work out the probability that the player wins by calculating the probability of its *complement*.

Namely, let A_i be the event that the ith die matches the player's guess. So $A_1 \cup A_2 \cup A_3$ is the event that the player wins. But

$$\Pr\{A_1 \cup A_2 \cup A_3\} = 1 - \Pr\left\{\overline{A_1 \cup A_2 \cup A_3}\right\} = 1 - \Pr\left\{\overline{A_1} \cap \overline{A_2} \cap \overline{A_3}\right\}.$$

Now, since the dice are independent, Theorem 12.3 implies

$$\Pr\left\{\overline{A_1} \cap \overline{A_2} \cap \overline{A_3}\right\} = \Pr\left\{\overline{A_1}\right\}\Pr\left\{\overline{A_2}\right\}\Pr\left\{\overline{A_3}\right\} = (5/6)^3.$$

Therefore

$$\Pr\{A_1 \cup A_2 \cup A_3\} = 1 - (5/6)^3 = \frac{91}{216}.$$

This is the same value we computed previously using Inclusion-Exclusion. But with independent events, the approach of calculating the complement is often easier than using Inclusion-Exclusion. Note that this example generalizes nicely to a larger number of dice—with 6 dice the probability of a match is $1 - (5/6)^6 \approx 67\%$, with 12 dice it is $1 - (5/6)^{12} \approx 89\%$. Using Inclusion-Exclusion in these cases would have been messy.

12.4 Circuit Failure Revisited

Let's reconsider the circuit problem from section 5.2, where a circuit containing n connections is to be wired up and A_i is the event that the ith connection is made correctly. Again, we want to know the probability that the entire circuit is wired correctly, but this time when we know that all the events A_i are *mutually independent*.

If $p ::= \Pr\left\{\overline{A_i}\right\}$ is the probability that the ith connection is made *incorrectly*, then because the event are independent, we can conclude that the probability that the circuit is correct is $\prod_1^n \Pr\{A_i\} = (1-p)^n$. For $n = 10$, and $p = 0.01$ as in section 5.2, this comes out to around 90.4%—very close to the lower bound. That's because the lower bound is achieved when at most one error occurs at a time, which is nearly true in this case of independent errors, because the chance of more than one error is relatively small (less than 1%).

12.5 A Red Sox Streak [Optional]

[Optional]

The Boston Red Sox baseball team has lost 14 consecutive playoff games. What are the odds of such a miserable streak?

Suppose that we assume that the Sox have a $1/2$ chance of winning each game and that the game results are mutually independent. Then we can compute the probability of losing 14 straight games as follows. Let L_i be the event that the Sox lose the ith game. This gives:

$$\begin{aligned}
\Pr\{L_1 \cap L_2 \cap \cdots \cap L_{14}\} &= \Pr\{L_1\}\Pr\{L_2\}\cdots\Pr\{L_{14}\} \\
&= \left(\frac{1}{2}\right)^{14} \\
&= \frac{1}{16,384}
\end{aligned}$$

The first equation follows from the second definition of mutual independence. The remaining steps use only substitution and simplification.

These are pretty long odds; of course, the probability that the Red Sox lose a playoff game may be greater than $1/2$. Maybe they're cursed.

12.6 An Experiment with Three Coins

This is a tricky problem that always confuses people! Suppose that we flip three fair coins and that the results are mutually independent. Define the following events:

- A_1 is the event that coin 1 matches coin 2
- A_2 is the event that coin 2 matches coin 3
- A_3 is the event that coin 3 matches coin 1

Are these three events mutually independent?

The sample space is easy enough to find that we will dispense with the tree diagram: there are eight outcomes, corresponding to every possible sequence of three flips: $HHH, HHT, HTH, \ldots$. We are interested in events A_1, A_2, and A_3, defined as above. Each outcome has probability $1/8$.

To see if the three events are mutually independent, we must prove a sequence of equalities. It will be helpful first to compute the probability of each event A_i:

$$\begin{aligned}
\Pr\{A_1\} &= \Pr\{HHH\} + \Pr\{HHT\} + \Pr\{TTT\} + \Pr\{TTH\} \\
&= \frac{1}{8} + \frac{1}{8} + \frac{1}{8} + \frac{1}{8} \\
&= \frac{1}{2}
\end{aligned}$$

By symmetry, $\Pr\{A_2\} = \Pr\{A_3\} = 1/2$. Now we can begin checking all the equalities required for mutual independence.

$$\begin{aligned}
\Pr\{A_1 \cap A_2\} &= \Pr\{HHH\} + \Pr\{TTT\} \\
&= \frac{1}{8} + \frac{1}{8} \\
&= \frac{1}{4} \\
&= \frac{1}{2} \cdot \frac{1}{2} \\
&= \Pr\{A_1\}\Pr\{A_2\}
\end{aligned}$$

By symmetry, $\Pr\{A_1 \cap A_3\} = \Pr\{A_1\}\Pr\{A_3\}$ and $\Pr\{A_2 \cap A_3\} = \Pr\{A_2\}\Pr\{A_3\}$ must hold as well. We have now proven that every pair of events is independent. But this is not enough to prove that A_1, A_2, and A_3 are mutually independent! We must check the fourth condition:

$$\begin{aligned}
\Pr\{A_1 \cap A_2 \cap A_3\} &= \Pr\{HHH\} + \Pr\{TTT\} \\
&= \frac{1}{8} + \frac{1}{8} \\
&= \frac{1}{4} \\
&\neq \Pr\{A_1\}\Pr\{A_2\}\Pr\{A_3\} = \frac{1}{8}.
\end{aligned}$$

The three events A_1, A_2, and A_3 are not mutually independent, even though all pairs of events are independent! When proving a set of events independent, remember to check all pairs of events, *and* all sets of three events, four events, etc.

12.7 Pairwise Independence

It's a common situation to have all pairs of events in some collection are independent, but not to know whether three or more of the events are going to be independent. It also turns out to be important enough that a special term has been defined for this situation:

Definition. Events $A_1, A_2, \ldots A_n, \ldots$ are *pairwise independent* if A_i and A_j are independent events for all $i \neq j$.

Note that mutual independence is stronger than pairwise independence. That is, if a set of events is mutually independent, then it must be pairwise independent, but the reverse is not true. For example, the events in the three coin experiment of the preceding subsection were pairwise independent, but not mutually independent.

In the blood example, suppose initially that we know nothing about independence. Then we can only say that the probability that a person has all three blood factors is no greater than the probability that a person has blood type O, which is $1/10$.

If we know that the three blood factors in the O. J. case appear pairwise independently, then we can conclude:

$$\begin{aligned}
\Pr\{\text{person has all 3 factors}\} &\leq \Pr\{\text{person is type } O \text{ and Rh positive}\} \\
&= \Pr\{\text{person is type } O\}\Pr\{\text{person is Rh positive}\} \\
&= \frac{1}{10} \cdot \frac{1}{5} \\
&= \frac{1}{50}
\end{aligned}$$

Knowing that a set of events is pairwise independent is useful! However, if all three factors are mutually independent, then the witness is right; the probability a person has all three factors is $1/200$. Knowing that the three blood characteristics are mutually independent is what justifies the witness's in multiplying the probabilities as in equation (13). The point is that we get progressively tighter upper bounds as we strengthen our assumption about independence.

This example also illustrates an

> **Important Technicality:** To prove a set of three or more events mutually independent, it is *not* sufficient to prove every pair of events independent! In particular, for three events we must also prove that equality (13) also holds.

13 The Birthday Problem

13.1 The Problem

What is the probability that two students among a group of 100 have the same birthday? There are 365 birthdays (month, date) and 100 is less than a third of 365, so an offhand guess might be that the probability is somewhere between 1/3 and 2/3. Another approach might be to think of the setup as having 100 chances of winning a 365-to-1 bet; there is roughly only a 25% chance of winning such a bet. But in fact, the probability that some two among the 100 students have the same birthday is overwhelming: there is less than one chance in thirty million that all 100 students have different birthdays!

As a matter of fact, by the time we have around two dozen students, the chances that two have the same birthday is close to 50%. This seems odd! There are 12 months in the year, yet at a point when we've only collected about two birthdays per month, we have usually already found two students with exactly the same birthday!

There are two assumptions underlying these assertions. First, we assume that all birth dates are equally likely. Second, we assume that birthdays are mutually independent. Neither of these assumptions are really true. Birthdays follow seasonal patterns, so they are not uniformly distributed. Also, birthdays are often related to major events. For example, nine months after a blackout in the 70's there was a sudden increase in the number of births in New England. Since students in the same class are generally the same age, their birthdays are more likely to be dependent on the same major event than the population at large, so they won't be mutually independent. But when there wasn't some unusual event 18 to 22 years ago, student birthdays are close enough to being uniform that we won't be too far off assuming uniformity and independence, so we will stick with these assumptions in the rest of our analysis.

13.2 Solution

There is an intuitive reason why the probability of matching birthdays is so high. The probability that a given pair of students have the same birthday is only $1/365$. This is very small. But with around two dozen students, we have around 365 *pairs* of students, and the probability one of these 365 attempts will result in an event with probability $1/365$ gets to be about 50-50. With 100 students there are about 5000 pairs, and it is nearly certain that an event with probability $1/365$ will occur at least once in 5000 tries.

In general, suppose there are m students and N days in the year. We want to determine the probability that at least two students have the same birthday. Let's try applying our usual method.

Step 1. Find the Sample Space

We can regard an outcome as an m-vector whose components are the birthdays of the m students in order. That is, the sample space is the set of all such vectors:

$$\mathcal{S} ::= \{\langle b_1, b_2, \dots, b_m \rangle \mid b_i \in \{1, 2, \dots, N\} \text{ for } 1 \leq i \leq m\}.$$

There are N^m such vectors.

Step 2: Define Events of Interest

Let A be the event that two or more students have the same birthday. That is,

$$A ::= \{\langle b_1, b_2, \dots, b_m \rangle \mid b_i = b_j \text{ for some } 1 \leq i \neq j \leq m\}.$$

Step 3: Compute Outcome Probabilities

The probability of outcome $\langle b_1, b_2, \dots, b_m \rangle$ is the probability that the first student has birthday b_1, the second student has birthday b_2, *etc.*. The ith person has birthday b_i with probability $1/N$. Assuming birth dates are independent, we can multiply probabilities to get the probability of a particular outcome:

$$\Pr\{\langle b_1, b_2, \dots, b_m \rangle\} = \frac{1}{N^m}.$$

So we have a uniform probability space—the probabilities of all the outcomes are the same.

Step 4: Compute Event Probabilities

The remaining task in the birthday problem is to compute the probability of the event that two or more students have the same birthday. Since the sample space is uniform, we need only count the number of outcomes in the event A. This can be done with Inclusion-Exclusion, but the calculation is involved.

A simpler method is to use the trick of "counting the complement." Let $\overline{A}$ be the complementary event; that is, let $\overline{A} ::= \mathcal{S} - A$. Then, since $\Pr\{A\} = 1 - \Pr\{\overline{A}\}$, we need only determine the probability of event $\overline{A}$.

In the event $\overline{A}$, all students have different birthdays. The event consists of the following outcomes:

$$\{\langle b_1, b_2, \dots, b_m \rangle \mid \text{ all the } b_i\text{'s are distinct}\}$$

In other words, the set $\overline{A}$ consists of all *m-permutations* of the set of N possible birthdays! So now we can compute the probability of $\overline{A}$:

$$\Pr\{\overline{A}\} = \frac{|\overline{A}|}{|\mathcal{S}|} = \frac{|\overline{A}|}{N^m} = \frac{P(N, m)}{N^m} = \frac{N!}{(N-m)!\, N^m},$$

and so

$$\Pr\{A\} = 1 - \frac{N!}{(N-m)!\, N^m},$$

which is a simple formula for the probability that at least two students among a group of m have the same birthday in a year with N days.

Letting $m = 22$ students and $N = 365$ days, we conclude that at least one pair of students have the same birthday with probability ≈ 0.476. If we have $m = 23$ students, then the probability rises to ≈ 0.507. So in a room with 23 students, the odds are in fact better than even that at least two have the same birthday.

13.3 Approximating the Answer to the Birthday Problem

We now know that $\Pr\{A\} = 1 - N!/((N-m)!\, N^m)$, but this formula is hard to work with because it is not a closed form. Evaluating the expression for, say, $N = 365$ and $m = 100$ is a lot of work. It's even harder to determine how big N must be for the probability of a birthday match among $m = 100$ students to equal, say, 90%. We'd also like to understand the growth rate of the probability as a function of m and N.

It turns out that there is a nice asymptotic formula for the probability, namely,

$$\Pr\left\{\overline{A}\right\} \sim e^{-\frac{m^2}{2N}}. \tag{15}$$

as long as $m = o(N^{2/3})$.

This formula actually has an intuitive explanation. The number of ways to pair m students is $\binom{m}{2} \approx m^2/2$. The event that a pair of students has the same birthday has probability $1/N$. Now if these events were mutually independent, then using the approximation $1 - x \approx e^{-x}$, we could essentially arrive at (15) by calculating

$$\begin{aligned}\Pr\left\{\overline{A}\right\} &\approx \left(1 - \frac{1}{N}\right)^{\frac{m^2}{2}} \\ &\approx e^{-\frac{1}{N}\cdot\frac{m^2}{2}} \\ &= e^{-\frac{m^2}{2N}}.\end{aligned}$$

The problem is that the events that pairs of students have distinct birthdays are *not* mutually independent. For example,

$$\Pr\{b_1 = b_3 \mid b_1 = b_2, b_2 = b_3\} = 1 \neq 1/N = \Pr\{b_1 = b_3\}.$$

But notice that if we have a set of *nonoverlapping* pairs of students, then the event that a given pair in the set have the same birthday really is independent of whether the other pairs have the same birthday. That is, we do have mutual independence for any set of nonoverlapping pairs. But if m is small compared to N, then the likelihood will be low that among the pairs with the same birthday, there are two overlapping pairs. In other words, we could expect that for small enough m, the events that pairs have the same birthday are likely to be distributed in the same way as if they were mutually independent, justifying the independence assumption in our simple calculation.

Of course this intuitive argument requires more careful justification. The asymptotic equality (15) can in fact be proved by an algebraic calculation using Stirling's Formula and the Taylor series for $\ln(1-x)$, but we will skip it.

This asymptotic equality also shows why the probability that all students have distinct birthdays drops off rapidly as the number of students grows beyond $\sqrt{N}$ toward $N^{2/3}$. The reason is that the probability (15) decreases in inverse proportion to a quantity obtained by *squaring and then exponentiating* the number of students.

13.4 The Birthday Principle

As a final illustration of the usefulness of the asymptotic equality (15), we determine as a function of N the number of students for which the probability that two have the same birthday is (approximately) $1/2$.

All we need do is set the probability that all birthdays are distinct to $1/2$ and solve for the number of students.

$$\begin{aligned}
e^{-\frac{m^2}{2N}} &\sim \frac{1}{2} \\
e^{\frac{m^2}{2N}} &\sim 2 \\
\frac{m^2}{2N} &\sim \ln 2 \\
m &\sim \sqrt{2N\ln 2} \approx 1.177\sqrt{N}.
\end{aligned}$$

Since the values of m here are $\Theta(\sqrt{N}) = o(N^{2/3})$, the conditions for our asymptotic equality are met and we can expect our approximation to be good.

For example, if $N = 365$, then $1.177\sqrt{N} = 22.49$. This is consistent with out earlier calculation; we found that the probability that at least two students have the same birthday is $1/2$ in a room with around 22 or 23 students. Of course, one has to be careful with the $\sim$ notation; we may end up with an approximation that is only good for very large values. In this case, though, our approximation works well for reasonable values.

The preceding result is called the Birthday Principle. It can be interpreted this way: if you throw about $\sqrt{N}$ balls into N boxes, then there is about a 50% chance that some box gets two balls.

For example, in 27 years there are about 10,000 days. If we put about $1.177\sqrt{10,000} \approx 118$ people under the age of 28 in a room, then there is a 50% chance that at least two were born on exactly the same day of the same year! As another example, suppose we have a roomful of people, and each person writes a random number between 1 and a million on a piece of paper. Even if there are only about $1.177\sqrt{1,000,000} = 1177$ people in the room, there is a 50% chance that two wrote exactly the same number!

Random Variables and Expectation

1 Random Variables

When we perform an experiment, we expect the results to be observable—did the player hit a home run or not?—or measurable—how far did the ball travel? how fast was the pitch? To describe the behavior of such probabilistic experiments with measurable outcomes, we use *random variables*.

For example, consider the experiment of tossing three independent, unbiased coins. We can define C to be the number of heads which appear, and M to be 1 iff all three coins match and 0 otherwise. Any outcome of the coin flips uniquely determines C and M. C can take the values 0,1,2, and 3, and M the values 0 an 1.

We use the notation $[C = 2]$ for the event that there are two heads. Similarly, $[C \geq 2]$ is the event that there are at least two heads, and $[C \in \{1, 3\}]$ is the event that there are an odd number of heads.

Now consider the event that the product of C and M is positive; we write this one as $[C \cdot M > 0]$. Since neither C nor M take negative values, $C \cdot M >$ iff both $C > 0$ and $M > 0$—in other words, there is a head, and all three dice match. So saying $C \cdot M > 0$ is just an obscure way of saying that all three coin flips come up heads. That is, the event $[C \cdot M > 0]$ consists of the single outcome HHH.

When the meaning is clear, we often omit the square brackets denoting events. For example, we say "the event $C = 0$" instead of "the event $[C = 0]$," or $\Pr\{C = 0\}$ instead of $\Pr\{[C = 0]\}$.

Saying that each outcome uniquely determines C and M means that we can think of C and M as functions from outcomes to their values. The natural sample space, $\mathcal{S}$, for this experiment consists of eight outcomes: HHH, HHT, HTH, *etc*. For example, $C(\texttt{HHH}) = 3$, $C(\texttt{HTH}) = 2$, $C(\texttt{TTT}) = 0$. Similarly, $M(\texttt{HHH}) = 1$, $M(\texttt{HTH}) = 0$, $M(\texttt{TTT}) = 1$.

We can formalize the idea of a random variable in general as follows.

Definition 1.1. A *random variable* over a given sample space is a function that maps every outcome to a real number.

Notice that calling a random variable a "variable" a misnomer: it is actually a function.

We will use the random variables C and M as continuing examples. Keep in mind that C ***c***ounts heads and M indicates that all coins ***m***atch.

1.1 Indicator Random Variables

Indicator random variables describe experiments to detect whether or not something happened. The random variable M is an example of an indicator variable, indicating whether or not all three coins match.

Definition 1.2. An *indicator random variable* is a random variable that maps every outcome to either 0 or 1.

Indicator random variables are also called *Bernoulli* or *characteristic* random variables. Typically, indicator random variables identify all outcomes that share some property ("characteristic"): outcomes with the property are mapped to 1, and outcomes without the property are mapped to 0.

1.2 Events Defined by a Random Variable

There is a natural relationship between random variables and events. Recall that an event is just a subset of the outcomes in the sample space of an experiment.

The relationship is simplest for an indicator random variable. An indicator random variable partitions the sample space into two blocks: outcomes mapped to 1 and outcomes mapped to 0. These two sets of outcomes are events. For example, the random variable M partitions the sample space as follows:

$$\underbrace{\texttt{HHH}\quad\texttt{TTT}}_{\text{mapped to 1}}\qquad\underbrace{\texttt{HHT}\quad\texttt{HTH}\quad\texttt{HTT}\quad\texttt{THH}\quad\texttt{THT}\quad\texttt{TTH}}_{\text{mapped to 0}}$$

Thus, the random variable M defines two events, the event $[M = 1]$ that all coins match and the event $[M = 0]$ that not all coins match.

The random variable C partitions the sample space into four blocks:

$$\underbrace{\texttt{TTT}}_{\text{mapped to 0}}\quad\underbrace{\texttt{TTH}\quad\texttt{THT}\quad\texttt{HTT}}_{\text{mapped to 1}}\quad\underbrace{\texttt{THH}\quad\texttt{HTH}\quad\texttt{HHT}}_{\text{mapped to 2}}\quad\underbrace{\texttt{HHH}}_{\text{mapped to 3}}$$

Thus, the random variable C defines the four events $[C = i]$ for $i \in \{0, 1, 2, 3\}$. These are the events that no coin is heads, that one coin is heads, that two coins are heads, and finally that three coins are heads.

A general random variable may partition the sample space into many blocks. A block contains all outcomes mapped to the same value by the random variable.

1.3 Probability of Events Defined by a Random Variable

Recall that the probability of an event is the sum of the probabilities of the outcomes it contains. From this rule, we can compute the probability of various events associated with a random variable. For example, if $R : \mathcal{S} \to \mathbb{R}$ is a random variable and x is a real number, then

$$\Pr\{R = x\} = \sum_{w \in [R=x]} \Pr\{w\}.$$

For example, we can compute $\Pr\{C = 2\}$ as follows:

$$\begin{aligned}
\Pr\{C = 2\} &= \sum_{w \in [C=2]} \Pr\{w\} && \text{(def of } \Pr\{\}\text{)} \\
&= \Pr\{\texttt{THH}\} + \Pr\{\texttt{HTH}\} + \Pr\{\texttt{HHT}\} && \text{(the 3 outcomes in } [C = 2]\text{)} \\
&= \frac{1}{8} + \frac{1}{8} + \frac{1}{8} = \frac{3}{8}.
\end{aligned}$$

Note that each outcome has probability $1/8$, since the three coins are fair and independent.

Similarly, we can compute $\Pr\{M = 1\}$ and $\Pr\{C \geq 2\}$

$$\begin{aligned}
\Pr\{M = 1\} &= \sum_{w \in [M=1]} \Pr\{w\} \\
&= \Pr\{\texttt{HHH}\} + \Pr\{\texttt{TTT}\} \\
&= \frac{1}{8} + \frac{1}{8} = \frac{1}{4}. \\
\Pr\{C \geq 2\} &= \sum_{w \in [C \geq 2]} \Pr\{w\} \\
&= \Pr\{\texttt{THH}\} + \Pr\{\texttt{HTH}\} + \Pr\{\texttt{HHT}\} + \Pr\{\texttt{HHH}\} \\
&= \frac{1}{8} + \frac{1}{8} + \frac{1}{8} + \frac{1}{8} = \frac{1}{2}.
\end{aligned}$$

The justification for each step is the same as before.

It's common in such calculations to group outcomes by their value. For instance, we could also have calculated:

$$\begin{aligned}
\Pr\{C \geq 2\} &= \Pr\{C = 2\} + \Pr\{C = 3\} \\
&= \Pr\{\texttt{THH}, \texttt{HTH}, \texttt{HHT}\} + \Pr\{\texttt{HHH}\} \\
&= \frac{3}{8} + \frac{1}{8} = \frac{1}{2}
\end{aligned}$$

Similarly, we find the probability of the event that $C \in \{1, 3\}$.

$$\begin{aligned}
\Pr\{C \in \{1, 3\}\} &= \Pr\{C = 1\} + \Pr\{C = 3\} \\
&= \Pr\{\texttt{TTH}, \texttt{THT}, \texttt{HTT}\} + \Pr\{\texttt{HHH}\} \\
&= \frac{3}{8} + \frac{1}{8} = \frac{1}{2}.
\end{aligned}$$

In general, for a set $A = \{a_0, a_1, \dots\}$ of real numbers, $\Pr\{R \in A\}$ can also be evaluated by summing over the values in A. That is,

$$\Pr\{R \in A\} = \sum_{a \in A} \Pr\{R = a\}.$$

1.4 Conditional Probability

Mixing conditional probabilities and events involving random variables creates no new difficulties. For example, $\Pr\{C \geq 2 \mid M = 0\}$ is the probability that at least two coins are heads ($C \geq 2$), given that all three coins are not the same ($M = 0$). We can compute this probability using the familiar Product Rule:

$$\begin{aligned}
\Pr\{C \geq 2 \mid M = 0\} &= \frac{\Pr\{C \geq 2 \wedge M = 0\}}{\Pr\{M = 0\}} \\
&= \frac{\Pr\{\{\texttt{THH}, \texttt{HTH}, \texttt{HHT}\}\}}{\Pr\{\{\texttt{THH}, \texttt{HTH}, \texttt{HHT}, \texttt{HTT}, \texttt{THT}, \texttt{TTH}\}\}} \\
&= \frac{3/8}{6/8} = \frac{1}{2}.
\end{aligned}$$

1.5 Independence

1.5.1 Independence for Two Random Variables

Definition 1.3. Two random variables R_1 and R_2 are *independent*[1] if for all $x_1, x_2 \in \mathbb{R}$ such that $\Pr\{R_2 = x_2\} \neq 0$, we have:

$$\Pr\{R_1 = x_1 \mid R_2 = x_2\} = \Pr\{R_1 = x_1\}$$

As with independence of events, we can also formulate independence of two random variables in terms of the conjunction of events:

Definition 1.4. Two random variables R_1 and R_2 are *independent* if for all $x_1, x_2 \in \mathbb{R}$, we have:

$$\Pr\{R_1 = x_1 \wedge R_2 = x_2\} = \Pr\{R_1 = x_1\} \cdot \Pr\{R_2 = x_2\}.$$

Definition 1.3 says that the probability that R_1 has a particular value is unaffected by the value of R_2, reflecting the intuition behind independence. Definition 1.4 has the slight technical advantage that it applies even if $\Pr\{R_2 = x_2\} = 0$. Otherwise, the two definitions are equivalent, and we will use them interchangably.

1.5.2 Proving that Two Random Variables are Not Independent

Are C and M independent? Intuitively, no: the number of heads, C, not only affects, but completely determines whether all three coins match, that is, whether $M = 1$. To verify this, let's use

[1] This definition works for sample spaces $\mathcal{S} = \{w_0, w_1, \ldots\}$ of the kind we consider in 6.042. For more general sample spaces, the definition is that

$$\Pr\{y_1 \leq R_1 \leq x_1 \mid y_2 \leq R_2 \leq x_2\} = \Pr\{y_1 \leq R_1 \leq x_1\}$$

for all $y_1, x_1, y_2, x_2 \in \mathbb{R}$ and $\Pr\{y_2 \leq R_2 \leq x_2\} \neq 0$.

the first definition 1.3 of independence. We must find some $x_1, x_2 \in \mathbb{R}$ such that the condition in the first definition is false. For example, the condition does not hold for $x_1 = 2$ and $x_2 = 1$:

$$\Pr\{C = 2 \wedge M = 1\} = 0 \quad \text{but} \quad \Pr\{C = 2\} \cdot \Pr\{M = 1\} = \frac{3}{8} \cdot \frac{1}{4} \neq 0$$

The first probability is zero because we never have exactly two heads ($C = 2$) when all three coins match ($M = 1$). The other two probabilities were computed earlier.

1.5.3 A Dice Example

Suppose that we roll two fair, independent dice. We can regard the numbers that turn up as random variables, D_1 and D_2. For example, if the outcome is $w = (3, 5)$, then $D_1(w) = 3$ and $D_2(w) = 5$.

Let $T = D_1 + D_2$. Then T is also a random variable, since it is a function mapping each outcome to a real number, namely the sum of the numbers shown on the two dice. For outcome $w = (3, 5)$, we have $T(w) = 3 + 5 = 8$.

Define S as follows:

$$S ::= \begin{cases} 1 & \text{if } T = 7, \\ 0 & \text{if } T \neq 7. \end{cases}$$

That is, $S = 1$ if the sum of the dice is 7, and $S = 0$ if the sum of the dice is not 7. For example, for outcome $w = (3, 5)$, we have $S(w) = 0$, since the sum of the dice is 8. Since S is a function mapping each outcome to a real number, S is also a random variable. In particular, S is an indicator random variable, since every outcome is mapped to 0 or 1.

The definitions of random variables T and S illustrate a general rule: *any function of random variables is also random variable*.

Are D_1 and T independent? That is, is the sum, T, of the two dice independent of the outcome, D_1, of the first die? Intuitively, the answer appears to be no. To prove this, let's use the Definition 1.4 of independence. We must find $x_1, x_2 \in \mathbb{R}$ such that $\Pr\{x_2\} \neq 0$ and the condition in the second definition does not hold.

For example, we can choose $x_1 = 2$ and $x_2 = 3$:

$$\Pr\{T = 2 \mid D_1 = 3\} = 0 \neq \frac{1}{36} = \Pr\{T = 2\}.$$

The first probability is zero, since if we roll a three on the first die ($D_1 = 3$), then there is no way that the sum of both dice is two ($T = 2$). On the other hand, if we throw both dice, the probability that the sum is two is $1/36$, since we could roll two ones.

Are S and D_1 independent? That is, is the probability of the event, S, that the sum of both dice is seven independent of the outcome, D_1, of the first die? Once again, intuition suggests that the answer is "no". Surprisingly, however, these two random variables *are* actually independent!

Proving that two random variables are independent requires some work. Let's use Definition 1.3 of independence based on conditional probability. We must show that for all x_1, x_2 in $\mathbb{R}$ such that $\Pr\{D_1 = x_2\} \neq 0$:

$$\Pr\{S = x_1 \mid D_1 = x_2\} = \Pr\{S = x_1\}.$$

First, notice that we only have to show the equation for values of x_2 such that $\Pr\{D_1 = x_2\} \neq 0$. This means we only have to consider x_2 equal to 1, 2, 3, 4, 5, or 6. If x_1 is neither 0 nor 1, then the condition holds trivially because both sides are zero. So it remains to check the equation for the cases where $x_1 \in \{0, 1\}$ and $x_2 \in \{1, 2, 3, 4, 5, 6\}$, that is, a total of $2 \cdot 6 = 12$ cases.

Two observations make this easier. First, there are $6 \cdot 6 = 36$ outcomes in the sample space for this experiment. The outcomes are equiprobable, so each outcome has probability $1/36$. The two dice sum to seven in six outcomes: $1 + 6$, $2 + 5$, $3 + 4$, $4 + 3$, $5 + 2$, and $6 + 1$. Therefore, the probability of rolling a seven, $\Pr\{S = 1\}$, is $6/36 = 1/6$.

Second, after we know the result of the first die, there is always exactly one value for the second die that makes the sum seven. For example, if the first die is 2, then the sum is seven only if the second die is a 5. Therefore, $\Pr\{S = 1 \mid D_1 = x_2\} = 1/6$ for $x_2 = 1, 2, 3, 4, 5$, or 6.

These two observations establish the independence condition in six cases:

$$\begin{aligned}
\Pr\{S = 1 \mid D_1 = 1\} = {} & \frac{1}{6} = \Pr\{S = 1\} \\
\Pr\{S = 1 \mid D_1 = 2\} = {} & \frac{1}{6} = \Pr\{S = 1\} \\
& \vdots \\
\Pr\{S = 1 \mid D_1 = 6\} = {} & \frac{1}{6} = \Pr\{S = 1\}
\end{aligned}$$

The remaining cases are complementary to the the first six. For example, we know that $\Pr\{S = 0\} = 5/6$, since the complementary event, $S = 1$, has probability $1/6$.

$$\begin{aligned}
\Pr\{S = 0 \mid D_1 = 1\} = {} & \frac{5}{6} = \Pr\{S = 0\} \\
\Pr\{S = 0 \mid D_1 = 2\} = {} & \frac{5}{6} = \Pr\{S = 0\} \\
& \vdots \\
\Pr\{S = 0 \mid D_1 = 6\} = {} & \frac{5}{6} = \Pr\{S = 0\}
\end{aligned}$$

We have established that the independence condition holds for all necessary $x_1, x_2 \in \mathbb{R}$. This proves that S and D_1 are independent after all!

1.5.4 Mutual Independence

The definition of mutual independence for random variables is similar to the definition for events.

Definition 1.5. Random variables $R_1, R_2, \ldots$ are *mutually independent* iff

$$\Pr\left\{\bigcap_i [R_i = x_i]\right\} = \prod_i \Pr\{R_i = x_i\},$$

for all $x_1, x_2, \dots \in \mathbb{R}$.

For example, consider the experiment of throwing three independent, fair dice. Random variable R_1 is the value of the first die. Random variable R_2 is the sum of the first two dice, mod 6. Random variable R_3 is the sum of all three values, mod 6. These three random variables are mutually independent. Can you prove it?

2 Probability Density Functions

A random variable is a function from the sample space of an experiment to the real numbers. As a result, every random variable is bound up in some particular experiment. Often, however, we want to describe a random variable independent of any experiment. This consideration motivates the notion of a *probability density function*.

Definition 2.1. The *probability density function (pdf)* for a random variable R is the function f_R : range $(R) \to [0,1]$ defined by:

$$f_R(x) ::= \Pr\{R = x\}$$

It's sometimes convenient to apply f_R to values that are not in the range of R. By convention, we say f_R equals zero for such values.

The probability density function is also sometimes called the *point density* function. A consequence of this definition is that $\sum_x f_R(x) = 1$, since we are summing the probabilities of all outcomes in the sample space.

Definition 2.2. The *cumulative distribution function* for a random variable, R, is the function F_R : $\mathbb{R} \to [0,1]$ defined by:

$$F_R(x) ::= \Pr\{R \leq x\} = \sum_{\substack{y \leq x, \\ y \in \text{range}\,(R)}} f_R(y).$$

Note that neither the probability density function nor the cumulative distribution function involves the sample space of an experiment; both are functions from $\mathbb{R}$ to $[0,1]$. This allows us to study random variables without reference to a particular experiment. In these Notes, we will look at three distributions and will see more in upcoming lectures.

2.1 Bernoulli Distribution

For our first example, let B be a Bernoulli (indicator) random variable that is 0 with probability p and 1 with probability $1-p$. We can compute the probability density function f_B at 0 and 1 as follows:

$$\begin{aligned} f_B(0) &= \Pr\{B = 0\} = p, \\ f_B(1) &= \Pr\{B = 1\} = 1 - p. \end{aligned}$$

Similarly, we can compute the cumulative distribution function F_B:

$$\begin{aligned} F_B(0) &= \Pr\{B \leq 0\} = p, \\ F_B(1) &= \Pr\{B \leq 1\} = 1. \end{aligned}$$

2.2 Uniform Distribution

Next, let U be a random variable that is uniform on $\{1, \ldots, N\}$. That is, U takes on value k with probability $1/N$ for all $1 \leq k \leq N$. Its probability density and cumulative distribution functions are:

$$\begin{aligned} f_U(k) &::= \Pr\{U = k\} = \frac{1}{N}, \\ F_U(k) &::= \Pr\{U \leq k\} = \frac{k}{N}, \end{aligned}$$

for $1 \leq k \leq N$.

Uniform distributions are very common. For example, the outcome of a fair die is uniform on $\{1, \ldots, 6\}$. An example based on uniform distributions will be presented in the next section. But first, let's define the third distribution.

2.3 Binomial Distribution

We now introduce a third distribution, called the *binomial distribution*. This is the most important and commonly occurring distribution in Computer Science.

Let H be the number of heads in n independent flips of a coin. The density function of H is called a *binomial* density function. The coin need not be fair; we allow biased coins where the probability is p that a Head will come up. To determine exactly what the density function of H is, we need to know the two parameters n and p.

More generally, the binomial distribution describes the probabilities for all possible numbers of occurrences of independent events, for example the number of faulty connections in a circuit where the probabilities of failure for the individual connections are independent.

Definition 2.3. The *unbiased binomial* density function is the function $f_n : \mathbb{R} \to [0, 1]$ defined by

$$f_n(k) ::= \binom{n}{k} 2^{-n}$$

where n is a positive integer parameter.

The *general binomial* density function is the function $f_{n,p} : \mathbb{R} \to [0, 1]$ defined by

$$f_{n,p}(k) ::= \binom{n}{k} p^k (1 - p)^{n-k}$$

where parameter n is a positive integer and $0 < p < 1$.

The unbiased binomial density function is the special case of the general binomial density function where the coin is fair, *viz.*, the parameter p is equal to $1/2$.

3 Examples Involving Probability Distributions

3.1 Uniform Distributions and the Numbers Game

Suppose we are given two envelopes, each containing an integer in the range $0, 1, \dots 100$, and we are guaranteed that the two integers are distinct. To win the game, we must determine which envelope contains the larger number. Our only advantage is that we are allowed to peek at the number in one envelope; we can choose which one. Can we devise a strategy that gives us a better than 50% chance of winning?

For example, suppose we are playing the game and are shown the two envelopes. Now we could guess randomly which envelope contains the larger number, and not even bother to peek in one envelope. With this strategy, we have a 50% chance of winning.

Suppose we try to do better. We peek in the left envelope and see the number 12. Since 12 is a small number, we guess that the right envelope probably contains the larger number. Now, we might be correct. On the other hand, maybe the the person who wrote the numbers decided to be tricky, and made *both* numbers small! Then our guess is not so good!

An important point to remember is that the integers in the envelope might *not* be random. We should assume that the person who writes the numbers is trying to defeat us; she may use randomness or she may not— we don't know!

3.1.1 A Winning Strategy

Amazingly, there is a strategy that wins more than 50% of the time, regardless of the integers in the envelopes. Here is the basic idea:

Suppose we somehow knew a number x between the larger and smaller number. Now we peek in an envelope and see some number. If this number is larger than x, then it must be the larger number. If the number we see is smaller than x, then the larger number must be in the other envelope. In other words, if we know x, then we are guaranteed to win.

Of course, we do not know the number x, so what can we do? Guess!

With some positive probability, we will guess x correctly. If we guess correctly, then we are guaranteed to win! If we guess incorrectly, then we are no worse off than before; our chance of winning is still 50%. Combining these two cases, our overall chance of winning is better than 50%!

This argument may sound implausible, but we can justify it rigorously. The key is *how* we guess the number x. That is, what is the probability density function of x? The best answer turns out to be a uniform density.

Let's describe the strategy more formally and then compute our chance of winning. Call the integers in the envelopes y and z and suppose $y < z$. For generality, suppose that each number is in the range $0, 1, \dots, n$. Above, we considered the case $n = 100$. The number we see by peeking is denoted r. Here is the winning strategy:

1. Guess a number x from the set

$$\left\{1 - \frac{1}{2}, 2 - \frac{1}{2}, \dots, n - \frac{1}{2}\right\}$$

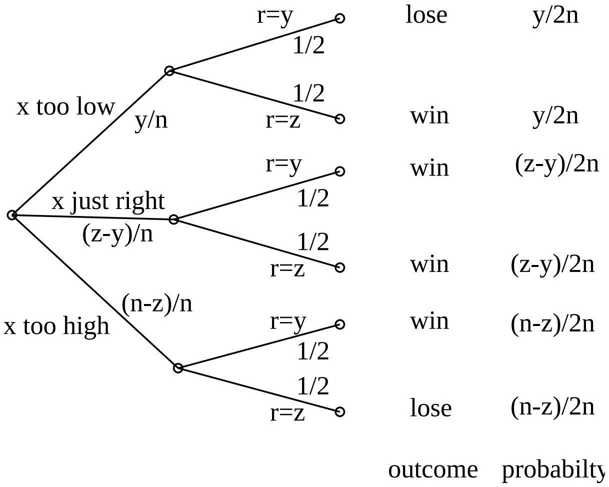


Figure 1: This is the tree diagram for the Numbers Game.

with the uniform distribution. That is, each value is selected with probability $1/n$. (We pick x to be something-and-a-half simply to avoid ties with integers in the envelopes.)

2. Peek into a random envelope. We see a value r that is either y or z. Each envelope is chosen with probability $1/2$, and the choice is independent of the number x.

3. Hope that $y < x < z$.

4. If $r > x$, then guess that r is the larger number, that is the envelope we peeked into is the one that contains the larger number. On the other hand, if $r < x$, then guess that the larger number is in the other envelope.

We can compute the probability of winning by using the tree diagram in Figure 1 and the usual four-step method.

Step 1: Find the sample space. We either choose x too low, too high, or just right. Then we either choose $r = y$ or $r = z$. As indicated in the figure, this gives a total of six outcomes.

Step 2: Define events of interest. We are interested in the event that we correctly pick the larger number. This event consists of four outcomes, which are marked "win" in the figure.

Step 3: Compute outcome probabilities. As usual, we first assign probabilities to edges. First, we guess x. The probability that our guess of x is too low is y/n, the probability that our guess is too high is $(n - z)/n$, and the probability of a correct guess is $(z - y)/n$. We then select an envelope; $r = y$ and $r = z$ occur with equal probability, independent of the choice of x. The probability of an outcome is the product of the probabilities on the corresponding root-to-leaf path, as shown in the figure.

Step 4: Compute event probabilities. The probability of winning is the sum of the probabilities of the four winning outcomes. This gives:

$$\begin{aligned}
\Pr\{\text{winning}\} &= \frac{y}{2n} + \frac{z-y}{2n} + \frac{z-y}{2n} + \frac{n-z}{2n} \\
&= \frac{n+z-y}{2n} \\
&= \frac{1}{2} + \frac{z-y}{2n} \\
&\geq \frac{1}{2} + \frac{1}{2n}
\end{aligned}$$

In the final equality, we use the fact that the larger number z is at least 1 greater than the smaller number y, since they must be distinct.

We conclude that the probability of winning with this strategy is at least $1/2 + 1/2n$, regardless of the integers in the envelopes!

For example, if the numbers in the envelopes are in the range $0, \dots 100$, then the probability of winning is at least $1/2 + 1/200 = 50.5\%$. Even better, if the numbers are constrained to be in the range $0, \dots, 10$, then the probability of winning rises to 55%! By Las Vegas standards, these are great odds!

3.1.2 Optimality of the Winning Strategy

What strategy should our opponent use in putting the numbers into the envelopes? That is, how can he ensure that we do not get, say, a 60% chance of winning?

Of course, our opponent could try to be clever, putting in two low numbers and then two high numbers, etc. But then there is no guarantee that we will not catch on and start winning every time!

It turns out that our opponent should also use a randomized strategy involving the uniform distribution. In particular, he should choose y from $\{0, \dots n-1\}$ uniformly, and then let $z = y + 1$. That is, he should randomly choose a pair of consecutive integers like $(6, 7)$ or $(73, 74)$ with the uniform distribution.

Claim 3.1. *If the opponent uses the strategy above, then* $\Pr\{\textit{we win}\} \leq 1/2 + 1/2n$ *for every strategy we can adopt.*

Claim 3.1 is not hard to prove once we define just what a "strategy" can be, but we won't elaborate that here. One of consequence is that both our strategy above of guessing x and the opponent's strategy above are *optimal*: we can win with probability *at least* $1/2 + 1/2n$ regardless of what our opponent does, and our opponent can ensure that we win with probability *at most* $1/2 + 1/2n$ regardless of what we do.

3.2 Binomial Distribution Examples

3.2.1 The Space Station *Mir*

The troubled space station *Mir* has n parts, each of which is faulty with probability p. Assume that faults occur independently, and let the random variable R be the number of faulty parts. What is the probability density of R, that is, what is $\Pr\{R = k\}$? We can answer this with the usual four-step method, though we will not draw a tree diagram.

Step 1: Find the sample space. We can characterize Mir with a string of W's and F's of length n. A W in the i-th position indicates that the i-th part is working, and an F indicates that the i-th part is faulty. Each such string is an outcome, and the sample space $\mathcal{S}$ is the set of all 2^n such strings.

Step 2: Define events of interest. We want to find the probability that there are exactly k faulty parts; that is, we are interested in the event that $R = k$.

Step 3: Compute outcome probabilities. Since faults occur independently, the probability of an outcome such as $FWFWW$ is simply a product such as $p(1-p)p(1-p)(1-p) = p^2(1-p)^3$. Each F contributes a p term and each W contributes a $(1-p)$ term. In general, the probability of an outcome with k faulty parts and $n-k$ working parts is $p^k(1-p)^{n-k}$.

Step 4: Compute event probabilities.

We can compute the probability that k parts are faulty as follows:

$$\begin{aligned}
\Pr\{R = k\} &= \sum_{w \in [R=k]} p^k(1-p)^{n-k} &&(1)\\
&= (\text{\# of length-}n\text{ strings with }k\ F\text{'s}) \cdot p^k(1-p)^{n-k} &&(2)\\
&= \binom{n}{k} p^k(1-p)^{n-k} &&(3)
\end{aligned}$$

Equation (1) uses the definition of the probability of an event. Then (2) follows because all terms in the summation are equal, and then (3) follows because there are $\binom{n}{k}$ strings of length n with k occurrrences of F.

We can now see that the probability density for the number of faulty parts is precisely the general binomial density:

$$f_R(k) ::= \Pr\{R = k\} = \binom{n}{k} p^k(1-p)^{n-k} = f_{n,p}(k).$$

As a "sanity" check, we should confirm that the sum, $\sum_k f_R(k)$, of these probabilities is one. This fact follows from the Binomial Theorem:

$$1 = (p + (1-p))^n = \sum_{k=0}^{n} \binom{n}{k} p^k(1-p)^{n-k}.$$

In general, the binomial distribution arises whenever we have n independent Bernoulli variables with the same distribution. In this case, the Bernoulli variables indicated whether a part was faulty or not. As another example, if we flip n fair coins, then the number of heads has an unbiased binomial density.

3.2.2 Leader Election

There are n persons in a room. They wish to pick one of themselves as their leader. They wish to do this in a fair and democratic way, so that each and everyone has the same chance to be the leader. The scheme they employ is for everyone to toss a coin. If exactly one person tosses a head that person is elected the leader. If no persons or more than one person tosses heads then they repeat the entire process.

If the coins they use have probability p of coming up heads then what should p be to maximize the probability of selecting a leader in a given round? If n coins are tossed then the probability of having exactly one head is $\binom{n}{1}p(1-p)^{n-1}$. Notice that if p is too large then the likelihood of tossing multiple heads becomes high, whereas if p is too small then no one tosses a head. By differentiating the probability w.r.t. p and then equating to 0, we find that the maximum occurs when $p = 1/n$. Hence, they should use coins so that the probability of coming up heads is $1/n$. When they use such coins then the probability of selecting a leader in a given round is

$$\binom{n}{1}\frac{1}{n}(1-\frac{1}{n})^{n-1} \sim 1/e.$$

Leader election is a very common and important idea in distributed computing. One example is how a set of devices that share a single communication channel (whether wireless or an ethernet cable) may decide which device gets to broadcast. If more than one device broadcasts at the same time, the message will be lost. So the devices keep trying to elect a leader and when they succeed, the leader gets to broadcast on the channel.[2] An interesting question is: given some probability of successfully choosing a leader in a given round, how many rounds do we expect the devices have to try before they successfully send a message? We'll consider this type of question in later Course Notes.

4 The Shape of the Binomial Distribution

The binomial distribution is somewhat complicated, and it's hard to see its qualitative behavior for large k and n.

For example, suppose I flip 100 coins. Here are some basic questions we might ask:

- what is the most likely number of heads?
- what the probability of exactly 50 heads?
- the probability of exactly 25 heads?
- the probability of less than 25 heads?
- probability of exactly 25 heads, given at most 25?

To answer these questions, we will develop some closed form approximations that will help us understand the properties of the binomial density and cumulative distribution. Let's first consider the case when the coin is fair: the *unbiased* density, namely,

$$f_{n,1/2}(k) ::= \binom{n}{k}2^{-n}.$$

[2] Ethernet uses a variant of this idea called *binary exponential backoff*, where the bias p of the leader election coin is constantly adjusted because n is unknown. Probabilistic analysis is an important part of Network theory.

4.1 The central term

Where is $f_{n,p}(k)$ maximized? It's shown in Spring '02, Problem Set 9 that $f_{n,p}(k)$ increases until $k = p(n+1)$, and decreases after. So for $p = 1/2$, the central term is essentially at $k = n/2$. Now, by Stirling's formula we have

$$\binom{n}{n/2} = \frac{n!}{(n/2)!(n/2)!} \sim \frac{\sqrt{2\pi n}\left(\frac{n}{e}\right)^n}{\left(\sqrt{\pi n}\left(\frac{n}{2e}\right)^{n/2}\right)^2} = \sqrt{\frac{2}{\pi n}}2^n.$$

So

$$f_{n,1/2}(n/2) \sim \sqrt{\frac{2}{\pi n}}. \tag{4}$$

Note this is an asymptotic bound. For $n = 100$ (our question about coins) we have $1/\sqrt{50\pi} \approx 0.079788$, so the probability of throwing exactly 50 heads in 100 tosses is about 8%. In fact, the bound given above is very close to the true value; in this case, the exact answer is $0.079589\ldots$. In general, to determine the accuracy of this estimate we'll need to use the form of Stirling's formula that gives upper and lower bounds, which we consider below.

4.2 The tails

We can generalize the estimate of the central term at $(1/2)n$ to terms at factors other than $1/2$. Namely, we estimate $f_{n,1/2}(\alpha n)$ when $\alpha \neq 1/2$ by first estimating the binomial coefficient

Lemma.

$$\binom{n}{\alpha n} \sim 2^{nH(\alpha)}/\sqrt{2\pi\alpha(1-\alpha)n} \tag{5}$$

where

$$H(\alpha) ::= -(\alpha\log_2\alpha + (1-\alpha)\log_2(1-\alpha)).$$

Proof.

$$\begin{aligned}
\binom{n}{\alpha n} &::= \frac{n!}{(\alpha n)!((1-\alpha)n)!} \\
&\sim \frac{\sqrt{2\pi n}\left(\frac{n}{e}\right)^n}{\sqrt{2\pi\alpha n}\left(\frac{\alpha n}{e}\right)^{\alpha n}\sqrt{2\pi(1-\alpha)n}\left(\frac{(1-\alpha)n}{e}\right)^{(1-\alpha)n}} \\
&= \left(\frac{1}{\alpha^\alpha(1-\alpha)^{(1-\alpha)}}\right)^n/\sqrt{2\pi\alpha(1-\alpha)n} \\
&= 2^{-(\alpha\log_2\alpha+(1-\alpha)\log_2(1-\alpha))n}/\sqrt{2\pi\alpha(1-\alpha)n} \\
&= 2^{nH(\alpha)}/\sqrt{2\pi\alpha(1-\alpha)n}.
\end{aligned}$$

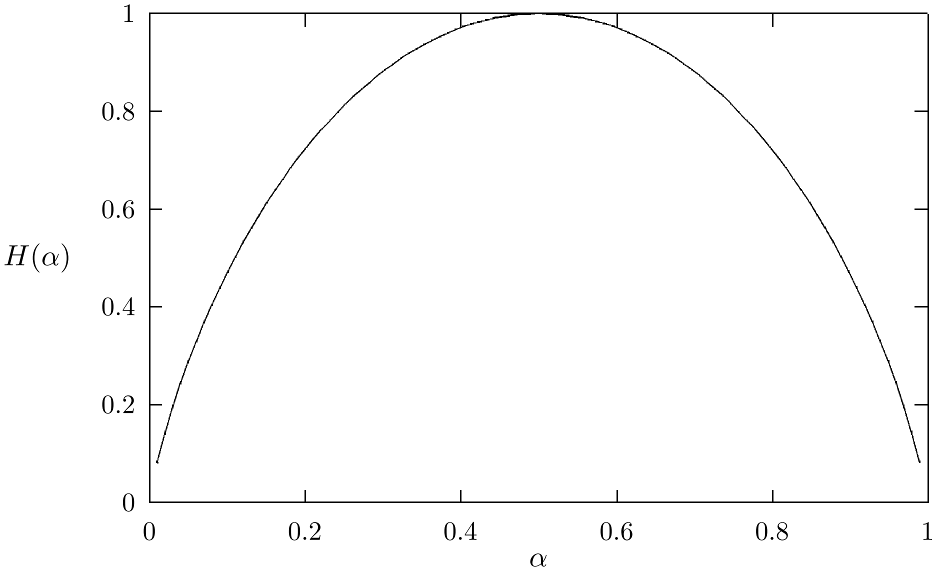


Figure 2: The Entropy Function

□

$H(\alpha)$ is the known as the *entropy function*. Its graph is shown in Figure 2. It is only defined for $0 \leq \alpha \leq 1$, and takes values between 0 and 1 with its maximum at $H(1/2) = 1$. The entropy function plays an important role in thermodynamics and in information theory.

For example, the entropy function arises in the study of how much information is carried in a binary string with a fraction α of the bits set to one. Since there are $\binom{n}{\alpha n}$ such n-bit strings, they can be numbered using $nH(\alpha) + o(\log n)$-bit binary numbers. So the information carried by these n-bits can be "compressed" into $nH(\alpha)$ bits. This observation underlies information-theoretic bounds on the rate at which bits can be reliably communicated over an unreliable communication channel.

With estimate (5) of the binomial coefficient, we conclude

$$f_{n,1/2}(\alpha n) = \binom{n}{\alpha n} 2^{-n} \sim 2^{-n(1-H(\alpha))} / \sqrt{2\pi\alpha(1-\alpha)n}. \tag{6}$$

For $\alpha = 1/2$, this approximation (6) matches our estimate (4) above. But now we can also estimate the probability of throwing exactly 25 heads in 100 tosses. In this case, we substitute $n = 100$, and $\alpha = 1/4$ into (6) and obtain $1.913 \cdot 10^{-7}$. The odds are less than 1 in 5 million for throwing exactly 25 heads in 100 tosses!

The estimate in (6) also provides some important qualitative understanding of the binomial density. Note that for $\alpha \neq 1/2$, we have $1 - H(\alpha) > 0$, so

$$f_{n,1/2}(\alpha n) = O(2^{-\epsilon n})$$

for $1 - H(\alpha) > \epsilon > 0$. In other words, for $\alpha \neq 1/2$,

$f_{n,1/2}(\alpha n)$ **is exponentially small in** n**.**

This means that as n increases, the values any fixed fraction away from $n/2$ rapidly become less likely, and the likely values concentrate more and more tightly around $n/2$.

To handle the general case, we define a generalized entropy function

$$H(\alpha, p) ::= -(\alpha \log_2 p + (1-\alpha)\log_2(1-p)).$$

Then a Stirling formula calculation like the ones above yields

$$f_{n,p}(\alpha n) = 2^{-n(H(\alpha,p)-H(\alpha))} \overbrace{e^{a_n - a_{\alpha n} - a_{(1-\alpha)n}}}^{\sim 1} / \sqrt{2\pi\alpha(1-\alpha)n} \tag{7}$$

The a_n symbols arise from the error in Stirling's approximation; a_n denotes a value between $1/(12n+1)$ and $1/12n$.

The important properties of $H(\alpha, p)$ are:

$$H(\alpha,\alpha) = H(\alpha), \qquad \text{(the ordinary entropy function)} \tag{8}$$

$$H(\alpha,p) > H(\alpha) \geq 0, \qquad \text{for } 0 < p < 1, 0 \leq \alpha \leq 1, p \neq \alpha \tag{9}$$

$$H(\alpha, 1/2) = 1. \tag{10}$$

We observed that the maximum value of $f_{n,p}(\alpha n)$ occurs when $\alpha = p$. For example, in the Mir problem, each part is faulty with probability p, so we would expect pn faulty parts to be the likeliest case. Substituting $\alpha = p$ into (7) and then using equation (8) gives:

$$f_{n,p}(pn) \leq \frac{1}{\sqrt{2\pi p(1-p)n}}.$$

The two sides of this inequality are actually asymptotically equal.

As in the unbiased case, the main term in our approximation (7) of $f_{n,p}(\alpha n)$ is the power of 2. If $p = \alpha$, then $H(p,\alpha) = H(\alpha)$ and the exponent is 0. However, if $p \neq \alpha$, then by equation (9), this term is of the form 2^{-cn} for $c = H(\alpha,p) - H(\alpha) > 0$. Again, this tells us that as n grows large, $f_{n,p}(\alpha n)$ shrinks exponentially, indicating that the values any fixed fraction away from pn rapidly become less likely, and the likely values concentrate more and more tightly around pn. That is, the general binomial density peaks more and more sharply around pn and has the shape shown in Figure 3.

4.3 The Cumulative Distribution Function

4.3.1 25 Heads in 100 Tosses

What is the probability of tossing 25 or fewer heads? Of course, we could sum the probability of zero heads, one head, two heads, . . . , and 25 heads. But there is also a simple formula in terms of the probability density function.

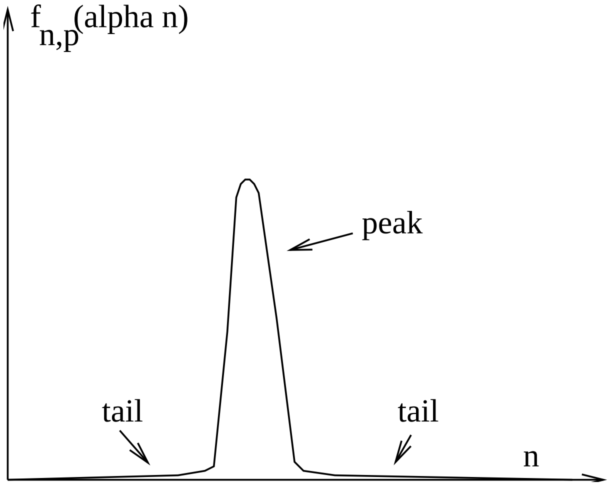


Figure 3: This diagram shows the approximate shape of the binomial density function, $f_{n,p}(\alpha n)$. The horizontal axis goes from 0 to n. The central peak is centered at $\alpha = p$ and has height $\Theta(1/\sqrt{n})$ and width $\Theta(\sqrt{n})$. The "tails" on either side fall off very quickly.

Lemma.

$$F_{n,p}(\alpha n) \leq \left(\frac{1-\alpha}{1-\alpha/p}\right) f_{n,p}(\alpha n) \tag{11}$$

for $\alpha < p$.

This Lemma can be proved by considering the ratio of successive values of $f_{n,p}$. The successive ratios from 0 to pn are approximately constant, so the sum of these values can be bounded by an increasing geometric series. We omit the details.

We can now bound the probability of throwing 25 *or fewer* heads by plugging in the values $n = 100$, $\alpha = 1/4$, and $p = 1/2$. This gives:

$$\Pr\{\text{at most 25 heads}\} = F_{100,1/2}(\frac{1}{4} \cdot 100) \leq \frac{3/4}{1/2} f_{100,1/2}(25) = \frac{3}{2} \cdot 1.913\ldots \cdot 10^{-7}.$$

In other words, the probability of throwing 25 or fewer heads is at most 1.5 times the probability of throwing exactly 25 heads. Therefore, we are at least twice as likely to throw exactly 25 heads as to throw 24 or fewer! This is somewhat surprising; the cases of 0 heads, 1 head, 2 heads, . . . , 24 heads are *together* less likely than the single case of 25 heads. This shows how quickly the tails of the binomial density function fall off!

4.3.2 Transmission Across a Noisy Channel

Suppose that we are transmitting bits across a noisy channel. (For example, say your modem uses a phone line that faintly picks up a local radio station.) Suppose we transmit $10,000$ bits, and each arriving bit is incorrect with probability 0.01. Assume that these errors occur independently. What is the probability that more than 2% of the bits are erroneous?

We can solve this problem using our bound (11) on $F_{n,p}$. However, one trick is required because of a technicality: this bound only holds if $\alpha < p$, so we switch to working with *correct* bits instead of erroneous bits.

$$\Pr\{> \text{ than 2\% errors}\} = \Pr\{\leq \text{ 98\% correct}\} = F_{n,0.99}(0.98n) \leq 1.98\frac{2^{-0.005646 \cdot 10{,}000}}{0.3509\sqrt{10,000}} \leq 2^{-60}$$

The probability that more than 2% of the bits are erroneous is incredibly small! This again demonstrates the extreme improbability of outcomes on the tails of the binomial density.

5 Expected Value

The *expectation* of a random variable is a central concept in the study of probability. It is the average of all possible values of a random variable, where a value is weighted according to the probability that it will appear. The expectation is sometimes also called the *average*. It is also called the *expected value* or the *mean* of the random variable. These terms are all synonymous.

5.1 Two Equivalent Definitions

Definition 5.1. The *expectation*, $\mathrm{E}[R]$, of a random variable, R, on sample space, $\mathcal{S}$, is defined as:

$$\mathrm{E}[R] ::= \sum_{s \in \mathcal{S}} R(s) \cdot \Pr\{s\}. \tag{12}$$

Another equivalent definition is:

Definition 5.2. The *expectation* of random variable, R, is:

$$\mathrm{E}[R] ::= \sum_{r \in \text{range}(R)} r \cdot \Pr\{R = r\}. \tag{13}$$

Actually, there is a technicality implicit in both these definitions that can cause trouble if ignored. In both series (12) and (13), the order of the terms in the series is not specified. This means that the limits of these series are not well-defined unless the series are *absolutely convergent*, *i.e.*, the sum of the *absolute values* of the terms converges. For absolutely convergent series, the order of summation does not matter—the series converges to the same value, or else always diverges, regardless of the order in which the terms are summed.

Definition 5.2 is equivalent to Definition 5.1, because each can be obtained from the other simply by grouping the terms in the series that have the same R value. Regrouping the terms is justified because the series are supposed to be absolutely convergent. Namely, letting r take values over range (R) we have

$$\begin{aligned}
\mathrm{E}[R] &= \sum_{s \in \mathcal{S}} R(s) \cdot \Pr\{s\} && \text{(Def. 5.1)} \\
&= \sum_{r} \sum_{s \in [R=r]} R(s) \cdot \Pr\{s\} && \text{(reordering terms)} \\
&= \sum_{r} \sum_{s \in [R=r]} r \cdot \Pr\{s\} \\
&= \sum_{r} r \sum_{s \in [R=r]} \Pr\{s\} && \text{(factor out constant } r\text{)} \\
&= \sum_{r} r \Pr\{R = r\}. && \text{(Def. of } \Pr\{[R=r]\}\text{)}
\end{aligned}$$

Like other averages, the expected value of a random variable doesn't say anything about what will happen in one trial. For example, the "average" American mother has 2.1 children, but obviously none of them has exactly this number. So we don't expect to see the expected value of a random variable in one trial. Remembering that "expected" value really means "average" value may reduce confusion about this point.

But over a large number of independent trials, we do expect the values to average out close to the expected value. We'll examine this connection between the average of a large number of independent trials and the expectation in detail in Course Notes 12.

5.2 Expected Value of One Die

Suppose we roll a fair, six-sided die. Let the random variable R be the number that comes up. We can compute the expected value of R directly from the definition of expected value. Using the second version of the definition:

$$\begin{aligned}
\mathrm{E}[R] &= \sum_{i=1}^{6} i \cdot \Pr\{R = i\} \\
&= \frac{1}{6} + \frac{2}{6} + \frac{3}{6} + \frac{4}{6} + \frac{5}{6} + \frac{6}{6} \\
&= 3.5
\end{aligned}$$

The average value thrown on a fair die is 3.5. Again, on one trial—a single die roll—we will never get an outcome closer to the expected value than $1/2$. But over many die rolls, the values will almost surely average to a number very close to 3.5.

By itself, the mean of a random variable doesn't say too much about the distribution of values of the variable. Random variables with very different distributions can have the same mean. For example, a nonstandard die with half its sides marked 1 and the other half marked 6 will also have expectation 3.5.

5.3 Expected Value of an Indicator Variable

The expected value of an indicator random variable for an event is just the probability of that event. Namely, let I_A is the indicator random variable for event A, that is, $I_A = 1$ iff A occurs, otherwise $I_A = 0$.

Lemma 5.3. *If I_A is the indicator random variable for event A, then*

$$\mathrm{E}[I_A] = \Pr\{A\}.$$

Proof.

$$\begin{aligned}
\mathrm{E}[I_A] &= 1 \cdot \Pr\{I_A = 1\} + 0 \cdot \Pr\{I_A = 0\} && \text{(Def. 5.2)} \\
&= \Pr\{I_A = 1\} \\
&= \Pr\{A\}. && \text{(Def. of } I_A\text{)}
\end{aligned}$$

□

5.4 The Median is Not the Mean

Expected value, average, and mean are the same thing, but median is entirely different. The median is defined below, but only to make the distinction clear. After this, we won't make further use of the median.

Definition 5.4. The *median* of a random variable R is the unique value r in the range of R such that $\Pr\{R < r\} \le 1/2$ and $\Pr\{R > r\} < 1/2$.

For example, with an ordinary die, the median thrown value is 4, which is not the same as the mean 3.5. The median and the mean can be very far apart. For example, consider a $2n$-sided die, with n 0s and n 100s. The mean is 50, and the median is 100.

5.5 Modified Carnival Dice

Let's look at a modified version of Carnival Dice. The player chooses a number from 1 to 6. He then throws three fair and mutually independent dice. He wins one dollar for *each* die that matches his number, and he loses one dollar if no die matches.

This is better than the original game where the player received one dollar if any die matched, and lost a dollar otherwise. At first glance the new game appears to be fair; after all, the player is now "justly compensated" if he rolls his number on more than one die. In fact, there is still another variant of Carnival Dice in which the payoff is \$2.75 instead of \$3 if all three dice match. In this case, the game appears fair except for the lost quarter in the rare case that all three dice match. This looks like a tiny, tolerable edge for the house.

Let's check our intuition by computing the expected profit of the player in one round of the \$3 variant of Carnival Dice. Let the random variable R be the amount of money won or lost by the player in a round. We can compute the expected value of R as follows:

$$\begin{aligned}
\mathrm{E}[R] &= -1 \cdot \Pr\{0 \text{ matches}\} + 1 \cdot \Pr\{1 \text{ match}\} + 2 \cdot \Pr\{2 \text{ matches}\} + 3 \cdot \Pr\{3 \text{ matches}\} \\
&= -1 \cdot \left(\frac{5}{6}\right)^3 + 1 \cdot 3\left(\frac{1}{6}\right)\left(\frac{5}{6}\right)^2 + 2 \cdot 3\left(\frac{1}{6}\right)^2\left(\frac{5}{6}\right) + 3 \cdot \left(\frac{1}{6}\right)^3 \\
&= \frac{-125 + 75 + 30 + 3}{216} \\
&= \frac{-17}{216}
\end{aligned}$$

Our intuition was wrong! Even with a \$3 payoff for three matching dice, the player can expect to lose $17/216$ of a dollar, or about 8 cents, in every round. This is still a horrible game for the player!

The \$2.75 variant is deceptive. One is tempted to believe that a player is shortchanged only a quarter in the rare case that all three dice match. This is a tiny amount. In fact, though, the player loses this tiny amount *in addition* to the comparatively huge 8 cents per game!

6 Expectation of Natural Number-valued Variables

When the codomain of a random variable is $\mathbb{N}$, there is an alternative way to compute the expected value that can be convenient. We can compute the expected value of a random variable R by summing terms of the form $\Pr\{R > i\}$ instead of terms of the form $\Pr\{R = i\}$. Remember, though, that the theorem only holds if the codomain of R is $\mathbb{N}$!

Theorem 6.1. *If R is a random variable with range $\mathbb{N}$, then*

$$\mathrm{E}[R] = \sum_{i=0}^{\infty} \Pr\{R > i\}.$$

Proof. We will begin with the right-hand expression and transform it into $\mathrm{E}[R]$. Because R is natural number valued, we can expand $\Pr\{R > i\}$ into a series:

$$\Pr\{R > i\} = \Pr\{R = i + 1\} + \Pr\{R = i + 2\} + \Pr\{R = i + 1\} + \cdots.$$

So,

$$\begin{aligned}
\sum_{i=0}^{\infty} \Pr\{R > i\} &= \Pr\{R > 0\} + \Pr\{R > 1\} + \Pr\{R > 2\} + \cdots \\
&= \underbrace{\Pr\{R = 1\} + \Pr\{R = 2\} + \Pr\{R = 3\} + \cdots}_{\Pr\{R>0\}} \\
&\qquad + \underbrace{\Pr\{R = 2\} + \Pr\{R = 3\} + \cdots}_{\Pr\{R>1\}} \\
&\qquad\qquad + \underbrace{\Pr\{R = 3\} + \cdots}_{\Pr\{R>2\}} \\
&= \Pr\{R = 1\} + 2 \cdot \Pr\{R = 2\} + 3 \cdot \Pr\{R = 3\} + \cdots \\
&= \sum_{i=0}^{\infty} i \cdot \Pr\{R = i\} \\
&= \mathrm{E}[R].
\end{aligned}$$

□

6.1 Mean Time to Failure

The Mir space station computer is constantly on the blink. Fortunately, a failure is not catastrophic. Suppose that Mir's main computer has probability p of failing every hour, and assume that failures occur independently. How long should a cosmonaut expect to wait until the main computer fails?

Let the random variable R be the number of hours until the first failure; more precisely, assuming that the hours are numbered $1, 2, 3, \ldots$, then R is the *number of the hour* in which the first failure occurs.

We want to compute the expected value of R. It turns out to be easy to compute $\Pr\{R > i\}$, the probability that the first failure occurs sometime after hour i. Since the range of R is $\mathbb{N}$, we can therefore apply Theorem 6.1 to comput the expected number of hours.

We can compute $\Pr\{R > i\}$ with the usual four-step method.

Step 1: Find the Sample Space. We can regard the sample space as a set of finite strings $W^n F$ for $n \in \mathbb{N}$. A W in the ith position means that the main computer is working during hour i. An F in the $n + 1$st position means that the computer went down during hour $n + 1$.

Step 2: Define Events of Interest. We are concerned with the event that $R > i$. This event consists of all outcomes with no F in the first i positions.

Step 3: Compute Outcome Probabilities. We want to compute the probability of a particular outcome $W^n F$. We reason that since the probability of a W is $(1 - p)$ and of F is p, then we shall define

$$\Pr\{W^n F\} ::= (1 - p)^n p.$$

Step 4: Compute Event Probabilities. We want to compute $\Pr\{R > i\}$. There is no F in the first position of an outcome string with probability $1 - p$. Since failures occur independently, if there is no F in first position, then the probability of F the second position is $1 - p$, etc. Now we can multiply conditional probabilities: the probability that there is no F in the first i positions is $(1 - p)^i$. Therefore,

$$\Pr\{R > i\} = (1 - p)^i. \tag{14}$$

Now we have

$$\begin{aligned}
\mathrm{E}[R] &= \sum_{i=0}^{\infty} \Pr\{R > i\} && \text{(Thm 6.1)} \\
&\sum_{i=0}^{\infty} (1 - p)^i && \text{(by (14))} \\
&= \frac{1}{1 - (1 - p)} && \text{(sum of geometric series)} \\
&= \frac{1}{p}.
\end{aligned}$$

So we have shown that the expected hour when the main computer fails is $1/p$. For example, if the computer has a 1% chance of failing every hour, then we would expect the first failure to occur at the 100th hour, or in about four days. On the bright side, this means that the cosmonaut can expect 99 comfortable hours *without* a computer failure.

6.2 Waiting for a Baby Girl

A couple really wants to have a baby girl. There is a 50% chance that each child they have is a girl, and the genders of their children are mutually independent. If the couple insists on having children until they get a girl, then how many baby boys should they expect to have first?

This is really a variant of the previous problem. The question, "How many hours until the main computer fails?" is mathematically the same as the question, "How many children must the couple have until they get a girl?" In this case, a computer failure corresponds to having a girl, so we should set $p = 1/2$. By the preceding analysis, the couple should expect a baby girl after having $1/p = 2$ children. Since the last of these will be the girl, they should expect just 1 baby boy.

This strategy may seem to be favoring girls, because the couple keeps trying until they have one. However, this effect is counterbalanced by the small possibility of a long sequence of boys.

Suppose the couple has a $3/4$ chance of having a boy instead of $1/2$. Then what is the expected number of children up to and including the first girl?

Let R be the number of children up to and including the first girl. Then

$$\mathrm{E}\left[R\right] = \frac{1}{1/4} = 4.$$

That is, the expected number of boys before the first girl is 3.

7 An Expectation Paradox

Here is a game that reveals a strange property of expectations.

First, you think of a probability distribution function on the natural numbers. This distribution can be absolutely anything you like. For example, you might choose a uniform distribution on $1, 2, \ldots, 6$, giving something like a fair die. Or you might choose a binomial distribution on $0, 1, \ldots, n$. You can even give every natural number a non-zero probability, provided, of course, that the sum of all probabilities is 1. Next, I pick a random number z according to whatever distribution you invent. In the final stage, you pick a random number y according to the same distribution. If your number is bigger than mine ($y > z$), then the game ends. Otherwise, if our numbers are equal or mine is bigger ($y \leq z$), then you pick again, and keep picking until you get a value that is bigger than z.

What is the expected number of picks that you must make?

Certainly, you always need at least one pick—and one pick won't always work—so the expected number is greater than one. An answer like 2 or 3 sounds reasonable, though you might suspect that the answer depends on the distribution. The real answer is amazing: the expected number of picks that you need is always *infinite, regardless of the distribution you choose!* This makes sense if you choose, say, the uniform distribution on $1, 2, \ldots, 6$. After all, there is a $1/6$ chance that I will pick 6. In this case, you must pick forever— you can never beat me!

To calculate the expected number of picks, let's first consider the probability that you need more than one pick. By symmetry there is at least a 50-50 chance that my z is greater than or equal to your y, and you will have to pick again. In other words, you need more than one pick with probability at least $1/2$.

What is the probability that you need more than two picks? Here is an erroneous argument.

False proof. On the first pick, you beat me with probability at most $1/2$. On the second pick, you beat me with probability at most $1/2$. The probability that you fail to beat me on both picks is at most

$$\frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4}.$$

Therefore, the probability that you need more than two picks is at most $1/4$. □

The problem with this reasoning is that beating me on your second pick is not independent of beating me on your first pick, so multiplying the probabilities of these two events isn't valid. It's going to be harder to beat me on your second pick: the fact that you are picking a second time implies that z beat a randomly chosen y. So this means z is likely to be a harder-than-average number to beat on the next pick.

Here is a correct argument: the probability that you need more than two picks is the same as the probability that if I pick z and you independently pick y_1 and y_2, then z is greater than or equal to the maximum of z, y_1, and y_2. But by symmetry, each of these number choices is as likely as any of the others to equal their maximum. So the probability that any one of them is equal to their maximum is at least $1/3$—it will actually be even larger than $1/3$ because of the possibility of ties for the maximum. So in particular, the probabilty that z is the maximum, and hence that you need more than two picks, is at least $1/3$.

Similarly, we can see that the probability that you need more than i picks is at least $1/(i+1)$—just replace "2" by "i" and "3" by "$i+1$" in the previous argument for more than two picks. So if we let T be the random variable equal to the number of picks you need to beat me, then

$$\Pr\{T > i\} \geq \frac{1}{i+1}. \tag{15}$$

This argument also shows your chance of needing more picks will be even larger when there are ties. So you should choose a distribution such that ties are very rare. For example, you might choose the uniform distribution on $\{1, \ldots, 10^{100}\}$. In this case, the probability that you need more than i picks to beat me is very close to $1/(i+1)$ for reasonable i. For example, the probability that you need more than 99 picks is almost exactly 1%. This may sound very promising to you; intuitively, you might expect to win within a reasonable number of picks on average. But now we can verify the claim that, contrary to intuition, the expected number of picks that you need in order to beat me is infinite. The proof is simple:

Proof.

$$\begin{aligned}
\mathrm{E}[T] &= \sum_{i=0}^{\infty} \Pr\{T > i\} && \text{(Thm. 6.1)} \\
&\geq \sum_{i=0}^{\infty} \frac{1}{i+1} && \text{(by (15))} \\
&= \infty. && \text{(sum of Harmonic series)}
\end{aligned}$$

□

This phenomenon can cause all sorts of confusion. For example, suppose we have a communication network. Assume that a packet has a $1/i$ chance of being delayed by i or more steps. This sounds good; there is only a 1% chance of being delayed by 100 or more steps. But, by the argument above, the expected delay for a packet is actually infinite!

8 Linearity of Expectation

8.1 Expectation of a Sum

Expected values obey a simple, very helpful rule called *Linearity of Expectation*. Its simplest form says that the expected value of a sum of random variables is the sum of the expected values of the variables.

Theorem 8.1. *For any random variables R_1 and R_2,*

$$\mathrm{E}[R_1 + R_2] = \mathrm{E}[R_1] + \mathrm{E}[R_2].$$

Proof. Let $T ::= R_1 + R_2$. The proof follows straightforwardly by rearranging terms using Definition 5.1 of $\mathrm{E}[T]$.

$$\begin{aligned} \mathrm{E}[R_1 + R_2] &::= \mathrm{E}[T] \\ &::= \sum_{s \in \mathcal{S}} T(s) \cdot \Pr\{s\} && \text{(Def. 5.1)} \\ &= \sum_{s \in \mathcal{S}} (R_1(s) + R_2(s)) \cdot \Pr\{s\} && \text{(Def. of } T\text{)} \\ &= \sum_{s \in \mathcal{S}} R_1(s) \Pr\{s\} + \sum_{s \in \mathcal{S}} R_2(s) \Pr\{s\} && \text{(rearanging terms)} \\ &= \mathrm{E}[R_1] + \mathrm{E}[R_2]. && \text{(Def. 5.1)} \end{aligned}$$

□

Similarly, we have

Lemma 8.2. *For any random variable, R, and constant, $a \in \mathbb{R}$,*

$$\mathrm{E}[aR] = a\,\mathrm{E}[R].$$

The proof follows easily from the definition of expectation, and we omit it.

Combining Theorem 8.1 and Lemma 8.2, we conclude

Theorem 8.3. *[Linearity of Expectation]*

$$\mathrm{E}[a_1 R_1 + a_2 R_2] = a_1\,\mathrm{E}[R_1] + a_2\,\mathrm{E}[R_2]$$

for all random variables R_1, R_2 and constants $a_1, a_2 \in \mathbb{R}$.

In other words, expectation is a linear function. The same rule holds for more than two random variables:

Corollary 8.4. *For any random variables $R_1, \ldots, R_k$, and constants $a_1, \ldots, a_k \in \mathbb{R}$,*

$$\mathrm{E}\left[\sum_{i=1}^{k} a_i R_i\right] = \sum_{i=1}^{k} a_i \,\mathrm{E}\left[R_i\right].$$

Corollary 8.4 follows from Theorem 8.3 by a routine induction on k which we omit.

The great thing about linearity of expectation is that *no independence is required*. This is really useful, because dealing with independence is a pain, and we often need to work with random variables that are not independent.

8.2 Expected Value of Two Dice

What is the expected value of the sum of two fair dice?

Let the random variable R_1 be the number on the first die, and let R_2 be the number on the second die. We observed earlier that the expected value of one die is 3.5. We can find the expected value of the sum using linearity of expectation:

$$\mathrm{E}\left[R_1 + R_2\right] = \mathrm{E}\left[R_1\right] + \mathrm{E}\left[R_2\right] = 3.5 + 3.5 = 7.$$

Notice that we did *not* have to assume that the two dice were independent. The expected sum of two dice is 7, even if they are glued together! (This is provided that gluing does not change weights to make the individual dice unfair.)

Proving that the expected sum is 7 with a tree diagram would be hard; there are 36 cases. And if we did not assume that the dice were independent, the job would be a nightmare!

8.3 The Hat-Check Problem

There is a dinner party where N men check their hats. The hats are mixed up during dinner, so that afterward each man receives a random hat. In particular, each man gets his own hat with probability $1/N$. What is the expected number of men who get their own hat?

Without linearity of expectation, this would be a very difficult question to answer. We might try the following. Let the random variable R be the number of men that get their own hat. We want to compute $\mathrm{E}\left[R\right]$. By the definition of expectation, we have:

$$\mathrm{E}\left[R\right] = \sum_{k=0}^{N} k \cdot \Pr\left\{R = k\right\}.$$

Now we are in trouble, because evaluating $\Pr\left\{R = k\right\}$ is a mess and we then need to substitute this mess into a summation. Furthermore, to have any hope, we would need to fix the probability of each permutation of the hats. For example, we might assume that all permutations of hats are equally likely.

Now let's try to use linearity of expectation. As before, let the random variable R be the number of men that get their own hat. The trick is to express R as a sum of indicator variables. In particular, let R_i be an indicator for the event that the ith man gets his own hat. That is, $R_i = 1$ is the event that he gets his own hat, and $R_i = 0$ is the event that he gets the wrong hat. The number of men that get their own hat is the sum of these indicators:

$$R = R_1 + R_2 + \cdots + R_N.$$

These indicator variables are *not* mutually independent. For example, if $N - 1$ men all get their own hats, then the last man is certain to receive his own hat. So R_N is not independent of the other indicator variables. But, since we plan to use linearity of expectation, we *don't care* whether the indicator variables are independent, because no matter what, we can take the expected value of both sides of the equation above and apply linearity of expectation:

$$\mathrm{E}[R] = \mathrm{E}[R_1 + R_2 + \cdots + R_N] = \mathrm{E}[R_1] + \mathrm{E}[R_2] + \cdots + \mathrm{E}[R_N].$$

Now by Lemma 5.3, the expected value of an indicator variable is always the probability that the indicator is 1. In this case, the quantity $\Pr\{R_i = 1\}$ is the probability that the ith man gets his own hat, which is just $1/N$. We can now compute the expected number of men that get their own hat:

$$\begin{array}{rcccccccccc} \mathrm{E}[R] & = & \mathrm{E}[R_1] & + & \mathrm{E}[R_2] & + & \cdots & + & \mathrm{E}[R_N] & & \\ & = & \dfrac{1}{N} & + & \dfrac{1}{N} & + & \cdots & + & \dfrac{1}{N} & = & 1. \end{array}$$

We should expect exactly one man to get the right hat!

Notice that we did not assume that all permutations of hats are equally likely or even that all permutations are possible. We only needed to know that each man received his own hat with probability $1/N$. This makes our solution very general, as the next example shows.

8.4 The Chinese Appetizer Problem

There are N people at a circular table in a Chinese restaurant. On the table, there are N different appetizers arranged on a big Lazy Susan. Each person starts munching on the appetizer directly in front of them. Then someone spins the Lazy Susan so that everyone is faced with a random appetizer. What is the expected number of people that end up with the appetizer that they had originally?

This is just a special case of the hat-check problem, with appetizers in place of hats. In the hat check problem, we assumed only that each man received his own hat with probability $1/N$; we made no assumptions about how the hats could be permuted. This problem is a special case, because we happen to know that appetizers are cyclically shifted relative to their initial position. (We assume that each cyclic shift is equally likely.) Our previous analysis still holds; the expected number of people that get their original appetizer is one.

Of course the event that exactly one person gets his original appetizer never happens: either everyone does or no one does. This is another example of the important point that the "expected value" is not the same as "the value we expect," since the expected value may never occur!

8.5 Expected Number of Events that Occur

We can generalize the hat-check and appetizer problems even further. Suppose that we have a collection of events in a sample space. What is the expected number of events that occur? For example, A_i might be the event that the ith man receives his own hat. The number of events that occur is then the number of men that receive their own hat. Linearity of expectation gives a general solution to this problem:

Theorem 8.5. *Given any collection of events $A_1, A_2, \ldots, A_N$, the expected number of these events that occur is*

$$\sum_{i=1}^{N} \Pr\{A_i\}.$$

The theorem says that the expected number of events that occur is the sum the probabilities of the events. For example, in the hat-check problem the probability of the event that the ith man receives his hat is $1/N$. Since there are N such events, the theorem says that the expected number of men that receive their hat is $N(1/N) = 1$. This matches our earlier result. No independence assumptions are needed.

The proof follows immediately from Lemma 5.3 and the fact that R is the sum of the indicator variables for the A_i. That is,

$$R = \sum_i I_{A_i},$$

and so

$$\mathrm{E}[R] = \mathrm{E}\left[\sum_i I_{A_i}\right] = \sum_i \mathrm{E}[I_{A_i}] = \sum_i \Pr\{A_i\}.$$

8.6 Expectation of a Binomial Distribution

Suppose that we independently flip n biased coins, each with probability p of coming up heads. What is the expected number that come up heads?

Let $H_{n,p}$ be the number of heads after the flips. Then $H_{n,p}$ has the binomial distribution with parameters n and p. Now let I_k be the indicator for the kth coin coming up heads. By Lemma 5.3, we have

$$\mathrm{E}[I_k] = p.$$

But

$$H_{n,p} = \sum_{k=1}^{n} I_k,$$

so by linearity

$$\mathrm{E}[H_{n,p}] = \mathrm{E}\left[\sum_{k=1}^{n} I_k\right] = \sum_{k=1}^{n} \mathrm{E}[I_k] = \sum_{k=1}^{n} p = pn.$$

That is, the expectation of a n, p-binomially distributed variable is pn.

9 Conditional Expectation

Just like event probabilities, expectations can be conditioned on some event.

Definition 9.1. We define *conditional expectation,* $\mathrm{E}[R \mid A]$, *of a random variable,* R, *given event,* A:

$$\mathrm{E}[R \mid A] ::= \sum_r r \cdot \Pr\{R = r \mid A\}.$$

In other words, it is the expected value of the variable R once we skew the distribution of R to be conditioned on event A.

Example 9.2. Let D be the outcome of a roll of a random fair die. What is $\mathrm{E}[D \mid D \geq 4]$?

$$\sum_{i=1}^{6} i \cdot \Pr\{D = i \mid D \geq 4\} = \sum_{i=4}^{6} i \cdot 1/3 = 5$$

Since $\mathrm{E}[R \mid A]$ is just an expectation over a different probability measure, we know that the rules for expectation will extend to conditional expectation. For example, conditional expectation will also be linear

Theorem 9.3.

$$\mathrm{E}[a_1 R_1 + a_2 R_2 \mid A] = a_1 \mathrm{E}[R_1 \mid A] + a_2 \mathrm{E}[R_2 \mid A].$$

A real benefit of conditional expectation is the way it lets us divide complicated expectation calculations into simpler cases.

Theorem 9.4. *[Law of Total Expectation] If the sample space is the disjoint union of events* $A_1, A_2, \cdots$, *then*

$$\mathrm{E}[R] = \sum_i \mathrm{E}[R \mid A_i] \Pr\{A_i\}.$$

Proof.

$$\begin{aligned}
\mathrm{E}[R] &= \sum_r r \cdot \Pr\{R = r\} && \text{(Def. 5.2)}\\
&= \sum_r r \cdot \sum_i \Pr\{R = r \mid A_i\} \Pr\{A_i\} && \text{(Total Probability)}\\
&= \sum_r \sum_i r \cdot \Pr\{R = r \mid A_i\} \Pr\{A_i\} && \text{(distribute constant } r\text{)}\\
&= \sum_i \sum_r r \cdot \Pr\{R = r \mid A_i\} \Pr\{A_i\} && \text{(exchange order of summation)}\\
&= \sum_i \Pr\{A_i\} \sum_r r \cdot \Pr\{R = r \mid A_i\} && \text{(factor constant } \Pr\{A_i\}\text{)}\\
&= \sum_i \Pr\{A_i\} \mathrm{E}[R \mid A_i] && \text{(Def. 9.1).}
\end{aligned}$$

□

Example 9.5. Half the people in the world are male, half female. The expected height of a randomly chosen male is $5'11''$, while the expected height of a randomly chosen female is $5'5''$. What is the expected height of a randomly chosen individual?

Let $H(P)$ be the height of the random person P. The events M =“P is male” and F =“P is female” are a partition of the sample space (at least for the moment—though with modern science you never know). Then

$$\begin{aligned}
\mathrm{E}[H] &= \mathrm{E}[H \mid M]\Pr\{M\} + \mathrm{E}[H \mid F]\Pr\{F\} \\
&= 5'11'' \cdot \frac{1}{2} + 5'5'' \cdot \frac{1}{2} \\
&= 5'8''
\end{aligned}$$

We will see in the following sections that the Law of Total Expectation has much more power than one might think.

10 The Expected Value of a Product

10.1 The Product of Independent Expectations

We have determined that the expectation of a sum is the sum of the expectations. The same is not always true for products: in general, the expectation of a product need *not* equal the product of the expectations. But it is true in an important special case, namely, when the random variables are *independent*.

Theorem 10.1. *For any two* independent *random variables, R_1 and R_2,*

$$\mathrm{E}[R_1 \cdot R_2] = \mathrm{E}[R_1] \cdot \mathrm{E}[R_2].$$

Proof. We apply the Law of Total Expectation by conditioning on the value of R_1.

$$\begin{aligned}
\mathrm{E}[R_1 \cdot R_2] &= \sum_{r \in \mathrm{range}(R_1)} \mathrm{E}[R_1 \cdot R_2 \mid R_1 = r] \cdot \Pr\{R_1 = r\} && \text{(Def. 9.1)} \\
&= \sum_r \mathrm{E}[r \cdot R_2 \mid R_1 = r] \cdot \Pr\{R_1 = r\} \\
&= \sum_r r \cdot \mathrm{E}[R_2 \mid R_1 = r] \cdot \Pr\{R_1 = r\} && \text{(Thm 9.3)} \\
&= \sum_r r \cdot \mathrm{E}[R_2] \cdot \Pr\{R_1 = r\} && \text{(R_2 independent of R_1)} \\
&= \mathrm{E}[R_2] \sum_r r \cdot \Pr\{R_1 = r\} && \text{(factor out constant $\mathrm{E}[R_2]$)} \\
&= \mathrm{E}[R_2] \cdot \mathrm{E}[R_1]. && \text{(Def. 5.2)}
\end{aligned}$$

□

Theorem 10.1 extends to a collection of mutually independent variables.

Corollary 10.2. *If random variables $R_1, R_2, \ldots, R_k$ are mutually independent, then*

$$\mathrm{E}\left[\prod_{i=1}^{k} R_i\right] = \prod_{i=1}^{k} \mathrm{E}\left[R_i\right].$$

We omit the simple proof by induction on k.

10.2 The Product of Two Dice

Suppose we throw two *independent*, fair dice and multiply the numbers that come up. What is the expected value of this product?

Let random variables R_1 and R_2 be the numbers shown on the two dice. We can compute the expected value of the product as follows:

$$\mathrm{E}\left[R_1 \cdot R_2\right] = \mathrm{E}\left[R_1\right] \cdot \mathrm{E}\left[R_2\right] = 3.5 \cdot 3.5 = 12.25.$$

Here the first equality holds by Theorem 10.1 because the dice are independent.

Now suppose that the two dice are *not* independent; in fact, assume that the second die is always the same as the first. In this case, the product of expectations will not equal the expectation of the product.

To verify this, let random variables R_1 and R_2 be the numbers shown on the two dice. We can compute the expected value of the product without Theorem 10.1 as follows:

$$\begin{aligned}
\mathrm{E}\left[R_1 \cdot R_2\right] &= \mathrm{E}\left[R_1^2\right] && (R_2 = R_1)\\
&= \sum_{i=1}^{6} i^2 \cdot \Pr\left\{R_1^2 = i^2\right\} && \text{(Def. 5.2)}\\
&= \sum_{i=1}^{6} i^2 \cdot \Pr\left\{R_1 = i\right\}\\
&= \frac{1^2}{6} + \frac{2^2}{6} + \frac{3^2}{6} + \frac{4^2}{6} + \frac{5^2}{6} + \frac{6^2}{6}\\
&= 15\,\frac{1}{6}\\
&\neq 12\,\frac{1}{4}\\
&= \mathrm{E}\left[R_1\right] \cdot \mathrm{E}\left[R_2\right]. && ((10.2))
\end{aligned}$$

11 Expectation of a Quotient

11.1 A RISC Paradox

The following data is taken from a paper by some famous professors. They wanted to show that programs on a RISC processor are generally shorter than programs on a CISC processor. For this purpose, they applied a RISC compiler and then a CISC compiler to some benchmark source programs and made a table of compiled program lengths.

Benchmark	RISC	CISC	CISC/RISC
E-string search	150	120	0.8
F-bit test	120	180	1.5
Ackerman	150	300	2.0
Rec 2-sort	2800	1400	0.5
Average			1.2

Each row contains the data for one benchmark. The numbers in the second and third columns are program lengths for each type of compiler. The fourth column contains the ratio of the CISC program length to the RISC program length. Averaging this ratio over all benchmarks gives the value 1.2 in the lower right. The authors conclude that "CISC programs are 20% longer on average".

However, some critics of their paper took the same data and argued this way: redo the final column, taking the other ratio, RISC/CISC instead of CISC/RISC.

Benchmark	RISC	CISC	RISC/CISC
E-string search	150	120	1.25
F-bit test	120	180	0.67
Ackerman	150	300	0.5
Rec 2-sort	2800	1400	2.0
Average			1.1

From this table, we would conclude that RISC programs are 10% longer than CISC programs on average! We are using the same reasoning as in the paper, so this conclusion is equally justifiable—yet the result is opposite! What is going on?

11.2 A Probabilistic Interpretation

To resolve these contradictory conclusions, we can model the RISC vs. CISC debate with the machinery of probability theory.

Let the sample space be the set of benchmark programs. Let the random variable R be the length of the compiled RISC program, and let the random variable C be the length of the compiled CISC program. We would like to compare the average length, $\mathrm{E}[R]$, of a RISC program to the average length, $\mathrm{E}[C]$, of a CISC program.

To compare average program lengths, we must assign a probability to each sample point; in effect, this assigns a "weight" to each benchmark. One might like to weigh benchmarks based on how frequently similar programs arise in practice. Lacking such data, however, we will assign all benchmarks equal weight; that is, our sample space is uniform.

In terms of our probability model, the paper computes C/R for each sample point, and then averages to obtain $\mathrm{E}[C/R] = 1.2$. This much is correct. The authors then conclude that "CISC programs are 20% longer on average"; that is, they conclude that $\mathrm{E}[C] = 1.2\ \mathrm{E}[R]$.

Similarly, the critics calculation correctly showed that $\mathrm{E}[R/C] = 1.1$. They then concluded that $\mathrm{E}[R] = 1.1\ \mathrm{E}[C]$, that is, a RISC program is 10% longer than a CISC program on average.

These arguments make a natural assumption, namely, that

False Claim 11.1. *If S and T are independent random variables with $T > 0$, then*

$$\mathrm{E}\left[\frac{S}{T}\right] = \frac{\mathrm{E}[S]}{\mathrm{E}[T]}.$$

In other words False Claim 11.1 simply generalizes the rule for expectation of a product to a rule for the expectation of a quotient. But the rule for requires independence, and we surely don't expect C and R to be independent: large source programs will lead to large compiled programs, so when the RISC program is large, so the CISC would be too.

However, we can easily compensate for this kind of dependence: we should compare the lengths of the programs *relative to the size of the source code*. While the lengths of C and R are dependent, it's more plausible that their *relative* lengths will be independent. So we really want to divide the second and third entries in each row of the table by a "normalizing factor" equal to the length of the benchmark program in the first entry of the row.

But note that normalizing this way will have no effect on the fourth column! That's because the normalizing factors applied to the second and and third entries of the rows will cancel. So the independence hypothesis of False Claim 11.1 may be justified, in which case the authors' conclusions would be justified. But then, so would the contradictory conclusions of the critics. Something must be wrong! Maybe it's False Claim 11.1 (duh!), so let's try and prove it.

False proof.

$$\begin{aligned}
\mathrm{E}\left[\frac{S}{T}\right] &= \mathrm{E}\left[S \cdot \frac{1}{T}\right] \\
&= \mathrm{E}[S] \cdot \mathrm{E}\left[\frac{1}{T}\right] && \text{(independence of } S \text{ and } T\text{)} \quad (16) \\
&= \mathrm{E}[S] \cdot \frac{1}{\mathrm{E}[T]}. && (17) \\
&= \frac{\mathrm{E}[S]}{\mathrm{E}[T]}.
\end{aligned}$$

Note that line (16) uses the fact that if S and T are independent, then so are S and $1/T$. This holds because functions of independent random variables yield independent random variables, as shown in Spring '02 Class Problems 10-1, problem 4. □

But this proof is bogus! The bug is in line (17), which assumes

False Theorem 11.2.

$$\mathrm{E}\left[\frac{1}{T}\right] = \frac{1}{\mathrm{E}[T]}.$$

Here is a counterexample:

Example. Suppose $T = 1$ with probability $1/2$ and $T = 2$ with probability $1/2$. Then

$$\begin{aligned}
\frac{1}{\mathrm{E}[T]} &= \frac{1}{1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{2}} \\
&= \frac{2}{3} \\
&\neq \frac{3}{4} \\
&= \frac{1}{1} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} \\
&= \mathrm{E}\left[\frac{1}{T}\right].
\end{aligned}$$

The two quantities are not equal, so False Claim 11.2 really is false.

Unfortunately, the fact that Claim 11.1 and 11.2 are false does not mean that they are never used!

11.3 The Proper Quotient

We can compute $\mathrm{E}[R]$ and $\mathrm{E}[C]$ as follows:

$$\begin{aligned}
\mathrm{E}[R] &= \sum_{i \in \text{Range(R)}} i \cdot \Pr\{R = i\} \\
&= \frac{150}{4} + \frac{120}{4} + \frac{150}{4} + \frac{2800}{4} \\
&= 805
\end{aligned}$$

$$\begin{aligned}
\mathrm{E}[C] &= \sum_{i \in \text{Range(C)}} i \cdot \Pr\{C = i\} \\
&= \frac{120}{4} + \frac{180}{4} + \frac{300}{4} + \frac{1400}{4} \\
&= 500
\end{aligned}$$

Now since $\mathrm{E}[R] / \mathrm{E}[C] = 1.61$, we conclude that the average RISC program is 61% longer than the average CISC program. This is a third answer, completely different from the other two! Furthermore, this answer makes RISC look really bad in terms of code length. This one is the correct conclusion, under our assumption that the benchmarks deserve equal weight. Neither of the earlier results were correct—not surprising since both were based on the same false Claim.

11.4 A Simpler Example [Optional]

[Optional]

The source of the problem is clearer in the following, simpler example. Suppose the data were as follows.

Benchmark	Processor A	Processor B	B/A	A/B
Problem 1	2	1	1/2	2
Problem 2	1	2	2	1/2
Average			1.25	1.25

Now the data for the processors A and B is exactly symmetric; the two processors are equivalent. Yet, from the third column we would conclude that Processor B programs are 25% longer on average, and from the fourth column we would conclude that Processor A programs are 25% longer on average. Both conclusions are obviously wrong.

The moral is that one must be very careful in summarizing data, we must not take an average of ratios blindly!

12 Infinite Linearity of Expectation

We know that expectation is linear over finite sums. It's useful to extend this result to infinite summations. This works as long as we avoid sums whose values may depend on the order of summation.

12.1 Convergence Conditions for Infinite Linearity

Theorem 12.1. *[Linearity of Expectation] Let R_0, R_1, ... , be random variables such that*

$$\sum_{i=0}^{\infty} \mathrm{E}\left[|R_i|\right]$$

converges. Then

$$\mathrm{E}\left[\sum_{i=0}^{\infty} R_i\right] = \sum_{i=0}^{\infty} \mathrm{E}\left[R_i\right].$$

Proof. Let $T ::= \sum_{i=0}^{\infty} R_i$.

We leave it to the reader to verify that, under the given convergence hypothesis, all the sums in the following derivation are absolutely convergent, which justifies rearranging them as follows:

$$\begin{aligned}
\sum_{i=0}^{\infty} \mathrm{E}\left[R_i\right] &= \sum_{i=0}^{\infty} \sum_{s \in \mathcal{S}} R_i(s) \cdot \Pr\{s\} && \text{(Def. 5.1)} \\
&= \sum_{s \in \mathcal{S}} \sum_{i=0}^{\infty} R_i(s) \cdot \Pr\{s\} && \text{(exchanging order of summation)} \\
&= \sum_{s \in \mathcal{S}} \left[\sum_{i=0}^{\infty} R_i(s)\right] \cdot \Pr\{s\} && \text{(factoring out } \Pr\{s\}\text{)} \\
&= \sum_{s \in \mathcal{S}} T(s) \cdot \Pr\{s\} && \text{(Def. of } T\text{)} \\
&= \mathrm{E}\left[T\right] && \text{(Def. 5.1)} \\
&= \mathrm{E}\left[\sum_{i=0}^{\infty} R_i\right]. && \text{(Def. of } T\text{).}
\end{aligned}$$

□

Note that the finite linearity of expectation we established in Corollary 8.4 follows as a special case of Theorem 12.1: since $\mathrm{E}\left[R_i\right]$ is finite, so is $\mathrm{E}\left[|R_i|\right]$, and therefore so is their sum for $0 \leq i \leq n$. Hence the convergence hypothesis of Theorem 12.1 is trivially satisfied if there are only finitely many R_i's.

12.2 A Paradox

One of the simplest casino bets is on "red" or "black" at the roulette table. In each play at roulette, a small ball is set spinning around a roulette wheel until it lands in a red, black, or green colored slot. The payoff for a bet on red or black matches the bet; for example, if you bet \$10 on red and the ball lands in a red slot, you get back your original \$10 bet plus another matching \$10.

In the US, a roulette wheel has 2 green slots among 18 black and 18 red slots, so the probability of red is $p ::= 18/38 \approx 0.473$. In Europe, where roulette wheels have only 1 green slot, the odds for red are a little better —that is, $p = 18/37 \approx 0.486$—but still less than even. To make the game fair, we might agree to ignore green, so that $p = 1/2$.

There is a notorious gambling strategy which seems to guarantee a profit at roulette: bet \$10 on red, and keep doubling the bet until a red comes up. This strategy implies that a player will leave the game as a net winner of \$10 as soon as the red first appears. Of course the player may need an awfully large bankroll to avoid going bankrupt before red shows up—but we know that the mean time until a red occurs is $1/p$, so it seems possible that a moderate bankroll might actually work out. (In this setting, a "win" on red corresponds to a "failure" in a mean-time-to-failure situation.)

Suppose we have the good fortune to gamble against a fair roulette wheel. In this case, our expected win on any spin is zero, since at the ith spin we are equally likely to win or lose $10 \cdot 2^{i-1}$ dollars. So our expected win after any finite number of spins remains zero, and therefore our expected win using this gambling strategy is zero. This is just what we should have anticipated in a fair game.

But wait a minute. As long as there is a fixed, positive probability of red appearing on each spin of the wheel—even if the wheel is unfair—it's *certain* that red will eventually come up. So with probability one, we leave the casino having won \$10, and our expected dollar win is obviously \$10, not zero!

Something's wrong here. What?

12.3 Solution to the Paradox

The expected amount won is indeed \$10.

The argument claiming the expectation is zero is flawed by an invalid use of linearity of expectation for an infinite sum. To pinpoint this flaw, let's first make the sample space explicit: a sample point is a sequence $B^n R$ representing a run of $n \geq 0$ black spins terminated by a red spin. Since the wheel is fair, the probability of $B^n R$ is $2^{-(n+1)}$.

Let C_i be the number of dollars won on the ith spin. So $C_i = 10 \cdot 2^{i-1}$ when red comes up for the first time on the ith spin, that is, at precisely one sample point, namely $B^{i-1}R$. Similarly, $C_i = -10 \cdot 2^{i-1}$ when the first red spin comes up after the ith spin, namely, at the sample points $B^n R$ for $n \geq i$. Finally, we will define C_i by convention to be zero at sample points in which the session ends before the ith spin, that is, at points $B^n R$ for $n < i - 1$.

The dollar amount won in any gambling session is the value of the sum $\sum_{i=1}^{\infty} C_i$. At any sample point $B^n R$, the value of this sum is

$$10 \cdot -(1 + 2 + 2^2 + \cdots + 2^{n-1}) + 10 \cdot 2^n = 10,$$

which trivially implies that its expectation is 10 as well. That is, the amount we are *certain* to leave the casino with, as well as expectation of the amount we win, is $10.

Moreover, our reasoning that $\mathrm{E}[C_i] = 0$ is sound, so

$$\sum_{i=1}^{\infty} \mathrm{E}[C_i] = \sum_{i=1}^{\infty} 0 = 0.$$

The flaw in our argument is the claim that, since the expectation at each spin was zero, therefore the final expectation would also be zero. Formally, this corresponds to concluding that

$$\mathrm{E}[\text{amount won}] = \mathrm{E}\left[\sum_{i=1}^{\infty} C_i\right] = \sum_{i=1}^{\infty} \mathrm{E}[C_i] = 0.$$

The flaw lies exactly in the second equality. This is a case where linearity of expectation fails to hold—even though both $\sum_{i=1}^{\infty} \mathrm{E}[C_i]$ and $\mathrm{E}[\sum_{i=1}^{\infty} C_i]$ are finite—because the convergence hypothesis needed for linearity is false. Namely, the sum

$$\sum_{i=1}^{\infty} \mathrm{E}[|C_i|]$$

does not converge. In fact, the expected value of $|C_i|$ is 10 because $|C_i| = 10 \cdot 2^i$ with probability 2^{-i} and otherwise is zero, so this sum rapidly approaches infinity.

Probability theory truly leads to this apparently paradoxical conclusion: a game allowing an unbounded—even though always finite—number of "fair" moves may not be fair in the end. In fact, our reasoning leads to an even more startling conclusion: even against an *unfair* wheel, as long as there is some fixed positive probability of red on each spin, we are certain to win $10!

This is clearly a case where naive intuition is unreliable: we don't expect to beat a fair game, and we do expect to lose when the odds are against us. Nevertheless, the "paradox" that in fact we always win by bet-doubling cannot be denied.

But remember that from the start we chose to assume that no one goes bankrupt while executing our bet-doubling strategy. This assumption is crucial, because the expected loss while waiting for the strategy to produce its ten dollar profit is actually infinite! So it's not surprising, after all, that we arrived at an apparently paradoxical conclusion from an unrealistic assumption.

This example also serves a warning that in making use of infinite linearity of expectation, the convergence hypothesis which justifies it had better be checked.

13 Wald's Theorem

13.1 Random Length Sums

Wald's Theorem concerns the expected sum of a random number of random variables. For example, suppose that I flip a coin. If I get heads, then I roll two dice. If I get tails, then I roll three dice. What is the expected sum of the dice that I roll? Wald's Theorem supplies a simple answer: the average number of dice I roll is 2 1/2, and the average value of a single die roll is (1+2+ . . . +6)/6 = 3 1/2, so the expected sum is (2 1/2)(3 1/2) = 8 3/4.

In the previous example, we are summing up only two or three values. In the next example, there is no bound on how many values we sum up:

Example 13.1. Repeatedly roll a die until it shows 6. What is the expected sum of the numbers shown in this process?

We can think of each die roll as a random variable: for every positive integer i, let X_i be the outcomes of the ith roll. For definiteness, say $X_i = 0$ if we roll a 6 in fewer than i rolls. So each X_i is a random variable taking values 0,1, . . . ,6. Define $Q = \min\{i \mid X_i = 6\}$. So Q is another random variable whose possible values are *all positive integers*.

The random variable whose expectation we want to calculate is the sum

$$X_1 + X_2 + \cdots + X_Q = \sum_{i=1}^{Q} X_i.$$

Now we know the expected value of each X_i is 3.5, and we also know the expected number of rolls to roll a 6 is 6 (as with the earlier Mir example). Wald's theorem allows us to conclude that the expected sum is $6 \cdot 3.5 = 21$.

The general situation to which Wald's Theorem applies is in computing the total expected cost of a step-by-step probabilistic process, where the cost of a step and the number of steps to complete the process may depend on what happens at each step.

Suppose the expected cost of each step is the same. Then it's reasonable to think that the expected cost of the process is simply this expected cost of a step, times the expected number of steps. In particular, if the cost of the ith step is a random variable, C_i, and Q is the integer-valued positive random variable equal to the number of steps to complete the process, then the total cost for completing the process is precisely $C_1 + C_2 + \cdots + C_Q$. So we reason that

$$\mathrm{E}\left[C_1 + C_2 + \cdots + C_Q\right] = (\text{Expected cost of a step}) \cdot \mathrm{E}\left[Q\right].$$

Actually we don't care about the cost of steps which are not performed. What we really want to say is that if the expected cost of each step is the same, *given that the step is performed*, then the equation above seems reasonable. That is, we only require that $\mathrm{E}\left[C_i \mid Q \geq i\right]$ is the same for all i.

Theorem 13.2. *[Wald] Let $C_1, C_2, \ldots,$ be a sequence of nonnegative random variables, and let Q be a positive integer-valued random variable, all with finite expectations. Suppose that*

$$\mathrm{E}\left[C_i \mid Q \geq i\right] = \mu$$

for some $\mu \in \mathbb{R}$ and for all $i \geq 1$. Then

$$\mathrm{E}\left[C_1 + C_2 + \cdots + C_Q\right] = \mu\, \mathrm{E}\left[Q\right].$$

Proof. Let I_k be the indicator variable for the event $[Q \geq k]$. That is, $I_k = 1$ if the process runs for at least k steps, and $I_k = 0$ if the process finishes in fewer than k steps. So

$$C_1 + C_2 + \cdots + C_Q = \sum_{k=1}^{\infty} C_k I_k. \tag{18}$$

Since all the variables are nonnegative, all the sums and expectations in the following derivation are well-defined, and if any of them is finite, then they all are:

$$\begin{aligned}
&\mathrm{E}\left[C_1 + C_2 + \cdots + C_Q\right] \\
&= \mathrm{E}\left[\sum_{k=1}^{\infty} C_k I_k\right] && \text{((18))} \\
&= \sum_{k=1}^{\infty} \mathrm{E}\left[C_k I_k\right] && \text{(Infinite Linearity Theorem 12.1)} \\
&= \sum_{k=1}^{\infty} \mathrm{E}\left[C_k I_k \mid I_k = 1\right] \cdot \Pr\left\{I_k = 1\right\} + \mathrm{E}\left[C_k I_k \mid I_k = 0\right] \cdot \Pr\left\{I_k = 0\right\} && \text{(Total expectation)} \\
&= \sum_{k=1}^{\infty} \mathrm{E}\left[C_k \cdot 1 \mid I_k = 1\right] \cdot \Pr\left\{I_k = 1\right\} + \mathrm{E}\left[C_k \cdot 0 \mid I_k = 0\right] \cdot \Pr\left\{I_k = 0\right\} \\
&= \sum_{k=1}^{\infty} \mathrm{E}\left[C_k \mid I_k = 1\right] \cdot \Pr\left\{I_k = 1\right\} \qquad + 0 \\
&= \sum_{k=1}^{\infty} \mathrm{E}\left[C_k \mid Q \geq k\right] \cdot \Pr\left\{Q \geq k\right\} && \text{(Def. of } C_k\text{)} \\
&= \sum_{k=1}^{\infty} \mu \cdot \Pr\left\{Q \geq k\right\} && \text{(Def. of } \mu\text{)} \\
&= \mu \cdot \sum_{k=1}^{\infty} \Pr\left\{Q \geq k\right\} && \text{(factoring out constant } \mu\text{)} \\
&= \mu \cdot \sum_{k=0}^{\infty} \Pr\left\{Q > k\right\} && (Q \geq k+1 \text{ iff } Q > k) \\
&= \mu \cdot \mathrm{E}\left[Q\right]. && \text{(Theorem 6.1).}
\end{aligned}$$

□

As a simple application of Wald's Theorem, we can give another proof of the result about mean time to failure:

Corollary 13.3. *In a series of independent trials with probability $p > 0$ of failure at any given trial, the expected number of trials until the first failure is $1/p$.*

Proof. Define the cost C_i of the ith trial to be zero if it succeeds and one if it fails. Let Q be the time to the first failure. So $\sum_{i=1}^{Q} C_i = 1$.

Since the trials are independent, $\mathrm{E}\left[C_i \mid Q \geq i\right] = p$ for all i. Now Wald's Theorem applies:

$$1 = \mathrm{E}\left[\sum_{i=1}^{Q} C_i\right] = \mathrm{E}\left[C_1\right] \cdot \mathrm{E}\left[Q\right] = p \cdot \mathrm{E}\left[Q\right],$$

and so

$$\mathrm{E}\left[Q\right] = \frac{1}{p}.$$

□

13.2 The Paradox Again [Optional]

[Optional]

Played on a fair roulette wheel, our bet-doubling strategy is a step-by-step random process, where the expected cost of a step and the expected number of steps are both finite. In this case, the expected cost is the expected amount won on the step, namely zero, and the expected number of steps is the expected number of spins until red occurs, which we know is $1/(1/2) = 2$. So applying Wald's Theorem,

$$\mathrm{E}\left[\text{amount won}\right] = \mathrm{E}\left[\text{gain on the first spin}\right] \cdot \mathrm{E}\left[\text{number of spins}\right] = 0 \cdot 2 = 0,$$

which is again what we naively would have anticipated in a fair game.

Of course, we know this isn't so. The problem this time is that the cost of a step is negative half the time, and we have proved Wald's Theorem only for nonnegative random variables. Indeed, bet-doubling is an example where the conclusion of Wald's Theorem fails to hold for random variables that are not nonnegative.

14 Building a System

Wald's Theorem turns out to be useful in analyzing algorithms and systems. The following problem was incorrectly solved in a well-known 1962 paper, *The Architecture of Complexity*, by Herbert Simon, who later won the Nobel Prize in economics. The paper is one of the regular readings in 6.033.

Suppose that we are trying to build a system with n components. We add one component at a time. However, whenever we add a component, there is a probability p that the whole system falls apart and we must start over from the beginning. Assume that these collapses occur mutually independently. What is the expected number of steps required to finish building the system?

14.1 The Sample Space

We can regard the sample points in this experiment as finite strings of S's and F's. An S in the ith position indicates that a component is successfully added in the ith step. An F in the ith position indicates that the system falls apart in the ith step. For example, in outcome $SSFSF\ldots$ we add two components, and then the system collapses while we are adding the third. So we start over from scratch. We then add one component successfully, but the system collapses again while we are adding the second. We start over again, etc.

Using this notation, the system is completed after we encounter a string of n consecutive S's. This indicates that all n components were added successfully without the system falling apart. For example, suppose we are building a system with $n = 3$ components. In outcome $SSFSFFSSS$, the system is completed successfully after 9 steps, since after 9 steps we have finally encountered a string of three consecutive S's.

14.2 Tries

Define a "try" to be a sequence of steps that starts with a system of zero components and ends when the system is completed or collapses. Let R_k be the number of steps in the kth try; $R_k ::= 0$ in case the system is completed before the kth try. Also, let Q be the number of tries required to complete the system. The number of steps needed to build the system is then $T ::= \sum_{k=1}^{Q} R_k$. For example, if we are building a system with $n = 3$ components, then we can break outcome $SSFSFFSSS$ into tries as shown below:

$$\underbrace{S \quad S \quad F}_{\substack{R_1 = 3 \\ \text{failure}}} \quad \underbrace{S \quad F}_{\substack{R_2 = 2 \\ \text{failure}}} \quad \underbrace{F}_{\substack{R_3 = 1 \\ \text{failure}}} \quad \underbrace{S \quad S \quad S}_{\substack{R_4 = 3 \\ \text{success!}}}$$

In the above example, four tries are required to complete the system, so we have $Q = 4$. The number of steps needed to complete the system is:

$$T = \sum_{k=1}^{Q} R_k = R_1 + R_2 + R_3 + R_4 = 3 + 2 + 1 + 3 = 9$$

14.3 Applying Wald's Theorem

Our goal is to determine $\mathrm{E}[T]$, the expected number of steps needed to complete the system, which we will do by applying Wald's Theorem.

Each R_k is nonnegative, so the first requirement for applying Wald's Theorem holds.

Since each try starts in the same way and has the same stopping condition, each of the random variables R_k have the same distribution, *given* that the kth try actually occurs. In particular, the expectation of each try has the same value, μ, providing that the try occurs. Of course μ is finite, because every try lasts at most n steps. So the second condition of Wald's Theorem is satisfied, namely, there is a constant $\mu \in \mathbb{R}$ such that

$$\mathrm{E}[R_k \mid Q \geq k] = \mu,$$

for all $k \geq 1$. Finally, we must show that that $\mathrm{E}[Q]$ is finite. We will do this by actually computing it.

14.4 The Expected Number of Tries

Let's compute $\mathrm{E}[Q]$, the expected number of tries needed to complete the system.

First, we will compute the probability that a particular try is successful. A successful try consists of n consecutive S's. The probability of an S in each position is $1-p$. The probability of n consecutive S's is therefore $(1-p)^n$; we can multiply probabilities, since system collapses during a try occur mutually independently.

Now, if a try is successful with probability $(1-p)^n$, what is the expected number of tries needed to succeed? We already encountered this question in another guise. Then we asked the expected number of hours until Mir's main computer went down, given that it went down with probability q in each hour. We found that the expected number of hours until a main computer failure was $1/q$. Here we want the number of tries before the system is completed, given that a try is successful with probability $(1-p)^n$. By the same analysis, the expected number of tries needed to succeed is $1/(1-p)^n$. Therefore, we have:

$$\mathrm{E}[Q] = \frac{1}{(1-p)^n}. \tag{19}$$

This also shows that Q is finite, provided $p \neq 1$.

14.5 The Expected Length of a Try

Notice that the expected number, μ, of steps in a try, given that the try occurs, simply equals $\mathrm{E}[R_1]$, since the first try always occurs. Using the shortcut from Theorem 6.1 to compute the expectation of R_1, we can write:

$$\mu = \sum_{i=0}^{\infty} \Pr\{R_1 > i\} = \sum_{i=0}^{n-1} \Pr\{R_1 > i\}.$$

The second equality holds because a try never lasts for more that n steps, so $\Pr\{R_1 > n\} = 0$.

Now we must evaluate $\Pr\{R_1 > i\}$, the probability that a try consists of more than i steps. This is just the probability that the system does not collapse in the first i steps, which is $(1-p)^i$. Therefore, $\Pr\{R_1 > i\} = (1-p)^i$. Substituting this into the equation above and summing the resulting geometric series gives the expected number of steps in a try:

$$\begin{aligned} \mu &= \sum_{i=0}^{n-1} (1-p)^i \\ &= \frac{1-(1-p)^n}{1-(1-p)} \\ &= \frac{1-(1-p)^n}{p} \end{aligned} \tag{20}$$

14.6 The Expected Number of Steps

Now we can apply Wald's Theorem and compute the expected number of steps needed to complete the system:

$$\begin{aligned} \mathrm{E}[T] &= \mu\,\mathrm{E}[Q] && \text{(Wald' Theorem 13.2.)} \\ &= \frac{1-(1-p)^n}{p} \cdot \frac{1}{(1-p)^n} && \text{(by (20) and (19))} \\ &= \frac{1-(1-p)^n}{p(1-p)^n} \\ &= \frac{1}{p(1-p)^n} - \frac{1}{p} \end{aligned}$$

For example, suppose that there is only a 1% chance that the system collapses when we add a component ($p = 0.01$). The expected number of steps to complete a system with $n = 10$ components is about 10. For $n = 100$ components, the number of steps is about 173. But for $n = 1000$ components, the number is about 2,316,257. As the number of components increases, the number of steps required increases exponentially! The intuition is that adding, say, 1000 components without a single failure is very unlikely; therefore, we need a tremendous number of tries!

14.7 A Better Way to Build Systems

The moral of this analysis is that one should build a system in pieces so that all work is not lost in a single accident.

For example, suppose that we break a 1000 components system into 10 modules, each with 10 submodules, each with 10 components. Assume that when we add a component to a submodule, the submodule falls apart with probability p. Similarly, we can add a submodule to a module in one step, but with probability p the module falls apart into submodules. (The submodules remain intact, however.) Finally, we can add a module into the whole system in one step, but the system falls apart into undamaged modules with probability p.

Altogether, we must build a system of 10 modules, build 10 modules consisting of 10 submodules each, and build 100 submodules consisting of 10 components each. This is equivalent to building 111 systems of 10 components each. The expected time to complete the system is approximately $111 \cdot 10.57 = 1173$ steps. This compares very favorably with the 2.3 million steps required in the direct method!

Deviation from the Mean

1 What the Mean Means

We have focused on the expectation of a random variable because it indicates the "average value" the random variable will take. But what precisely does this mean?

We know a random variable may never actually equal its expectation. We also know, for example, that if we flip a fair coin 100 times, the chance that we actually flip *exactly* 50 heads is only about 8%. In fact, it gets less and less likely as we continue flipping that the number of heads will exactly equal the expected number, *e.g.*, the chance of exactly 500 heads in 1000 flips is less than 3%, in 1,000,000 flips less than 0.1%, ...

But what is true is that the fraction of heads flipped is likely to be *close* to half of the flips, and the more flips, the closer the fraction is likely to be to 1/2. For example, the chance that the fraction of heads is within 5% of 1/2 is

- more than 24% in 10 flips,
- more than 38% in 100 flips,
- more than 56% in 200 flips, and
- more than 89% in 1000 flips.

These numbers illustrate the single most important phenomenon of probability: the average value from repeated experiments is likely to be close to the expected value of one experiment. And it gets more likely to be closer as the number of experiments increases. This result was first formulated and proved by Jacob D. Bernoulli in his book *Ars Conjectandi* (The Art of Guessing) published posthumously in 1713. In his Introduction, Bernoulli comments that[1]

> even the stupidest man—by some instinct of nature *per se* and by no previous instruction (this is truly amazing)—knows for sure that the more observations . . . that are taken, the less the danger will be of straying from the mark.

But he goes on to argue that this instinct should not be taken for granted:

[1] These quotes are taken from Grinstead & Snell, *Introduction to Probability*, American Mathematical Society, p. 310.

> Something further must be contemplated here which perhaps no one has thought about until now. It certainly remains to be inquired whether after the number of observations has been increased, the probability . . . of obtaining the true ratio . . . finally exceeds any given degree of certainty; or whether the problem has, so to speak, its own asymptote—that is, whether some degree of certainty is given which one can never exceed.

Here's how to give a technical formulation of the question Bernoulli wants us to contemplate. Repeatedly performing some random experiment corresponds to defining n random variables equal to the results of n trials of the experiment. That is, we let $G_1, \dots, G_n$ be independent random variables with the same distribution and the same expectation, μ. Now let A_n be the average of the results, that is,

$$A_n ::= \frac{\sum_{i=1}^{n} G_i}{n}.$$

How sure we can be that the average value, A_n, will be close to μ? By letting n grow large enough, can we be as certain as we want that the average will be close, or is there is some irreducible degree of uncertainty that remains no matter how many trials we perform? More precisely, given any positive tolerance, ϵ, how sure can we be that the average, A_n, will be within the tolerance of μ as n grows? In other words, we are asking about the limit

$$\lim_{n\to\infty} \Pr\left\{|A_n - \mu| < \epsilon\right\}.$$

Bernuolli asks if we can be sure this limit approaches certainty, that is, equals one, or whether it approaches some number slightly less than one that cannot be increased to one no matter how many times we repeat the experiment. His answer is that the limit is indeed one. This result is now known as the Weak Law of Large Numbers. Bernoulli says of it:

> Therefore, this is the problem which I now set forth and make known after I have pondered over it for twenty years. Both its novelty and its very great usefulness, coupled with its just as great difficulty, can exceed in weight and value all the remaining chapters of this thesis.

With the benefit of three centuries of mathematical development since Bernoulli, it will be a lot easier for us to resolve Bernoulli's questions than it originally was for him.

2 The Weak Law of Large Numbers

The Weak Law of Large Numbers crystallizes, and confirms, the intuition of Bernoulli's "stupidest man" that the average of a large number of independent trials is more and more likely to be within a smaller and smaller tolerance around the expectation as the number of trials grows.

Theorem 2.1. *[Weak Law of Large Numbers] Let $G_1, \dots, G_n, \dots$ be independent variables with the same distribution and the same expectation, μ. For any $\epsilon > 0$,*

$$\lim_{n\to\infty} \Pr\left\{\left|\frac{\sum_{i=1}^{n} G_i}{n} - \mu\right| \le \epsilon\right\} = 1.$$

This Law gives a high-level description of a fundamental probabilistic phenomenon, but as it stands it does not give enough information to be of practical use. The main problem is that it does not say anything about the *rate* at which the limit is approached. That is, how big must n be to be within a given tolerance of the expected value with a specific desired probability? This information is essential in applications. For example:

- Suppose we want to estimate the number of voters who are registered Republicans. Exactly *how many* randomly selected voters should we poll in order to be sure that 99% of the time, the average number of Republicans in our poll is within 1/2% of the actual percentage in the whole country?
- Suppose we want to estimate the number of fish in a lake. Our procedure will be to catch, tag and release 500 fish caught in randomly selected locations in the lake at random times of day. Then we wait a few days, and catch another 100 random fish. Suppose we discover that 10 of the 100 were previously tagged. Assuming that our 500 tagged fish represent the same proportion of the whole fish population as the ones in our sample of 100, we would estimate that the total fish population was 5000. But how confident can we be of this? Specifically, *how confident* should we be that our estimate of 5000 is within 20% of the actual fish population?
- Suppose we want to estimate the average size of fish in the lake by taking the average of the sizes of the 500 in our initial catch. How confident can we be that this average is within 2% of the average size of all the fish in the lake?

In these Notes we will develop three basic results about this topic of *deviation from the mean*. The first result is Markov's Theorem, which gives a simple but coarse upper bound on the probability that the value of the random variable is more than a certain multiple of its mean. Markov's result holds if we know nothing more than the value of the mean of a random variable. As such, it is very general, but also is much weaker than results which take more information about the random variable into account.

In many situations, we not only know the mean, but also another numerical quantity called the *variance* of the random variable. Our second basic result is Chebyshev's Theorem, which combines Markov's Theorem and information about the variance to give more refined bounds.

The third basic result we call the Pairwise Independent Sampling Theorem. It provides the additional information about rate of convergence we need to calculate numerical answers to questions such as those above. The Sampling Theorem follows from Chebyshev's Theorem and properties of the variance of a sum of independent variables.

Finally, the Weak Law of Large Numbers will be an easy corollary of the Pairwise Independent Sampling Theorem.

2.1 Markov's Theorem

We want to consider the problem of bounding the probability that the value of a random variable is far away from the mean. Our first theorem, Markov's theorem, gives a very rough estimate, based only on the value of the mean.

The idea behind Markov's Theorem can be explained with a simple example of I.Q. measurment. I.Q. was devised so that the average I.Q. measurement would be 100. Now from this fact alone we can conclude that at most 1/2 the population can have an I.Q. of 200 or more, because if more than half had an I.Q. of 200, then the average would have to be more than $(1/2)200 = 100$, contradicting the fact that the average is 100. So the probability that a randomly chosen person has an I.Q. of 200 or more is at most 1/2. Of course this is not a very strong conclusion; in fact no I.Q. of over 200 has ever been recorded. But by the same logic, we can also conclude that at most 2/3 of the population can have an I.Q. of 150 or more. I.Q.'s of over 150 have certainly been recorded, though again, a much smaller fraction of the population actually has an I.Q. that high.

But although these conclusions about I.Q. are weak, they are actually the *strongest possible* general conclusions that can be reached about a random variable using *only* the fact that its mean is 100. For example, if we choose a random variable equal to 200 with probability 1/2, and 0 with probability 1/2, then its mean is 100, and the probability of a value of 200 or more is really 1/2. So we can't hope to get a upper better bound on the probability of 200 than 1/2.

Theorem 2.2 (Markov's Theorem). *If R is a nonnegative random variable, then for all $x > 0$*

$$\Pr\{R \geq x\} \leq \frac{\mathrm{E}[R]}{x}.$$

Proof. We will show that $\mathrm{E}[R] \geq x \Pr\{R \geq x\}$. Dividing both sides by x gives the desired result.

So let I_x be the indicator variable for the event $[R \geq x]$, and consider the random variable xI_x. Note that $R \geq xI_x$, because if $R(w) \geq x$ then $xI_x(w) = x \cdot 1 = x$, and if $R(w) < x$ then $xI_x(w) = x \cdot 0 = 0$. Therefore,

$$\begin{aligned} \mathrm{E}[R] &\geq \mathrm{E}[xI_x] && (R \geq xI_x) \\ &= x\,\mathrm{E}[I_x] && \text{(linearity of expectation)} \\ &= x \Pr\{R \geq x\}. && (\mathrm{E}[I_x] = \Pr\{I_x = 1\}) \end{aligned}$$

□

Markov's Theorem is often expressed in an alternative form, stated below as a corollary.

Corollary 2.3. *If R is a nonnegative random variable, then for all $c > 0$*

$$\Pr\{R \geq c \cdot \mathrm{E}[R]\} \leq \frac{1}{c}.$$

Proof. In Markov's Theorem, set $x = c \cdot \mathrm{E}[R]$. This gives:

$$\Pr\{R \geq c \cdot \mathrm{E}[R]\} \leq \frac{\mathrm{E}[R]}{c \cdot \mathrm{E}[R]} = \frac{1}{c}.$$

□

2.1.1 Examples of Markov's Theorem

Suppose that N men go to a dinner party and check their hats. At the end of the night, each man is given his own hat back with probability $1/N$. What is the probability that x or more men get the right hat?

We can compute an upper bound with Markov's Theorem. Let the random variable, R, be the number of men that get the right hat. In previous notes, we used linearity of expectation to show that $\mathrm{E}[R] = 1$. By Markov's Theorem, the probability that x or more men get the right hat is:

$$\Pr\{R \geq x\} \leq \frac{\mathrm{E}[R]}{x} = \frac{1}{x}.$$

For example, there is no better than a 20% chance that 5 men get the right hat, regardless of the number of people at the dinner party.

The Chinese Appetizer problem is very similar. In this case, N people are eating Chinese appetizers arranged on a circular, rotating tray. Someone then spins the tray so that each person receives a random appetizer. What is the probability that everyone gets the same appetizer as before?

There are N equally likely orientations for the tray after it stops spinning. Everyone gets the right appetizer in just one of these N orientations. Therefore, the correct answer is $1/N$.

But what probability do we get from Markov's Theorem? Let the random variable, R, be the number of people that get the right appetizer. We showed in previous notes that $\mathrm{E}[R] = 1$. Applying Markov's Theorem, we find:

$$\Pr\{R \geq N\} \leq \frac{\mathrm{E}[R]}{N} = \frac{1}{N}.$$

In this case, Markov's Theorem is tight!

On the other hand, Markov's Theorem gives the same $1/N$ bound for the probability everyone gets their hat in the hat check problem. But in reality, the probability of this event is $1/N!$. So Markov's Theorem in this case gives probability bounds that are way off.

2.1.2 Why *R* Must be Nonnegative

The proof of Markov's Theorem requires that the random variable, R, be nonnegative. The following example shows that the theorem is false if this restriction is removed. Let R be -10 with probability $1/2$ and 10 with probability $1/2$. Then we have:

$$\mathrm{E}[R] = -10 \cdot \frac{1}{2} + 10 \cdot \frac{1}{2} = 0$$

Suppose that we now tried to compute $\Pr\{R \geq 5\}$ using Markov's Theorem:

$$\Pr\{R \geq 5\} \leq \frac{\mathrm{E}[R]}{5} = \frac{0}{5} = 0.$$

This is the wrong answer! Obviously, R is at least 5 with probability $1/2$. Remember that Markov's Theorem applies only to nonnegative random variables!

On the other hand, we can still apply Markov's Theorem to bound the probability that an arbitrary variable like R is 5 more. Namely, given any random variable, R with expectation 0 and values ≥ -10, we can conclude that $\Pr\{R \geq 5\} \leq 2/3$.

Proof. Let $T ::= R + 10$. Now T is a nonnegative random variable with expectation $\mathrm{E}[R + 10] = \mathrm{E}[R] + 10 = 10$, so Markov's Theorem applies and tells us that $\Pr\{T \geq 15\} \leq 10/15 = 2/3$. But $T \geq 15$ iff $R \geq 5$, so $\Pr\{R \geq 5\} \leq 2/3$, as claimed. □

2.1.3 Deviation Below the Mean

Markov's Theorem says that a random variable is unlikely to greatly exceed the mean. Correspondingly, there is a theorem that says a random variable is unlikely to be much smaller than its mean.

Theorem 2.4. *Let L be a real number and let R be a random variable such that $R \leq L$. For all $x < L$, we have:*

$$\Pr\{R \leq x\} \leq \frac{L - \mathrm{E}[R]}{L - x}.$$

Proof. The event that $R \leq x$ is the same as the event that $L - R \geq L - x$. Therefore:

$$\begin{aligned} \Pr\{R \leq x\} &= \Pr\{L - R \geq L - x\} \\ &\leq \frac{\mathrm{E}[L - R]}{L - x}. \qquad \text{(by Markov' Theorem)} \qquad (1) \end{aligned}$$

Applying Markov's Theorem in line (1) is permissible since $L-R$ is a nonnegative random variable and $L - x > 0$. □

For example, suppose that the class average on the 6.042 midterm was 75/100. What fraction of the class scored below 50?

There is not enough information here to answer the question exactly, but Theorem 2.4 gives an upper bound. Let R be the score of a random student. Since 100 is the highest possible score, we can set $L = 100$ to meet the condition in the theorem that $R \leq L$. Applying Theorem 2.4, we find:

$$\Pr\{R \leq 50\} \leq \frac{100 - 75}{100 - 50} = \frac{1}{2}.$$

That is, at most half of the class scored 50 or worse. This makes sense; if more than half of the class scored 50 or worse, then the class average could not be 75, even if everyone else scored 100. As with Markov's Theorem, Theorem 2.4 often gives weak results. In fact, based on the data given, the entire class could have scored above 50.

2.1.4 Using Markov To Analyze Non-Random Events [Optional]

[Optional]

In the previous examples, we used a theorem about a random variable to conclude facts about non-random data. For example, we concluded that if the average score on a test is 75, then at most 1/2 the class scored 50 or worse. There is no randomness in this problem, so how can we apply Theorem 2.4 to reach this conclusion?

The explanation is not difficult. For any set of scores $S = \{s_1, s_2, \ldots, s_n\}$, we introduce a random variable, R, such that

$$\Pr\{R = s_i\} = \frac{(\text{\# of students with score } s_i)}{n}$$

We then use Theorem 2.4 to conclude that $\Pr\{R \leq 50\} \leq 1/2$. To see why this means (with certainty) that at most 1/2 of the students scored 50 or less, we observe that

$$\begin{aligned} \Pr\{R \leq 50\} &= \sum_{s_i \leq 50} \Pr\{R = s_i\} \\ &= \sum_{s_i \leq 50} \frac{(\text{\# of students with score } s_i)}{n} \\ &= \frac{1}{n}(\text{\# of students with score 50 or less}). \end{aligned}$$

So, if $\Pr\{R \leq 50\} \leq 1/2$, then the number of students with score 50 or less is at most $n/2$.

3 Chebyshev's Theorem

We have versions of Markov's Theorem for deviations above and below the mean, but often we want bounds that apply in both directions, that is, bounds on the probability that $|R - \mathrm{E}[R]|$ is large.

It is a bit messy to use Markov's inequality directly to bound the probabilty that $|R - \mathrm{E}[R]| \geq x$, since we then would have to compute $\mathrm{E}[|R - \mathrm{E}[R]|]$. However, since $|R|$ and hence $|R|^k$ are nonnegative variables for any R, Markov's inequality also applies to the event $[|R|^k \geq x^k]$. But this event is equivalent to the event $[|R| \geq x]$, so we have:

Corollary 3.1. *For any random variable R, any positive integer k, and any $x > 0$,*

$$\Pr\{|R| \geq x\} \leq \frac{\mathrm{E}\left[|R|^k\right]}{x^k}.$$

The special case of this corollary when $k = 2$ can be applied to bound the random variable, $R - \mathrm{E}[R]$, that measures R's deviation from its mean. Namely

$$\Pr\{|R - \mathrm{E}[R]| \geq x\} = \Pr\left\{(R - \mathrm{E}[R])^2 \geq x^2\right\} \leq \frac{\mathrm{E}\left[(R - \mathrm{E}[R])^2\right]}{x^2}, \tag{2}$$

where the inequality (2) follows from Corollary 3.1 applied to the random variable, $R - \mathrm{E}[R]$. So we can bound the probability that the random variable R deviates from its mean by more than x by an expression decreasing as $1/x^2$ multiplied by the constant $\mathrm{E}\left[(R - \mathrm{E}[R])^2\right]$. This constant is called the *variance of R*.

Definition 3.2. The *variance*, $\mathrm{Var}[R]$, of a random variable, R, is:

$$\mathrm{Var}[R] ::= \mathrm{E}\left[(R - \mathrm{E}[R])^2\right].$$

So we can restate (2) as

Theorem 3.3 (Chebyshev). *Let R be a random variable, and let x be a positive real number. Then*

$$\Pr\{|R - \mathrm{E}[R]| \geq x\} \leq \frac{\mathrm{Var}[R]}{x^2}.$$

The expression $\mathrm{E}\left[(R - \mathrm{E}[R])^2\right]$ for variance is a bit cryptic; the best approach is to work through it from the inside out. The innermost expression, $R - \mathrm{E}[R]$, is precisely the deviation of R from the mean. Squaring this, we obtain, $(R - \mathrm{E}[R])^2$. This is a random variable that is near 0 when R is close to the mean and is a large positive number when R deviates far above or below the mean. The variance is just the average of this random variable, $\mathrm{E}\left[(R - \mathrm{E}[R])^2\right]$. Therefore, intuitively, if R is always close to the mean, then the variance will be small. If R is often far from the mean, then the variance will be large. For this reason, variance is useful in studying the probability that a random variable deviates far from the mean.

3.1 Example: Gambling Games

The relevance of variance is apparent when we compare the following two gambling games.

Game A: We win \$2 with probability $2/3$ and lose \$1 with probability $1/3$.

Game B: We win \$1002 with probability $2/3$ and lose \$2001 with probability $1/3$.

Which game is better financially? We have the same probability, 2/3, of winning each game, but that does not tell the whole story. What about the expected return for each game? Let random variables A and B be the payoffs for the two games. For example, A is 2 with probability 2/3 and -1 with probability $1/3$. We can compute the expected payoff for each game as follows:

$$\begin{aligned} \mathrm{E}[A] &= 2 \cdot \frac{2}{3} + (-1) \cdot \frac{1}{3} = 1, \\ \mathrm{E}[B] &= 1002 \cdot \frac{2}{3} + (-2001) \cdot \frac{1}{3} = 1. \end{aligned}$$

The expected payoff is the same for both games, but they are obviously very different! This difference is hidden by expected value, but captured by variance. We can compute the $\mathrm{Var}[A]$ by working "from the inside out" as follows:

$$\begin{aligned} A - \mathrm{E}[A] &= \begin{cases} 1 & \text{with probability } \frac{2}{3} \\ -2 & \text{with probability } \frac{1}{3} \end{cases} \\ (A - \mathrm{E}[A])^2 &= \begin{cases} 1 & \text{with probability } \frac{2}{3} \\ 4 & \text{with probability } \frac{1}{3} \end{cases} \\ \mathrm{E}\left[(A - \mathrm{E}[A])^2\right] &= 1 \cdot \frac{2}{3} + 4 \cdot \frac{1}{3} \\ \mathrm{Var}[A] &= 2. \end{aligned}$$

Similarly, we have for $\mathrm{Var}[B]$:

$$\begin{aligned} B - \mathrm{E}[B] &= \begin{cases} 1001 & \text{with probability } \frac{2}{3} \\ -2002 & \text{with probability } \frac{1}{3} \end{cases} \\ (B - \mathrm{E}[B])^2 &= \begin{cases} 1,002,001 & \text{with probability } \frac{2}{3} \\ 4,008,004 & \text{with probability } \frac{1}{3} \end{cases} \\ \mathrm{E}\left[(B - \mathrm{E}[B])^2\right] &= 1,002,001 \cdot \frac{2}{3} + 4,008,004 \cdot \frac{1}{3} \\ \mathrm{Var}[B] &= 2,004,002. \end{aligned}$$

The variance of Game A is 2 and the variance of Game B is more than two million! Intuitively, this means that the payoff in Game A is usually close to the expected value of \$1, but the payoff in Game B can deviate very far from this expected value.

High variance is often associated with high risk. For example, in ten rounds of Game A, we expect to make \$10, but could conceivably lose \$10 instead. On the other hand, in ten rounds of game B, we also expect to make \$10, but could actually lose more than \$20,000!

4 Properties of Variance

4.1 Why Variance?

The definition of variance of R as $\mathrm{E}\left[(R - \mathrm{E}[R])^2\right]$ may seem rather arbitrary. The variance is the average *of the square* of the deviation from the mean. For this reason, variance is sometimes called the "mean squared deviation." But why bother squaring? Why not simply compute the average deviation from the mean? That is, why not define variance to be $\mathrm{E}[R - \mathrm{E}[R]]$?

The problem with this definition is that the positive and negative deviations from the mean exactly cancel. By linearity of expectation, we have:

$$\mathrm{E}[R - \mathrm{E}[R]] = \mathrm{E}[R] - \mathrm{E}[\mathrm{E}[R]].$$

Since $\mathrm{E}[R]$ is a constant, its expected value is itself. Therefore

$$\mathrm{E}[R - \mathrm{E}[R]] = \mathrm{E}[R] - \mathrm{E}[R] = 0.$$

By this definition, every random variable has zero variance. That is not useful! Because of the square in the conventional definition, both positive and negative deviations from the mean increase the variance; positive and negative deviations do not cancel.

Of course, we could also prevent positive and negative deviations from cancelling by taking an absolute value. That is, we could define variance to be $\mathrm{E}[|R - \mathrm{E}[R]|]$. There is no great reason not to use this definition. However, the conventional version of variance has some pleasant mathematical properties that the absolute value variant does not. For example, for independent random variables, the variance of a sum is the sum of the variances; that is, $\mathrm{Var}[R_1 + R_2] = \mathrm{Var}[R_1] + \mathrm{Var}[R_2]$. We will prove this fact below.

4.2 Standard Deviation

Due to squaring, the variance of a random variable may be very far from a typical deviation from the mean. For example, in Game B above, the deviation from the mean is 1001 in one outcome and -2002 in the other. But the variance is a whopping 2,004,002. From a dimensional analysis viewpoint, the "units" of variance are wrong: if the random variable is in dollars, then the expectation is also in dollars, but the variance is in square dollars. For this reason, people often describe random variables using standard deviation instead of variance.

Definition 4.1. The *standard deviation* of a random variable R is denoted σ_R and defined as the square root of the variance:

$$\sigma_R ::= \sqrt{\mathrm{Var}[R]} = \sqrt{\mathrm{E}[(R - \mathrm{E}[R])^2]}.$$

So the standard deviation is the square root of the mean of the square of the deviation, or the "root mean square" for short. It has the same units—dollars in our example—as the original random variable and as the mean. Intuitively, it measures the "expected (average) deviation from the mean," since we can think of the square root on the outside as cancelling the square on the inside.

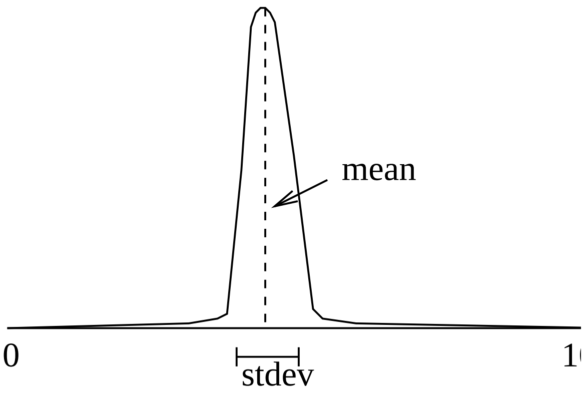


Figure 1: The standard deviation of a distribution says how wide the "main" part of it is.

Example 4.2. The standard deviation of the payoff in Game B is:

$$\sigma_B = \sqrt{\text{Var}[B]} = \sqrt{2,004,002} \approx 1416.$$

The random variable B actually deviates from the mean by either positive 1001 or negative 2002; therefore, the standard deviation of 1416 describes the situations reasonably well.

As can be seen in Figure 1, the standard deviation measures the "width" of the main part of the distribution graph.

4.3 An Alternative Definition of Variance

There is an equivalent way to define the variance of a random variable that is less intuitive, but is often easier to use in calculations and proofs:

Theorem 4.3.

$$\text{Var}[R] = \text{E}\left[R^2\right] - \text{E}^2[R],$$

for any random variable, R.

Here we use the notation $\text{E}^2[R]$ as shorthand for $(\text{E}[R])^2$.

Remember that $\text{E}\left[R^2\right]$ is generally not equal to $\text{E}^2[R]$. We know the expected value of a product is the product of the expected values for independent variables, but not in general. And R is not independent of itself unless it is constant.

Proof. Let $\mu = \text{E}[R]$. Then

$$\begin{aligned}
\text{Var}[R] &= \text{E}\left[(R - \text{E}[R])^2\right] && \text{(Def. 3.2 of variance)} \\
&= \text{E}\left[(R - \mu)^2\right] && \text{(Def. of } \mu\text{)} \\
&= \text{E}\left[R^2 - 2\mu R + \mu^2\right] && \\
&= \text{E}\left[R^2\right] - 2\mu\,\text{E}[R] + \mu^2 && \text{(linearity of expectation)} \\
&= \text{E}\left[R^2\right] - 2\mu^2 + \mu^2 && \text{(definition of } \mu\text{)} \\
&= \text{E}\left[R^2\right] - \mu^2 && \\
&= \text{E}\left[R^2\right] - \text{E}^2[R]. && \text{(definition of } \mu\text{)}
\end{aligned}$$

□

[Optional]

Theorem 4.3 gives a convenient way to compute the variance of a random variable: find the expected value of the square and subtract the square of the expected value. For example, we can compute the variance of the outcome of a fair die as follows:

$$\begin{aligned}
\mathrm{E}\left[R^2\right] &= \frac{1}{6}(1^2+2^2+3^2+4^2+5^2+6^2) = \frac{91}{6},\\
\mathrm{E}^2\left[R\right] &= \left(3\frac{1}{2}\right)^2 = \frac{49}{4},\\
\mathrm{Var}\left[R\right] = \mathrm{E}\left[R^2\right] - \mathrm{E}^2\left[R\right] &= \frac{91}{6} - \frac{49}{4} = \frac{35}{12}.
\end{aligned}$$

This result is particularly useful when we want to estimate the variance of a random variable from a sequence $x_1, x_2, \ldots, x_n$, of sample values of the variable.

Definition 4.4. For any sequence of real numbers $x_1, x_2, \ldots, x_n$, define the *sample mean*, μ_n, and the *sample variance*, v_n, of the sequence to be:

$$\begin{aligned}
\mu_n &::= \frac{\sum_{i=1}^n x_i}{n},\\
v_n &::= \frac{\sum_{i=1}^n (x_i - \mu_n)^2}{n}.
\end{aligned}$$

Notice that if we define a random variable, R, which is equally likely to take each of the values in the sequence, that is $\Pr\{R = x_i\} = 1/n$ for $i = 1, \ldots, n$, then $\mu_n = \mathrm{E}[R]$ and $v_n = \mathrm{Var}[R]$. So Theorem 4.3 applies to R and lets us conclude that

$$v_n = \frac{\sum_{i=1}^n x_i^2}{n} - \left(\frac{\sum_{i=1}^n x_i}{n}\right)^2. \tag{3}$$

This leads to a simple procedure for computing the sample mean and variance while reading the sequence $x_1, \ldots, x_n$ from left to right. Namely, maintain a sum of all numbers seen and also maintain a sum of the squares of all numbers seen. That is, we store two values, starting with the values x_1 and x_1^2. Then, as we get to the next number, x_i, we add it to the first sum and add its square, x_i^2, to the second sum. After a single pass through the sequence $x_1, \ldots, x_n$, we wind up with the values of the two sums $\sum_{i=1}^n x_i$ and $\sum_{i=1}^n x_i^2$. Then we just plug these two values into (3) to find the sample variance.

4.3.1 Expectation Squared [Optional]

[Optional]

The alternate definition of variance given in Theorem 4.3 has a cute implication:

Corollary 4.5. *If R is a random variable, then* $\mathrm{E}\left[R^2\right] \geq \mathrm{E}^2\left[R\right]$.

Proof. We first defined $\mathrm{Var}[R]$ as an average of a squared expression, so $\mathrm{Var}[R]$ is nonnegative. Then we proved that $\mathrm{Var}[R] = \mathrm{E}\left[R^2\right] - \mathrm{E}^2[R]$. This implies that $\mathrm{E}\left[R^2\right] - \mathrm{E}^2[R]$ is nonnegative. Therefore, $\mathrm{E}\left[R^2\right] \geq \mathrm{E}^2[R]$. □

In words, the expectation of a square is at least the square of the expectation. The two are equal exactly when the variance is zero:

$$\mathrm{E}\left[R^2\right] = \mathrm{E}^2[R] \text{ iff } \mathrm{E}\left[R^2\right] - \mathrm{E}^2[R] = 0 \text{ iff } \mathrm{Var}[R] = 0.$$

4.3.2 Zero Variance

When does a random variable, R, have zero variance? ... when the random variable *never* deviates from the mean!

Lemma 4.6. *The variance of a random variable, R, is zero if and only if $R = \mathrm{E}[R]$ for all outcomes with positive probability.*

The final phrase is a technicality; for an outcome with zero probability, R can take on any value without affecting the variance.

Proof. By the definition of variance, $\mathrm{Var}[R] = 0$ is equivalent to the condition $\mathrm{E}\left[(R - \mathrm{E}[R])^2\right] = 0$.

The inner expression, $(R - \mathrm{E}[R])^2$, is always nonnegative because of the square. As a result, $\mathrm{E}\left[(R - \mathrm{E}[R])^2\right] = 0$ if an only if $(R - \mathrm{E}[R])^2 = 0$ for all outcomes with positive probability. Now, the conditions $(R - \mathrm{E}[R])^2 = 0$ and $R = \mathrm{E}[R]$ are also equivalent. Therefore, $\mathrm{Var}[R] = 0$ iff $R = \mathrm{E}[R]$ for all outcomes with positive probability. □

4.3.3 Dealing with Constants

The following theorem describes how the variance of a random variable changes when it is scaled or shifted by a constant.

Theorem 4.7. *Let R be a random variable, and let a and b be constants. Then*

$$\mathrm{Var}[aR + b] = a^2 \,\mathrm{Var}[R]. \tag{4}$$

This theorem makes two points. First, adding a constant b to a random variable does not affect the variance. Second, multiplying a random variable by a constant changes the variance by a *square factor*.

Proof. We will transform the left side of (4) into the right side. The first step is to expand $\mathrm{Var}[aR + b]$ using the alternate definition of variance.

$$\mathrm{Var}[aR + b] = \mathrm{E}\left[(aR + b)^2\right] - \mathrm{E}^2[aR + b].$$

We will work on the first term and then the second term. For the first term, note that by linearity of expectation,

$$\mathrm{E}\left[(aR + b)^2\right] = \mathrm{E}\left[a^2 R^2 + 2abR + b^2\right] = a^2\,\mathrm{E}\left[R^2\right] + 2ab\,\mathrm{E}[R] + b^2. \tag{5}$$

Similarly for the second term:

$$\mathrm{E}^2[aR + b] = (a\,\mathrm{E}[R] + b)^2 = a^2\mathrm{E}^2[R] + 2ab\,\mathrm{E}[R] + b^2. \tag{6}$$

Finally, we substract the expanded second term from the first.

$$\begin{aligned}
\text{Var}\left[aR+b\right] &= \text{E}\left[(aR+b)^2\right] - \text{E}^2\left[aR+b\right] && \text{(Theorem 4.3)}\\
&= a^2\,\text{E}\left[R^2\right] + 2ab\,\text{E}\left[R\right] + b^2 - \\
&\quad (a^2\text{E}^2\left[R\right] + 2ab\,\text{E}\left[R\right] + b^2) && \text{(by (5) and (6))}\\
&= a^2\,\text{E}\left[R^2\right] - a^2\text{E}^2\left[R\right]\\
&= a^2(\text{E}\left[R^2\right] - \text{E}^2\left[R\right])\\
&= a^2\,\text{Var}\left[R\right] && \text{(Theorem 4.3)}
\end{aligned}$$

□

A similar rule holds for the standard deviation when a random variable is adjusted by a constant. Recall that standard deviation is the square root of variance. Therefore, adding a constant b to a random variable does not change the standard deviation. Multiplying a random variable by a constant a multiplies the standard deviation by a. So we have

Corollary 4.8. *The standard deviation of $aR + b$ equals a times the standard deviation of R.*

4.4 Variance of a Sum

Earlier, we claimed that for independent random variables, the variance of a sum is the sum of the variances. An independence condition is necessary. If we ignored independence, then we would conclude that $\text{Var}\left[R + R\right] = \text{Var}\left[R\right] + \text{Var}\left[R\right]$. However, by Theorem 4.7, the left side is equal to $4\,\text{Var}\left[R\right]$, whereas the right side is $2\,\text{Var}\left[R\right]$. This implies that $\text{Var}\left[R\right] = 0$, which, by Lemma 4.6, holds only if R is constant.

However, *mutual* independence is not necessary: *pairwise* independence will do. This is useful to know because there are some important situations involving variables that are pairwise independent but not mutually independent. Matching birthdays is an example of this kind, as we shall see below.

Theorem 4.9. *[Pairwise Independent Additivity of Variance] If $R_1, R_2, \ldots, R_n$ are* pairwise *independent random variables, then*

$$\text{Var}\left[R_1 + R_2 + \ldots + R_n\right] = \text{Var}\left[R_1\right] + \text{Var}\left[R_2\right] + \cdots + \text{Var}\left[R_n\right].$$

Proof. By linearity of expectation, we have

$$\begin{aligned}
\text{E}\left[\left(\sum_{i=1}^{n} R_i\right)^2\right] &= \text{E}\left[\sum_{i=1}^{n}\sum_{j=1}^{n} R_i R_j\right]\\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} \text{E}\left[R_i R_j\right] && \text{(linearity)}\\
&= \sum_{1\le i\ne j\le n} \text{E}\left[R_i\right]\text{E}\left[R_j\right] + \sum_{i=1}^{n} \text{E}\left[R_i^2\right]. && \text{(pairwise independence)} \quad (7)
\end{aligned}$$

In (7), we use the fact from previous Notes that the expectation of the product of two independent variables is the product of their expectations.

Also,

$$\begin{aligned}
\mathrm{E}^2\left[\sum_{i=1}^n R_i\right] &= \left(\mathrm{E}\left[\sum_{i=1}^n R_i\right]\right)^2 \\
&= \left(\sum_{i=1}^n \mathrm{E}\left[R_i\right]\right)^2 && \text{(linearity)} \\
&= \sum_{i=1}^n \sum_{j=1}^n \mathrm{E}\left[R_i\right]\mathrm{E}\left[R_j\right] \\
&= \sum_{1\le i\ne j\le n} \mathrm{E}\left[R_i\right]\mathrm{E}\left[R_j\right] + \sum_{i=1}^n \mathrm{E}^2\left[R_i\right]. && (8)
\end{aligned}$$

So,

$$\begin{aligned}
\mathrm{Var}\left[\left(\sum_{i=1}^n R_i\right)\right] &= \mathrm{E}\left[\left(\sum_{i=1}^n R_i\right)^2\right] - \mathrm{E}^2\left[\sum_{i=1}^n R_i\right] && \text{(Theorem 4.3)} \\
&= \sum_{1\le i\ne j\le n} \mathrm{E}\left[R_i\right]\mathrm{E}\left[R_j\right] + \sum_{i=1}^n \mathrm{E}\left[R_i^2\right] - \\
&\quad \left(\sum_{1\le i\ne j\le n} \mathrm{E}\left[R_i\right]\mathrm{E}\left[R_j\right] + \sum_{i=1}^n \mathrm{E}^2\left[R_i\right]\right) && \text{(by (7) and (8))} \\
&= \sum_{i=1}^n \mathrm{E}\left[R_i^2\right] - \sum_{i=1}^n \mathrm{E}^2\left[R_i\right] \\
&= \sum_{i=1}^n (\mathrm{E}\left[R_i^2\right] - \mathrm{E}^2\left[R_i\right]) && \text{(reordering the sums)} \\
&= \sum_{i=1}^n \mathrm{Var}\left[R_i\right]. && \text{(Theorem 4.3)}
\end{aligned}$$

□

4.5 Variance of a Binomial Distribution

We now have enough tools to find the variance of a binomial distribution. Recall that if a random variable, R, has a binomial distribution, then

$$\Pr\left\{R = k\right\} = \binom{n}{k} p^k (1-p)^{n-k}$$

where n and p are parameters such that $n \ge 1$ and $0 < p < 1$.

We can think of R as the sum of n independent Bernoulli variables. For example, we can regard R as the number of heads that come up when we toss n independent coins, where each coin comes up heads with probability p. Formally, we can write $R = R_1 + R_2 + \cdots + R_n$ where

$$R_i = \begin{cases} 1 & \text{with probability } p, \\ 0 & \text{with probability } 1 - p. \end{cases}$$

Now we can compute the variance of the binomially distributed variable R.

$$\begin{aligned} \text{Var}[R] &= \text{Var}[R_1] + \text{Var}[R_2] + \ldots + \text{Var}[R_n] && \text{(Theorem 4.9)} \\ &= n \,\text{Var}[R_1] && (\text{Var}[R_i] = \text{Var}[R_j]) \\ &= n(\text{E}\left[R_1^2\right] - \text{E}^2[R_1]) && \text{(Theorem 4.3)} \\ &= n(\text{E}[R_1] - \text{E}^2[R_1]) && (R_1^2 = R_1) \\ &= n(p - p^2). && (\text{E}[R_1] = \Pr\{R_1 = 1\} = p) \end{aligned}$$

This shows that the binomial distribution has variance $p(1-p)n$ and standard deviation $\sqrt{p(1-p)n}$. In the special case of an unbiased binomial distribution ($p = 1/2$), the variance is $n/4$ and the standard deviation is $\sqrt{n}/2$.

5 Applications of Chebyshev's Theorem

There is a nice reformulation of Chebyshev's Theorem in terms of standard deviation.

Corollary 5.1. *Let R be a random variable, and let c be a positive real number.*

$$\Pr\{|R - \text{E}[R]| \geq c\sigma_R\} \leq \frac{1}{c^2}.$$

Here we see explicitly how the "likely" values of R are clustered in an $O(\sigma_R)$-sized region around $\text{E}[R]$, confirming that the standard deviation measures how spread out the distribution of R is around its mean.

Proof. Substituting $x = c\sigma_R$ in Chebyshev's Theorem gives:

$$\Pr\{|R - \text{E}[R]| \geq c\sigma_R\} \leq \frac{\text{Var}[R]}{(c\sigma_R)^2} = \frac{\sigma_R^2}{(c\sigma_R)^2} = \frac{1}{c^2}.$$

□

5.1 I.Q. Example

Suppose that, in addition to the average I.Q. being 100, we also know the standard deviation of I.Q.'s is 10. How rare is an I.Q. of 200 or more?

Let the random variable, R, be the I.Q. of a random person. So we are supposing that $\text{E}[R] = 100$, $\sigma_R = 10$, and R is nonnegative. We want to compute $\Pr\{R \geq 200\}$.

We have already seen that Markov's Theorem 2.2 gives a coarse bound, namely,

$$\Pr\{R \geq 200\} \leq \frac{1}{2}.$$

Now we apply Chebyshev's Theorem to the same problem:

$$\Pr\{R \geq 200\} = \Pr\{|R - 100| \geq 100\} \leq \frac{\text{Var}[R]}{100^2} = \frac{10^2}{100^2} = \frac{1}{100}.$$

The purpose of the first step is to express the desired probability in the form required by Chebyshev's Theorem; the equality holds because R is nonnegative. Chebyshev's Theorem then yields the inequality.

So Chebyshev's Theorem implies that at most one person in a hundred has an I.Q. of 200 or more. We have gotten a much tighter bound using the additional information, namely the variance of R, than we could get knowing only the expectation.

5.2 A One-Sided Bound

Chebyshev's Theorem gives a "two-sided bound". That is, it bounds the probability that a random variable deviates *above or below* the mean by some amount. What if we want only a one-sided bound? For example, what is the probability that a random variable deviates *above* the mean by some amount?

This question is often answered incorrectly. The erroneous argument runs as follows. By Chebyshev's Theorem, R deviates above or below the mean by some amount with probability p. Therefore, R deviates above the mean by this amount with probability $p/2$, and R deviates below the mean by this amount with probability $p/2$.

While this argument is correct for a probability distribution function that is symmetric about the mean, it is not correct for random variables that are more likely to deviate above the mean than below. For example, in the I.Q. question, some people deviate 100 points above the mean; that is, there are people with I.Q. greater than 200. However, by assumption everyone has a positive I.Q.; no one deviates more than 100 points below the mean. For this reason it turns out we could actually improve the bound of Section 5.1 slightly—from 1 in 100 to 1 in 101. In general, there is a Chebyshev bound for the one-sided case that slightly improves our two-sided bound, but we don't need to go into it.

6 Deviation of Repeated Trials

Using Chebyshev's Theorem and the facts about variance and expectation, we are finally in a position to be show how the average of many trials approaches the mean.

6.1 Estimation from Repeated Trials

For example, suppose we want to estimate the fraction of the U.S. voting population who would favor Al Gore over George Bush in the year 2004 presidential election. Let p be this unknown fraction. Let's suppose we have some random process—say throwing darts at voter registration lists—which will select each voter with equal probability. Now we can define a Bernoulli variable, G, by the rule that $G = 1$ if a random voter most prefers Gore, and $G = 0$ otherwise. In this case, $G = G^2$, so

$$\mathrm{E}\left[G^2\right] = \mathrm{E}\left[G\right] = \Pr\left\{G = 1\right\} = p,$$

and

$$\mathrm{Var}\left[G\right] = \mathrm{E}\left[G^2\right] - \mathrm{E}^2\left[G\right] = p - p^2 = p(1-p).$$

To estimate p, we take a large number, n, of sample voters and count the fraction who favor Gore. We can describe this estimation as taking independent Bernoulli variables $G_1, G_2, \ldots, G_n$, each with the same expectation as G, computing their sum

$$S_n ::= \sum_{i=1}^{n} G_i, \tag{9}$$

and then using the average, S_n/n, as our estimate of p.

More generally, we can consider *any* set of random variables $G_1, G_2, \ldots, G_n$, with the same mean, μ, and likewise use the average, S_n/n, to estimate μ. One of the properties of S_n/n that is critical for this purpose is that S_n/n has the same expectation as the G_i's, namely,

$$\mathrm{E}\left[\frac{S_n}{n}\right] = \mu, \tag{10}$$

Proof.

$$\begin{aligned}
\mathrm{E}\left[\frac{S_n}{n}\right] &= \mathrm{E}\left[\frac{\sum_{i=1}^{n} G_i}{n}\right] && \text{(by def. (9) of } S_n\text{)} \\
&= \frac{\sum_{i=1}^{n} \mathrm{E}\left[G_i\right]}{n} && \text{(linearity of expectation)} \\
&= \frac{\sum_{i=1}^{n} \mu}{n} \\
&= \frac{n\mu}{n} = \mu.
\end{aligned}$$

□

Note that the random variables G_i need not be Bernoulli or even independent for (10) to hold, because linearity of expectation always holds.

Now suppose the G_i's also have the same deviation, σ. The second critical property of S_n/n is that

$$\mathrm{Var}\left[\frac{S_n}{n}\right] = \frac{\sigma^2}{n}. \tag{11}$$

This follows as long as the variance of S_n is the sum of the variances of the G_i's. For example, by Theorem 4.9, the variances can be summed if the G_i's are pairwise independent. Then we calculate:

$$\begin{aligned}
\operatorname{Var}\left[\frac{S_n}{n}\right] &= \frac{1}{n^2}\operatorname{Var}[S_n] && \text{(Theorem 4.7)}\\
&= \frac{1}{n^2}\operatorname{Var}\left[\sum_{i=1}^{n} G_i\right] && \text{(def (9) of } S_n\text{)}\\
&= \frac{1}{n^2}\sum_{i=1}^{n}\operatorname{Var}[G_i] && \text{(variances assumed to add)}\\
&= \frac{1}{n^2}\cdot n\sigma^2 = \frac{\sigma^2}{n}.
\end{aligned}$$

This is enough to apply Chebyshev's Bound and conclude:

Theorem 6.1. *[Pairwise Independent Sampling] Let*

$$S_n ::= \sum_{i=1}^{n} G_i$$

where $G_1, \ldots, G_n$ are pairwise independent variables with the same mean, μ, and deviation, σ. Then

$$\Pr\left\{\left|\frac{S_n}{n} - \mu\right| \geq x\right\} \leq \frac{1}{n}\left(\frac{\sigma}{x}\right)^2. \tag{12}$$

Proof.

$$\begin{aligned}
\Pr\left\{\left|\frac{S_n}{n} - \mu\right| \geq x\right\} &\leq \frac{\operatorname{Var}[S_n/n]}{x^2}. && \text{(Chebyshev's bound, Theorem 3.3)}\\
&= \frac{\sigma^2/n}{x^2} && \text{(by (11))}\\
&= \frac{1}{n}\left(\frac{\sigma}{x}\right)^2.
\end{aligned}$$

□

Theorem 6.1 finally provides a precise statement about how the average of independent samples of a random variable approaches the mean. It generalizes to many cases when S_n is the sum of independent variables whose mean and deviation are not necessarily all the same, though we shall not develop such generalizations here.

6.2 Birthdays again

We observed in lecture that the expected number of matching birthday pairs among n people was $\binom{n}{2}/365$. But how close to this expected value can we expect a typical sample to be? We can apply the Pairwise Independent Sampling Theorem to answer this question.

Now having matching birthdays for different pairs of students are not mutually independent events. For example, knowing that Alice and Bob have matching birthdays, and also that Ted and Alice have matching birthdays obviously implies that Bob and Ted have matching birthdays. On the other hand, knowing that Alice and Bob have matching birthdays tells us nothing about whether Alice and Carol have matching birthdays, *viz.*, these two events really are independent. We already already observed this phenomenon in Notes 11-12, §4.3.1, for the case of matching pairs among three coins. So even though the events that a pair of students have matching birthdays are not mutually independent, indeed not even three-way independent, they are *pairwise* independent.

This allows us to apply the Sampling Theorem. Let $B_1, B_2, \ldots, B_n$ be the birthdays of the n people, let $E_{i,j}$ be the indicator variable for the event $[B_i = B_j]$. For $i \neq j$, the probability that $B_i = B_j$ is $1/365$, so $\mathrm{E}[E_{i,j}] = 1/365$.

Now let M_n be the number of matching pairs, *i.e.*,

$$M_n ::= \sum_{1 \leq i < j \leq n} E_{i,j}. \tag{13}$$

So by linearity of expectation

$$\mathrm{E}[M_n] = \mathrm{E}\left[\sum_{1 \leq i < j \leq n} E_{i,j}\right] = \sum_{1 \leq i < j \leq n} \mathrm{E}[E_{i,j}] = \binom{n}{2}\frac{1}{365},$$

as we noted above. Also, by linearity of variance for pairwise independent variables

$$\mathrm{Var}[M_n] = \mathrm{Var}\left[\sum_{1 \leq i < j \leq n} E_{i,j}\right] = \sum_{1 \leq i < j \leq n} \mathrm{Var}[E_{i,j}] = \binom{n}{2}\frac{1}{365}\left(1 - \frac{1}{365}\right).$$

Now for our 6.042 class of 146 students, we have $\mathrm{E}[M_{146}] = 29$ and $\mathrm{Var}[M_{146}] = 29(1 - 1/365) < 29$. So by Theorem 6.1,

$$\Pr\{|M_{146} - 29| \geq x\} < \frac{29}{x^2}.$$

Letting $x = 8$, we conclude that there is a better than 50% chance that in a class of 146 students, the number of pairs of students with the same birthday will be between 21 and 37. In our class, we actually found that there were 17 matching pairs and 2 triples, for a total of 23 matching pairs.

6.3 Size of a Poll

Theorem 6.1 allows us to calculate poll size. How many people should we poll to get a reliable estimate of voters' preference?

Suppose, in particular, we want to know within tolerance $x ::= 0.02$ what fraction of the voters favor Gore. By choosing n large enough in Theorem 6.1 that we can reduce the probability that our estimate is off by more than x to as close to zero as we please.

For example, ninety-five per cent "confidence level" is a standard used in many statistical applications. So let's suppose we want our estimate of p to be within the tolerance 95% of the time, that is, with probability 0.95. Then we choose n so that $(1/n)(\sigma/x)^2 \leq 1 - 0.95$. That is, we want

$$n \geq \frac{\sigma^2}{(0.02)^2(1-0.95))} = \frac{p(1-p)}{0.00002} = 50,000p(1-p).$$

Solving for the sample size n in terms of the unknown p that we are trying to estimate in the first place may not seem to be making progress, but it's easy to see that the maximum value of $p(1-p)$ in the interval $0 \leq p \leq 1$ occurs at $p = 1/2$. So we conclude that if we sample

$$n \geq 50,000(1-1/2)1/2 = 12,500$$

voters, we can say that 95% of the time, our estimate $S_{12,500}/12,500$ will be within 0.02 of the fraction of voters who favor Gore.

Note that this bound on poll size holds regardless of how large the total voting population may be—whether we are trying to determine the preferences of a few tens of thousands of voters in a small city like Cambridge, or of the tens of millions of voters in a large nation like the U.S., the same poll size is adequate.

6.4 Confidence Levels

Now suppose a pollster dutifully checks with 12,500 randomly chosen voters and finds that 6,300 prefer Gore. It's tempting, but sloppy, to say that this means "With probability 0.95, the fraction, p, of voters who prefer Gore is $6300/12,500 = 0.504 \pm 0.02$."

What's objectionable about this statement is that it talks about the probability of a real world fact, namely the actual value of the fraction p. But p is what it is, and it simply makes no sense to talk about the probability that it is something else. For example, suppose p is actually 0.53; then it's nonsense to ask about the probability that it is within 0.02 of 0.504—it simply isn't.

A more careful summary of what we have accomplished goes this way: we have described a probabilistic procedure for estimating the actual value of the fraction p. The probability that *our estimation procedure* will yield a value within 0.02 of p is 0.95. This is a bit of a mouthful, so special phrasing closer to the sloppy language is commonly used. The pollster would describe his conclusion by saying that "At the 95% *confidence level*, the fraction of voters who prefer Gore is 0.504 ± 0.02."

Actually, polling 12,500 voters is excessive. We derived this bound on poll size solely by applying Chebyshev's Theorem to the value of the variance of S_n/n. But in fact we know the exact distribution of S_n, namely, it has a binomial distribution with parameters n, p. In the next section we do a more detailed calculation of probabilities of deviation from the mean specifically for the binomial distribution; we can show that the poll size need only be about $1/5$ of the size derived from the Chebyshev bound.

6.5 Better Polling

Let ϵ be the acceptable error tolerance of our poll. In the previous section we chose $\epsilon = 0.02$. We can define δ, the probability that our poll is off by more than ϵ as follows:

$$\begin{aligned}\delta \quad ::= \quad & \underbrace{\Pr\left\{\frac{S_n}{n} < p - \epsilon\right\}}_{\substack{\text{too many in sample}\\ \text{say ``Bush''}}} \quad + \quad \underbrace{\Pr\left\{\frac{S_n}{n} > p + \epsilon\right\}}_{\substack{\text{too many in sample}\\ \text{say ``Gore''}}} \\ = \quad & \Pr\{S_n < (p-\epsilon)n\} + \Pr\{S_n > (p+\epsilon)n\}.\end{aligned}$$

Since S_n has the binomial distribution with parameters n and p, the two terms in the definition of δ can be bounded using the bound (14) on $F_{n,p}$ from Notes 10:

Lemma.

$$F_{n,p}(\alpha n) \leq \left(\frac{1-\alpha}{1-\alpha/p}\right) f_{n,p}(\alpha n) \tag{14}$$

for $\alpha < p$.

To ensure that $\alpha < p$, we observe that

$$\Pr\left\{\frac{S_n}{n} > p + \epsilon\right\} = \Pr\left\{\frac{n - S_n}{n} < 1 - p - \epsilon\right\},$$

where $(n - S_n)/n$ is the fraction of people polled who say that they prefer Bush, and $1 - p$ is the fraction of all Americans who prefer Bush. This gives

$$\delta \leq F_{n,p}((p-\epsilon)n) + F_{n,1-p}((1-p-\epsilon)n). \tag{15}$$

As in the previous section, the bound (15) contains p, the fraction of Americans that favor Gore, which is the number we are trying to determine by polling. But as before, the worst case for the bound is when $p = 1/2$, though we shall not prove this. So we get

$$\delta \leq F_{n,\frac{1}{2}}((\frac{1}{2}-\epsilon)n) + F_{n,1-\frac{1}{2}}((1-\frac{1}{2}-\epsilon)n) = 2F_{n,\frac{1}{2}}((\frac{1}{2}-\epsilon)n). \tag{16}$$

Now plugging in $\epsilon = 0.02$ into (16) gives:

$$\begin{aligned}\delta \leq 2F_{n,\frac{1}{2}}(0.48n) &\leq 2 \cdot \frac{1-\alpha}{1-2\alpha} f_{n,1/2}(0.48n) \\ &\approx 2 \cdot 13 \cdot 2^{-n(1-H(0.48))} / \sqrt{2\pi \cdot 0.48(1-0.48)n} \\ &= 26 \cdot \frac{2^{-0.00115n}}{1.2523\sqrt{n}}.\end{aligned}$$

We want to poll enough people so that δ is less than 0.05. The easiest way is to plug in values for n, the number of people polled:

n = people polled	upper bound on probability poll is wrong
1000	29.4%
2000	9.3%
3000	3.4%
2500	5.6%
2750	4.4%
2616	5.004%
2617	4.999%

So polling 2617 people is sufficient to determine public opinion to within 2% with confidence of 95%. Again, the remarkable point is that the population of the country has no effect on the poll size. Whether there are ten thousand people or a billion in a country, polling 2617 people is sufficient!

Here we got a much better estimate of probable deviation from the mean using the fact that the samples were independent—and hence that the sampling distribution was binomial—than in the previous section using the Pairwise Independent Sampling Theorem 6.1. This should not be surprising, since the Sampling Theorem is based on Chebyshev's bound, and we've already seen that the Chebyshev bound can be much weaker than bounds derived using more information about a density function than simply its variance.

However, there are situations—matching birthdays is a good example—where mutual independence of the samples doesn't hold, but pairwise independence does, and that's where the Pairwise Independent Sampling Theorem becomes our main handle on predicting sample deviations.

7 Proof of the Weak Law

An equivalent way to state the conclusion of the Weak Law of Large Numbers, Theorem 2.1, is that the probability that the average *differs* from the expectation by more than any given tolerance approaches zero.

Theorem 7.1. *[Weak Law of Large Numbers] Let*

$$S_n ::= \sum_{i=1}^{n} G_i,$$

where $G_1, \dots, G_n, \dots$ are pairwise independent variables with the same expectation, μ and standard deviation, σ. For any $\epsilon > 0$,

$$\lim_{n\to\infty} \Pr\left\{ \left| \frac{S_n}{n} - \mu \right| \geq \epsilon \right\} = 0.$$

Proof. Choose x in Theorem 6.1 to be ϵ. Then, given any $\delta > 0$, choose n large enough to make $(\sigma/x)^2/n < \delta$. By Theorem 6.1,

$$\Pr\left\{\left|\frac{S_n}{n} - \mu\right| \geq \epsilon\right\} < \delta.$$

So the limiting probability must equal zero. □

Notice that this version of the Weak Law is slightly different from the version we first stated in Theorem 2.1. Theorem 7.1 requires that the G_i's have finite variance but Theorem 2.1 only requires finite expectation. On the other hand, the original version 2.1 requires mutual independence, while Theorem 7.1 requires only pairwise independence. The case when the variance may be infinite is not important to us, and we will not try to prove it.

A weakness of both the Weak Law as well as our Pairwise Independence Sampling Theorem 6.1 is that neither provides any information about the way the average value of the observations may be expected to *oscillate* in the course of repeated experiments. In later Notes we will briefly consider a *Strong* Law of Large Numbers which deals with the oscillations. Such oscillations may not be important in our example of polling about Gore's popularity or of birthday matches, but they are critical in gambling situations, where large oscillations can bankrupt a player, even though the player's average winnings are assured in the long run. As the famous economist Keynes is alleged to have remarked, the problem is that "In the long run, we are all dead."

8 Random Walks and Gamblers' Ruin

Random Walks nicely model many natural phenomena in which a person, or particle, or process takes steps in a randomly chosen sequence of directions. For example in Physics, three-dimensional random walks are used to model Brownian motion and gas diffusion. In Computer Science, the Google search engine uses random walks through the graph of world-wide web links to determine the relative importance of websites. In Finance Theory, there is continuing debate about the degree to which one-dimensional random walks can explain the moment-to-moment or day-to-day fluctuations of market prices. In these Notes we consider 1-dimensional random walks: walks along a straight line. Our knowledge of expectation and deviation will make 1-dimensional walks easy to analyze, but even these simple walks exhibit probabilistic behavior that can be astonishing.

In the Mathematical literature, random walks are for some reason traditionally discussed in the context of some social vice. A one-dimensional random walk is often described as the path of a drunkard who randomly staggers left or right at each step. In the rest of these Notes, we examine one-dimensional random walks using the language of gambling. In this case, a position during the walk is a gambler's cash-on-hand or *capital*, and steps on the walk are bets whose random outcomes increase or decrease his capital. We will be interested in two main questions:

1. What is the probability that the gambler wins?

2. How long must the gambler expect to wait for the walk to end?

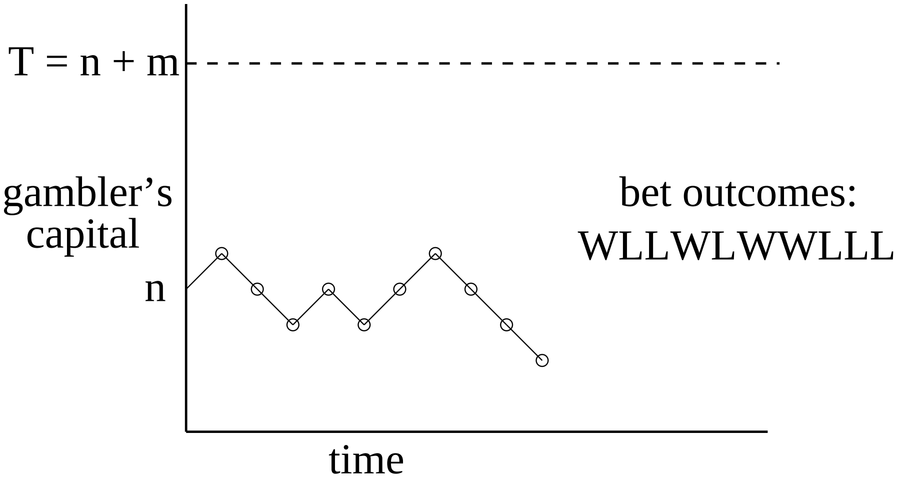


Figure 2: *This is a graph of the gambler's capital versus time for one possible sequence of bet outcomes. At each time step, the graph goes up with probability* p *and down with probability* $1-p$*. The gambler continues betting until the graph reaches either 0 or* $T = n + m$*.*

In particular, we suppose a gambler starts with n dollars. He makes a sequence of \$1 bets. If he wins an individual bet, he gets his money back plus another \$1. If he loses, he loses the \$1. In each bet, he wins with probability $p > 0$ and loses with probability $q ::= 1 - p > 0$. The gambler plays until either he is bankrupt or increases his capital to a goal amount of T dollars. If he reaches his goal, then he is called an overall *winner*, and his *profit* will be $m ::= T - n$ dollars. If his capital reaches zero dollars before reaching his gaol, then we say that he is "ruined" or *goes broke*.

The gambler's situation as he proceeds with his \$1 bets is illustrated in Figure 2. The random walk has boundaries at 0 and T. If the random walk ever reaches either of these boundary values, then it terminates. We want to determine the probability, w, that the walk terminates at boundary T, namely, the probability that the gambler is a winner.

In a fair game, $p = q = 1/2$. The corresponding random walk is called *unbiased*. The gambler is more likely to win if $p > 1/2$ and less likely to win if $p < 1/2$; the corresponding random walks are called *biased*.

Example 8.1. Suppose that the gambler is flipping a coin, winning \$1 on Heads and losing \$1 on Tails. Also, the gambler's starting capital is $n = 500$ dollars, and he wants to make $m = 100$ dollars. That is, he plays until he goes broke or reaches a goal of $T = n + m = \$600$. What is the probability that he is a winner? We will show that in this case the probability $w = 5/6$. So his chances of winning are really very good, namely, 5 chances out of 6.

Now suppose instead, that the gambler chooses to play roulette in an American casino, always betting \$1 on red. A roulette wheel has 18 black numbers, 18 red numbers, and 2 green numbers. In this game, the probability of winning a single bet is $p = 18/38 \approx 0.47$. It's the two green numbers that slightly bias the bets and give the casino an edge. Still, the bets are almost fair, and you might expect that the gambler has a reasonable chance of reaching his goal—the 5/6 probability of winning in the unbiased game surely gets reduced, but perhaps not too drastically. Not so! His odds of winning against the "slightly" unfair roulette wheel are less than 1 in 37,000. If that seems surprising, listen to this: *no matter how much money* the gambler has to start, *e.g.*, \$5000, \$50,000, $\$5 \cdot 10^{12}$, his odds are still less than 1 in 37,000 of winning a mere 100 dollars!

Moral: Don't play!

The theory of random walks is filled with such fascinating and counter-intuitive conclusions.

9 The Probability Space

Each random-walk game corresponds to a path like the one in Figure 2 that starts at the point $(n, 0)$. A winning path never touches the x axis and ends when it first touches the line $y = T$. Likewise, a losing path never touches the line $y = T$ and ends when it first touches the x axis.

Any length k path can be characterized by the history of wins and losses on individual $1 bets, so we use a length k string of W's and L's to model a path, and assign probability $p^r q^{k-r}$ to a string that contains r W's. The *outcomes* in our sample space will be precisely those string corresponding to winning or losing walks.

What about the infinite walks in which the gambler plays forever, neither reaching his goal nor going bankrupt? We saw in an in-class problem that the probability of playing forever is zero, so we don't need to include any such outcomes in our sample space.

As a sanity check on this definition of the probability space, we should verify that the sum of the outcome probabilities is one, but we omit this calculation.

10 The Probability of Winning

10.1 The Unbiased Game

Let's begin by considering the case of a fair coin, that is, $p = 1/2$, and determine the probability, w, that the gambler wins. We can handle this case by considering the expectation of the random variable G equal to the gambler's dollar gain. That is, $G = T - n$ if the gambler wins, and $G = -n$ if the gambler loses, so

$$\mathrm{E}[G] = w(T - n) - (1 - w)n = wT - n.$$

Notice that we're using the fact that the only outcomes are those in which the gambler wins or loses—there are no infinite games—so the probability of losing is $1 - w$.

Now let G_i be the amount the gambler gains on the ith flip: $G_i = 1$ if the gambler wins the flip, $G_i = -1$ if the gambler loses the flip, and $G_i = 0$ if the game has ended before the ith flip. Since the coin is fair, $\mathrm{E}[G_i] = 0$.

The random variable G is the sum of all the G_i's, so by linearity of expectation[2]

$$wT - n = E(G) = \sum_{i=1}^{\infty} E(G_i) = 0,$$

which proves

Theorem 10.1. *In the unbiased Gambler's Ruin game with probability $p = 1/2$ of winning each individual bet, with initial capital, n, and goal, T,*

$$\Pr\{\textit{the gambler is a winner}\} = \frac{n}{T}. \tag{17}$$

Example 10.2. Suppose we have \$100 and we start flipping a fair coin, betting \$1 with the aim of winning \$100. Then the probability of reaching the \$200 goal is $100/200 = 1/2$—the same as the probability of going bankrupt. In general, if $T = 2n$, then the probability of doubling your money or losing all your money is the same. This is about what we would expect.

Example 10.3. Suppose we have \$500 and we start flipping a fair coin, betting \$1 with the aim of winning \$100. So $n = 500, T = 600$, and $\Pr\{\text{win}\} = 500/600 = 5/6$, as we claimed at the outset.

Example 10.4. Suppose Albert starts with \$100, and Radhi starts with \$10. They flip a fair coin, and every time a Head appears, Albert wins \$1 from Radhi, and vice versa for Tails. They play this game until one person goes bankrupt. What is the probability of Albert winning?

This problem is identical to the Gambler's Ruin problem with $n = 100$ and $T = 100+10 = 110$. The probability of Albert winning is $100/110 = 10/11$, namely, the ratio of his wealth to the combined wealth. Radhi's chances of winnning are $1/11$.

Note that although Albert will win most of the time, the game is still fair. When Albert wins, he only wins \$10; when he loses, he loses big: \$100. Albert's—and Radhi's—expected win is zero dollars.

Another intuitive idea is confirmed by this analysis: the larger the gambler's initial stake, the larger the probability that he will win a fixed amount.

Example 10.5. If the gambler started with one million dollars instead of 500, but aimed to win the same 100 dollars as in the Example 10.3, the probability of winning would increase to $1M/(1M + 100) > .9999$.

[2] We've been stung by paradoxes in this kind of situation, so we should be careful to check that the condition for infinite linearity of expectation is satisfied. Namely, we have to check that $\sum_{i=1}^{\infty} \mathrm{E}[|G_i|]$ converges.

In this case, $|G_i| = 1$ iff the walk is of length at least i, and $|G_i| = 0$ otherwise. So

$$\mathrm{E}[|G_i|] = \Pr\{\text{the walk is of length } \geq i\}.$$

But we show in an in-class problem that there is a constant $r < 1$ such that

$$\Pr\{\text{the walk is of length } \geq i\} \leq \Theta(r^i).$$

So the $\sum_{i=1}^{\infty} \mathrm{E}[|G_i|]$ is bounded term-by-term by a convergent geometric series, and therefore it also converges.

10.2 A Recurrence for the Probability of Winning

To handle the case of a biased game we need a more general approach. We consider the probability of the gambler winning as a function of his initial capital. That is, let p and T be fixed, and let w_n be the gambler's probabiliity of winning when his initial capital is n dollars. For example, w_0 is the probability that the gambler will win given that he starts off broke; clearly, $w_0 = 0$. Likewise, $w_T = 1$.

Otherwise, the gambler starts with n dollars, where $0 < n < T$. Consider the outcome of his first bet. The gambler wins the first bet with probability p. In this case, he is left with $n + 1$ dollars and becomes a winner with probability w_{n+1}. On the other hand, he loses the first bet with probability $1 - p$. Now he is left with $n - 1$ dollars and becomes a winner with probability w_{n-1}. Overall, he is a winner with probability $w_n = pw_{n+1} + qw_{n-1}$. Solving for w_{n+1} we have

$$w_{n+1} = \frac{w_n}{p} - w_{n-1}\frac{q}{p}. \tag{18}$$

This kind of inductive definition of a quantity w_{n+1} in terms of a linear combination of values w_k for $k < n+1$ is called a *homogeneous linear recurrence*. There is a simple general method for solving such recurrences which we now illustrate. The method is based on a guess that the form of the solution is $w_n = c^n$ for some $c > 0$. It's not obvious why this is a good guess, but we now show how to find the constant c and verify the guess.

Namely, from (18) we have

$$w_{n+1} - \frac{w_n}{p} + w_{n-1}\frac{q}{p} = 0. \tag{19}$$

If our guess is right, then this is equivalent to

$$c^{n+1} - \frac{c^n}{p} + c^{n-1}\frac{q}{p} = 0.$$

Now factoring out c^{n-1} gives

$$c^2 - \frac{c}{p} + \frac{q}{p} = 0.$$

Solving this quadratic equation in c yields two roots, $(1 - p)/p$ and 1. So if we define $w_n ::= ((1 - p)/p)^n = (q/p)^n$, then (19), and hence (18) is satisifed. We can also define $w_n ::= 1^n$ and satisfy (19). Since the lefthand side of (19) is zero using either definition, it follows that any definition of the form

$$w_n ::= A\left(\frac{q}{p}\right)^n + B \cdot 1^n$$

will also satisfy (19). Now our boundary conditions, namely the values of w_0 and w_T, let us solve for A and B:

$$\begin{aligned} 0 &= w_0 = A + B, \\ 1 &= w_T = A\left(\frac{q}{p}\right)^T + B, \end{aligned}$$

so

$$A = \frac{1}{(q/p)^T - 1}, \qquad B = -A, \tag{20}$$

and therefore

$$w_n = \frac{(q/p)^n - 1}{(q/p)^T - 1}. \tag{21}$$

Now we could verify our guess work and prove (21) by a routine induction on n which we omit.

The solution (21) only applies to biased walks, since we require $p \neq q$ so the denominator is not zero. That's ok, since we already worked out that the case when $p = q$ in Theorem 10.1. So we have shown:

Theorem 10.6. *In the biased Gambler's Ruin game with probability, $p \neq 1/2$, of winning each bet, with initial capital, n, and goal, T,*

$$\Pr\{\textit{the gambler is a winner}\} = \frac{(q/p)^n - 1}{(q/p)^T - 1}. \tag{22}$$

The expression (22) for the probability that the Gambler wins in the biased game is a little hard to interpret. There is a simpler upper bound which is nearly tight when the gambler's starting capital is large.

Suppose that $p < 1/2$; that is, the game is biased *against* the gambler. Then both the numerator and denominator in the quotient in (22) are positive, and the quotient is less than one. So adding 1 to both the numerator and denominator increases the quotient[3], and the bound (22) simplifies to $(q/p)^n/(q/p)^T = (p/q)^{T-n}$, which proves

Corollary 10.7. *In the Gambler's Ruin game biased against the Gambler, that is, with probability $p < 1/2$ of winning each bet, with initial capital, n, and goal, T,*

$$\Pr\{\textit{the gambler is a winner}\} < \left(\frac{p}{q}\right)^m, \tag{23}$$

where $m ::= T - n$.

The amount $m = T - n$ is called the Gambler's *intended profit*. So the gambler gains his intended profit, m, before going broke with probability at most $(p/q)^m$. Notice that this upper bound does not depend on the gambler's starting capital, but only on his intended profit. The consequences of this are amazing:

[3] If $0 < a < b$, then

$$\frac{a}{b} < \frac{a+1}{b+1},$$

because

$$\frac{a}{b} = \frac{a(1+1/b)}{b(1+1/b)} = \frac{a + a/b}{b+1} < \frac{a+1}{b+1}.$$

Example 10.8. Suppose that the gambler starts with \$500 aiming to profit \$100, this time by making \$1 bets on red in roulette. By (23), the probability, w_n, that he is a winner is less than

$$\left(\frac{18/38}{20/38}\right)^{100} = \left(\frac{9}{10}\right)^{100} < \frac{1}{37,648}.$$

This is a dramatic contrast to the unbiased game, where we saw in Example 10.3 that his probability of winning was $5/6$.

Example 10.9. We also observed that with \$1,000,000 to start in the unbiased game, he was almost certain to win \$100. But betting against the "slightly" unfair roulette wheel, even starting with \$1,000,000, his chance of winning \$100 remains less than 1 in 37,648! He will almost surely lose all his \$1,000,000. In fact, because the bound (23) depends only on his intended profit, his chance of going up a mere \$100 is less than 1 in 37,648 *no matter how much money he starts with*!

The bound (23) is exponential in m. So, for example, doubling his intended profit will square his probability of winning.

Example 10.10. The probability that the gambler's stake goes up 200 dollars before he goes broke playing roulette is at most

$$(9/10)^{200} = ((9/10)^{100})^2 = \left(\frac{1}{37,648}\right)^2,$$

which is about 1 in 70 billion.

The odds of winning a little money are not so bad.

Example 10.11. Applying the exact formula (22), we find that the probability of winning \$10 before losing \$10 is

$$\frac{\left(\frac{20/38}{18/38}\right)^{10} - 1}{\left(\frac{20/38}{18/38}\right)^{20} - 1} = 0.2585\ldots$$

This is somewhat worse than the 1 in 2 chance in the fair game, but not dramatically so.

Thus, in the fair case, it helps a lot to have a large bankroll, whereas in the unfair case, it doesn't help much.

10.3 Intuition

Why is the gambler so unlikely to make money when the game is slightly biased against him? Intuitively, there are two forces at work. First, the gambler's capital has random upward and downward *swings* due to runs of good and bad luck. Second, the gambler's capital will have a steady, downward *drift*, because he has a small, negative expected return on every bet. The situation is shown in Figure 3.

For example, in roulette the gambler wins a dollar with probability $9/19$ and loses a dollar with probability $10/19$. Therefore, his expected return on each bet is $9/10 - 10/19 = -1/19 \approx -0.053$ dollars. That is, on each bet his capital is expect to drift downward by a little over 5 cents.

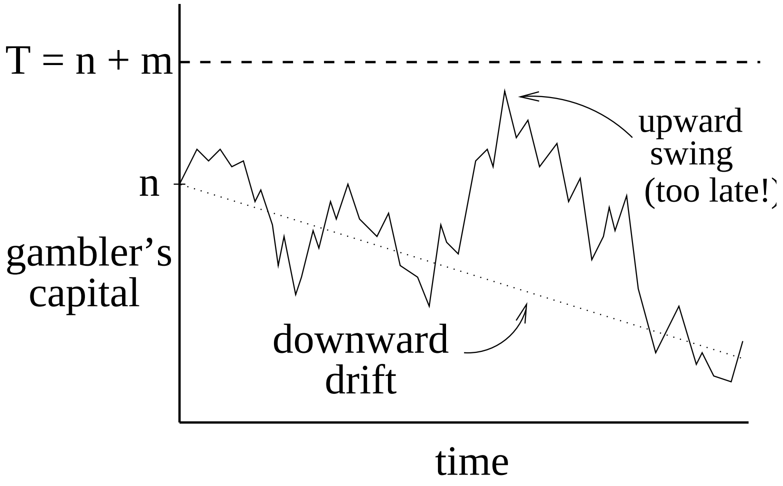


Figure 3: *In an unfair game, the gambler's capital swings randomly up and down, but steadily drifts downward. If the gambler does not have a winning swing early on, then his capital drifts downward, and later upward swings are insufficient to make him a winner.*

Our intuition is that if the gambler starts with a trillion dollars, then he will play for a very long time, so at some point there should be a lucky, upward swing that puts him \$100 ahead. The problem is that his capital is steadily drifting downward. If the gambler does not have a lucky, upward swing early on, then he is doomed. After his capital drifts downward a few hundred dollars, he needs a huge upward swing to save himself. And such a huge swing is extremely improbable. As a rule of thumb, *drift dominates swings* in the long term.

We can quantify these drifts and swings. After k rounds, the number of wins by our player has a binomial distribution with parameters $p < 1/2$ and k. His expected win on any single bet is $p - q = 2p - 1$ dollars, so his expected capital is $n - k(1 - 2p)$. Now to be a winner, his actual number of wins must exceed the expected number by $m + k(1 - 2p)$. But we saw before that the binomial distribution has a standard deviation of only $\sqrt{kp(1-p)}$. So for the gambler to win, he needs his number of wins to deviate by

$$\frac{m + k(1-2p)}{\sqrt{kp(1-2p)}} = \Theta(\sqrt{k})$$

times its standard deviation. In our study of binomial tails we saw that this was extremely unlikely.

In a fair game, there is no drift; swings are the only effect. In the absence of downward drift, our earlier intuition is correct. If the gambler starts with a trillion dollars then almost certainly there will eventually be a lucky swing that puts him \$100 ahead.

If we start with \$10 and play to win only \$10 more, then the difference between the fair and unfair games is relatively small. We saw that the probability of winning is $1/2$ versus about $1/4$. Since swings of \$10 are relatively common, the game usually ends before the gambler's capital can drift very far. That is, the game does not last long enough for drift to dominate the swings.

11 How Long a Walk?

Now that we know the probability, w_n, that the gambler is a winner in both fair and unfair games, we consider how many bets he needs on average to either win or go broke.

11.1 Duration of an Biased Walk

Let Q be the number of bets the gambler makes until the game ends. Since the gambler's expected win on any bet is $2p-1$, Wald's Theorem should tell us that his game winnings, G, will have expectation $\mathrm{E}[Q](2p-1)$. That is,

$$\mathrm{E}[G] = (2p-1)\,\mathrm{E}[Q], \tag{24}$$

In an unbiased game (24) is trivially true because both $2p-1$ and the expected overall winnings, $\mathrm{E}[G]$, are zero. On the other hand, in the unfair case, $2p-1 \neq 0$. Also, we know that

$$\mathrm{E}[G] = w_n(T-n) - (1-w_n)n = w_nT - n.$$

So assuming (24), we conclude

Theorem 11.1. *In the biased Gambler's Ruin game with initial capital, n, goal, T, and probability, $p \neq 1/2$, of winning each bet,*

$$\mathrm{E}[\textit{number of bets till game ends}] = \frac{\Pr\{\textit{gambler is a winner}\}\,T - n}{2p-1}. \tag{25}$$

The only problem is that (24) is not a special case of Wald's Theorem because $G = \sum_{i=1}^{Q} G_i$ is not a sum of *nonnegative* variables: when the gambler loses the ith bet, the random variable G_i equals -1. However, this is easily dealt with.[4]

Example 11.2. If the gambler aims to profit \$100 playing roulette with n dollars to start, he can expect to make $((n+100)/37,648-n)/(2(18/38)-1) \approx 19n$ bets before the game ends. So he can enjoy playing for a good while before almost surely going broke.

[4] The random variable G_i+1 is nonnegative, and $\mathrm{E}[G_i+1 \mid Q \geq i] = \mathrm{E}[G_i \mid Q \geq i]+1 = 2p$, so by Wald's Theorem

$$\mathrm{E}\left[\sum_{i=1}^{Q}(G_i+1)\right] = 2p\,\mathrm{E}[Q]. \tag{26}$$

But

$$\begin{aligned}
\mathrm{E}\left[\sum_{i=1}^{Q}(G_i+1)\right] &= \mathrm{E}\left[\sum_{i=1}^{Q}G_i + \sum_{i=1}^{Q}1\right] \\
&= \mathrm{E}\left[(\sum_{i=1}^{Q}G_i) + Q\right] \\
&= \mathrm{E}\left[\sum_{i=1}^{Q}G_i\right] + \mathrm{E}[Q] \\
&= \mathrm{E}[G] + \mathrm{E}[Q]. \qquad (27)
\end{aligned}$$

Now combining (26) and (27) confirms the truth of our assumption (24).

11.2 Duration of an Unbiased Walk

This time, we need the more general approach of recurrences to handle the unbiased case. We consider the expected number of bets as a function of the gambler's initial capital. That is, for fixed p and T, let e_n be the expected number of bets until the game ends when the gambler's initial capital is n dollars. Since the game is over in no steps if $n = 0$ or T, the boundary conditions this time are $e_0 = e_T = 0$.

Otherwise, the gambler starts with n dollars, where $0 < n < T$. Now by the conditional expectation rule, the expected number of steps can be broken down into the expected number of steps given the outcome of the first bet weighted by the probability of that outcome. That is,

$$e_n = p \operatorname{E}[Q \mid \text{gambler wins first bet}] + q \operatorname{E}[Q \mid \text{gambler loses first bet}].$$

But after the gambler wins the first bet, his capital is $n + 1$, so he can expect to make another e_{n+1} bets. That is,

$$\operatorname{E}[Q \mid \text{gambler wins first bet}] = 1 + e_{n+1},$$

and similarly,

$$\operatorname{E}[Q \mid \text{gambler loses first bet}] = 1 + e_{n-1}.$$

So we have

$$e_n = p(1 + e_{n+1}) + q(1 + e_{n-1}) = pe_{n+1} + qe_{n-1} + 1,$$

which yields the linear recurrence

$$e_{n+1} = \frac{e_n}{p} - \frac{q}{p}e_{n-1} - \frac{1}{p}.$$

For $p = q = 1/2$, this equation simplifies to

$$e_{n+1} = 2e_n - e_{n-1} - 2. \tag{28}$$

There is a general theory for solving linear recurrences like (28) in which the value at $n + 1$ is a linear combination of values at some arguments $k < n + 1$ plus another simple term—in this case plus the constant -2. This theory implies that

$$e_n = (T - n)n. \tag{29}$$

Fortunately, we don't need the general theory to *verify* this solution. Equation (29) can be verified routinely from the boundary conditions and (28) using strong induction on n.

So we have shown

Theorem 11.3. *In the unbiased Gambler's Ruin game with initial capital, n, and goal, T, and probability, $p = 1/2$, of winning each bet,*

$$\operatorname{E}[\textit{number of bets till game ends}] = n(T - n). \tag{30}$$

Another way to phrase Theorem 11.3 is

$$\text{E}\left[\text{number of bets till game ends}\right] = \text{initial capital} \cdot \text{intended profit}. \tag{31}$$

Now for example, we can conclude that if the gambler starts with \$10 dollars and plays until he is broke or ahead \$10, then $10 \cdot 10 = 100$ bets are required on average. If he starts with \$500 and plays until he is broke or ahead \$100, then the expected number of bets until the game is over is $500 \times 100 = 50,000$.

Notice that (31) is a very simple answer that cries out for an intuitive proof, but we have not found one.

12 Quit While You Are Ahead

Suppose that the gambler never quits while he is ahead. That is, he starts with $n > 0$ dollars, ignores any goal T, but plays until he is flat broke. Then it turns out that if the game is not favorable, *i.e.*, $p \leq 1/2$, the gambler is sure to go broke. In particular, he is even sure to go broke in a "fair" game with $p = 1/2$. [5]

Lemma 12.1. *If the gambler starts with one or more dollars and plays a fair game until he is broke, then he will go broke with probability 1.*

Proof. If the gambler has initial capital n and goes broke in a game without reaching a goal T, then he would also go broke if he were playing and ignored the goal. So the probability that he will lose if he keeps playing without stopping at any goal T must be at least as large as the probability that he loses when he has a goal $T > n$.

But we know that in a fair game, the probability that he loses is $1 - n/T$. This number can be made arbitrarily close to 1 by choosing a sufficiently large value of T. Hence, the probability of his losing while playing without any goal has a lower bound arbitrarily close to 1, which means it must in fact be 1. □

So even if the gambler starts with a million dollars and plays a perfectly fair game, he will eventually lose it all with probability 1. In fact, if the game is unfavorable, then Theorem 11.1 and Corollary 10.7 imply that his expected time to go broke is essentially proportional to his initial capital, *i.e.*, $\Theta(n)$.

But there is good news: if the game is fair, he can "expect" to play for a very long time before going broke; in fact, he can expect to play forever!

Lemma 12.2. *If the gambler starts with one or more dollars and plays a fair game until he goes broke, then his expected number of plays is infinite.*

Proof. Consider the gambler's ruin game where the gambler starts with initial capital n, and let u_n be the expected number of bets for the *unbounded* game to end. Also, choose any $T \geq n$, and as above, let e_n be the expected number of bets for the game to end when the gambler's goal is T.

[5] If the game is favorable to the gambler, *i.e.*, $p > 1/2$, then we could show that there is a positive probability that the gambler will play forever, but we won't examine this case in these Notes.

The unbounded game will have a larger expected number of bets compared to the bounded game because, in addition to the possibility that the gambler goes broke, in the bounded game there is also the possibility that the game will end when the gambler reaches his goal, T. That is,

$$u_n \geq e_n.$$

So by (29),

$$u_n \geq n(T-n).$$

But $n \geq 1$, and T can be any number greater than or equal to n, so this lower bound on u_n can be arbitrarily large. This implies that u_n must be infinite.

Now by Lemma 12.1, with probability 1, the unbounded game ends when the gambler goes broke. So the expected time for the unbounded game to *end* is the *same* as the expected time for the gambler to *go broke*. Therefore, the expected time to go broke is infinite. □

In particular, even if the gambler starts with just one dollar, his expected number of plays before going broke is infinite! Of course, this does not mean that it is likely he will play for long. For example, there is a 50% chance he will lose the very first bet and go broke right away.

Lemma 12.2 says that the gambler can "expect" to play forever, while Lemma 12.1 says that with probability 1 he will go broke. These Lemmas sound contradictory, but our analysis showed that they are not.

13 Infinite Expectation

So what are we to make of such a random variable with infinite expectation? For example, suppose we repeated the experiment of having the gambler make fair bets with initial stake one dollar until he went broke, and we kept a record of the average number of bets per experiment. Our theorems about deviation from the mean only apply to random variables with finite expectation, so they don't seem relevant to this situation. But in fact they are.

For example, let Q be the number of bets required for the gambler to go broke in a fair game starting with one dollar. We could use some of our combinatorial techniques to show that

$$\Pr\{Q=m\} = \Theta(m^{-3/2}). \tag{32}$$

This implies that

$$\mathrm{E}[Q] = \Theta\left(\sum_{m=1}^{\infty} m \cdot m^{-3/2}\right) = \Theta\left(\sum_{m=1}^{\infty} m^{-1/2}\right).$$

We know this last series is divergent, so we have another proof that Q has infinite expectation.

But suppose we let $R ::= Q^{1/5}$. Then the estimate (32) also lets us conclude that

$$\mathrm{E}[R] = \Theta\left(\sum_{m=1}^{\infty} m^{-13/10}\right)$$

and

$$\mathrm{E}\left[R^2\right] = \Theta\left(\sum_{m=1}^{\infty} m^{-11/10}\right).$$

Since both these series are convergent, we can conclude that Var $[R]$ is finite. Now our theorems about deviation can be applied to tell us that the average *fifth root* of the number of bets to go broke is very likely to converge to a finite expected value.

We won't go further into the details, but the moral of this discussion is that our results about deviation from a finite mean can still be applied to natural models like random walks where variables with infinite expectation may play an important role.

14 The Chernoff Bound

The Chernoff bound applies to a sum of independent random variables that satisfy conditions that lie between the conditions needed for the Pairwise Independent Sampling Theorem of Notes 11-12 and conditions that imply the sum has a binomial distribution. When it applies, the Chernoff bound gives nearly as good a bound as our estimates for the binomial distribution in Notes 11-12. In particular, the Chernoff bound is exponentially smaller than bound given by the Pairwise Independent Sampling Theorem.

The Chernoff bound plays a larger role in Computer Science than the more traditional Central Limit Theorem (which will be briefly considered in later Notes). Both theorems give bounds on deviation from the mean, but the Chernoff bound gives better estimates on the probability of deviating from the mean by many standard deviations.

For example, suppose we are designing a system whose components may occasionally fail, but we want the system as a whole to be very reliable. The Chernoff bound can provide good estimates for the number of failures the system should be designed to survive in order to meet the specified high level of reliability. That is, the system will only fail only if a the number of component failures exceeds a designated threshold, but the Chernoff bound tells us that this threshold is very unlikely to be exceeded.

Another typical application is in designing probabilistic algorithms. We expect that such algorithms might give a wrong answer, but will do so only if the number of mistaken probabilistic "guesses" it makes is much larger than should be expected. The likelihood of this unusually large number of mistakes can often be estimated well using the Chernoff bound.

15 The Probability of at Least One Event

Let $A_1, A_2, \ldots, A_n$ be a sequence of events, and let T be the number of these events that occur. What is $\Pr\{T \geq 1\}$, the probability that at least 1 event occurs? Note that the event $[T \geq 1]$ is precisely the same as the event $\bigcup A_i$. In Notes 10, §5.2, and in Class Problems 10W, Problem 1, we verified the general bounds

$$\max_{1 \leq i \leq n} \Pr\{A_i\} \leq \bigcup A_i \leq \sum_{i=1}^{n} \Pr\{A_i\}, \tag{33}$$

and described situations in which each of these bounds were achieved. So in general, we cannot improve the bounds given in (33).

On the other hand, if the events A_i are mutually independent, we can be much more precise about the probability that one or more of them occur. In fact, we will show that if we expect several events to occur, then almost certainly at least one event will occur. Another way to say this is that if we expect more than one event to occur, then the probability that no event occurs is practically zero. Specifically, we have:

Theorem 15.1. *Let $A_1, A_2, \ldots A_n$ be independent events, and let T be the number of these events that occur. The probability that none of the events occur is at most $e^{-\operatorname{E}[T]}$.*

Interestingly, Theorem 15.1 does not depend on n, the number of events. It gives the same bound whether there are 100 events each with probability 0.1 or 1000 events each with probability 0.01. In both cases, the expected number of events is 10, and so the probability of no event occurring is at most e^{-10} or about 1 in 22,000. Note that the actual probabilities are somewhat different in these two cases, indicating that the given bound is not always tight.

Theorem 15.1 can be interpreted as a sort of "Murphy's Law": if we expect some things to go wrong, then something probably will. For example, suppose that we are building a microprocessor, and the fabrication process is such that each transistor is faulty mutually independently with a probability of one in a million. This sounds good. However, microprocessors now contain about ten million transistors, so the expected number of faulty transistors is 10 per chip. Since we expect some things to go wrong, something probably will. In fact, Theorem 15.1 implies that the probability of a defect-free a chip is less than 1 in 22,000!

In proving Theorem 15.1, we first note that

$$T = T_1 + T_2 + \cdots + T_n, \tag{34}$$

where T_i is the indicator variable for the event A_i. We also use the fact that

$$1 + x \leq e^x \tag{35}$$

for all x, which follows from the Taylor expansion

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots .$$

Proof.

$$\begin{aligned}
\Pr\{T = 0\} &= \overline{A_1 \cup A_2 \cup \cdots \cup A_n} && \text{(def. of } T\text{)}\\
&= \Pr\left\{\overline{A_1} \cap \overline{A_2} \cap \cdots \cap \overline{A_n}\right\} && \text{(De Morgan's law)}\\
&= \prod_{i=1}^{n} \Pr\left\{\overline{A_i}\right\} && \text{(mutual independence of } A_i\text{'s)}\\
&= \prod_{i=1}^{n} 1 - \Pr\{A_i\} && \text{(complement rule)}\\
&\leq \prod_{i=1}^{n} e^{-\Pr\{A_i\}} && \text{(by (35))}\\
&= e^{-\sum_{i=1}^{n} \Pr\{A_i\}} && \text{(exponent algebra)}\\
&= e^{-\sum_{i=1}^{n} \mathrm{E}[T_i]} && \text{(expectation of indicator variable)}\\
&= e^{-\mathrm{E}[T]}. && \text{((34) \& linearity of expectation)}
\end{aligned}$$

□

Two special cases of Theorem 15.1 are worth singling out because they come up all the time.

Corollary 15.2. *Suppose an event has probability $1/m$. Then the probability that the event will occur at least once in m independent trials is at least approximately $1 - 1/e \approx 63\%$. There is at least 50% chance the event will occur in $n = m \log 2 \approx 0.69m$ trials.*

16 Chernoff Bounds

16.1 Probability of at least *k* events

Now we consider the more general question than the probability that one event occurs, namely, the probability that k events occur, still assuming mutual independence of the events A_i. In other words, what is $\Pr\{T \geq k\}$, given that the events A_i are mutually independent?

For example, suppose we want to know the probability that at least k heads come up in N tosses of a coin. Here A_i is the event that the coin is heads on the ith toss, T is the total number of heads, and $\Pr\{T \geq k\}$ is the probability that at least k heads come up.

As a second example, suppose that we want the probability of a student answering at least k questions correctly on an exam with N questions. In this case, A_i is the event that the student answers the ith question correctly, T is the total number of questions answered correctly, and $\Pr\{T \geq k\}$ is the probability that the student answers at least k questions correctly.

There is an important difference between these two examples. In the first example, all events A_i have equal probability, *i.e.*, the coin is as likely to come up heads on one flip as on another. So T has a binomial distribution whose tail bounds we have already characterized in Notes 11-12.

In the second example, however, some exam questions might be more difficult than others. If Question 1 is easier than Question 2, then the probability of event A_1 is greater than the probability of event A_2. In this section we develop a method to handle this more general situation in which the events A_i may have different probabilities.

We will prove that the number of events that occur is almost never much greater than the expectation. This result is called the Chernoff Bound. For example, if we toss N coins, the expected number of heads is $N/2$ heads. The Chernoff Bound implies that for sufficiently large N, the number of heads is almost always not much greater than $N/2$.

A nice feature of the Chernoff Bound is that we do not even need to know the probability of each event A_i or even the number of events N; rather, we need only the expected number of events that occur and the fact that the events are mutually independent.

16.2 Statement of the Bound

We state Chernoff's Theorem in terms of Bernoulli variables instead of events. However, we can regard T_i as an indicator for the event A_i.

Theorem 16.1 (Chernoff Bound). *Let $T_1, T_2, \ldots, T_n$ be mutually independent Bernoulli variables, and let $T = T_1 + T_2 + \cdots + T_n$. Then for all $c \geq 1$, we have*

$$\Pr\{T \geq c\,\mathrm{E}[T]\} \leq e^{-(c \ln c - c + 1)\,\mathrm{E}[T]}. \tag{36}$$

The formula for the exponent in the bound is a little awkward. The situation is simpler when $c = e = 2.718\ldots$. In this case, $c \ln c - c + 1 = e \ln e - e + 1 = e \cdot 1 - e + 1 = 1$, so we have as an immediate corollary of Theorem 16.1:

Corollary 16.2. *Let $T_1, T_2, \ldots, T_n$ be mutually independent Bernoulli variables, and let $T = T_1 + T_2 + \cdots + T_n$. Then*

$$\Pr\{T \geq e\,\mathrm{E}[T]\} \leq e^{-\mathrm{E}[T]}.$$

We will prove the Chernoff Bound shortly. First, let's see an example of how it is used.

16.3 Example: Pick 4

There is a lottery game called Pick 4. In this game, each player picks 4 digits, defining a number in the range 0 to 9999. A winning number is drawn each week. The players who picked the winning number win some cash. A million people play the lottery, so the expected number of winners each week is

$$\frac{1}{10,000} \cdot 1,000,000 = 100.$$

However, on some lucky day thousands of people might all pick the winning number, costing the lottery operators loads of money. How likely is this?

Assume that all players pick numbers uniformly and mutually independently. Let T_i be an indicator variable for the event that the ith player picks the winning number. Let $T = T_1 + T_2 + \cdots + T_n$. Then T is the total number of players that pick the winning number. As noted above, an average of 100 people win each week, so $\mathrm{E}[T] = 100$. We can use Corollary 16.2 to bound the probability that number of winners is greater than 272 as follows:

$$\Pr\{T \geq 272\} \leq \Pr\{T \geq e\,\mathrm{E}[T]\} \leq e^{-Ex(T)} = e^{-100}.$$

The probability of 272 or more people winning is absurdly small! It appears that the lottery operators should not worry that ten thousand people will pick correctly one day!

But there is a catch. The assumption that people choose Pick 4 numbers uniformly and mutually independently is empirically false; people choose certain "favorite" numbers far more frequently than others.

Chernoff used this fact in devising a scheme to actually make money on the lottery. In this case, a fraction of all money taken in by the lottery was divided up equally among the winners. A bad strategy would be to pick a popular number. Then, even if you pick the winning number, you must share the cash with many other players. A better strategy is to pick a lot of unpopular numbers. You are just as likely to win with an unpopular number, but will not have to share with anyone. Chernoff found that peoples' picks were so highly correlated that he could actually turn a 7% profit by picking unpopular numbers!

16.4 The constant in the exponent

For general c, what can we say about the factor $c \ln c - c + 1$ in the exponent? First, note that when $c = 1$, the exponent factor equals $1 \cdot 0 - 1 + 1 = 0$. This means that the Chernoff bound cannot say anything useful about the probability simply of exceeding the mean. However, the exponent factor increases with c for $c > 1$. This follows because its derivative respect to c is positive:

$$\frac{d(c \ln c - c + 1)}{dc} = (\frac{c}{c} + \ln c) - 1 = \ln c > 0$$

when $c > 1$. In particular, for any $c > 1$, the factor $(c \ln c - c + 1)$ in the exponent is positive.

Let's consider the case of c close to 1, say $c = 1 + \epsilon$. Then a Taylor expansion gives:

$$\begin{aligned}
c \ln c - c + 1 &= (1+\epsilon)\ln(1+\epsilon) - (1+\epsilon) + 1 \\
&= (1+\epsilon)(\epsilon - \frac{\epsilon^2}{2} + \frac{\epsilon^3}{3} - \frac{\epsilon^4}{4} + \cdots) - \epsilon \\
&= (\epsilon - \frac{\epsilon^2}{2} + \frac{\epsilon^3}{3} - \frac{\epsilon^4}{4} + \cdots) + (\epsilon^2 - \frac{\epsilon^3}{2} + \frac{\epsilon^4}{3} - \frac{\epsilon^5}{4} + \cdots) - \epsilon \\
&= \frac{\epsilon^2}{2} - \frac{\epsilon^3}{2 \cdot 3} + \frac{\epsilon^4}{3 \cdot 4} - \frac{\epsilon^5}{4 \cdot 5} + \cdots .
\end{aligned}$$

In particular, for very small ϵ, we have $c \ln c - c + 1 \approx \epsilon^2/2$. In fact one can prove the following:

Lemma 16.3. *For any* $0 < \epsilon < 1$, $\Pr\{T \geq (1+\epsilon)\,\mathrm{E}[T]\} < e^{-\epsilon^2 \mathrm{E}[T]/3}$.

In other words, the probability of deviation above the mean by a fraction ϵ decays *exponentially* in the expected value, for any $\epsilon > 0$.

Another useful observation is that the Chernoff bound starts to "kick in" when $\epsilon \approx 1/\sqrt{\mathrm{E}[T]}$. In other words, the typical deviation of our random variable is going to be about the square root of its expectation. This is in line with our analysis of binomial random variables: the number of heads in n unbiased coin flips has expectation $n/2$, but has standard deviation $\sqrt{n}/2$, or roughly the square root of the expectation.

16.5 Proof of the Bound

The Chernoff Bound uses an ingenious trick along the lines of the way we derived Chebyshev's Bounds:

$$\Pr\{T \geq c\,\mathrm{E}[T]\} = \Pr\left\{c^T \geq c^{c\,\mathrm{E}[T]}\right\} \leq \frac{\mathrm{E}\left[c^T\right]}{c^{c\,\mathrm{E}[T]}} \tag{37}$$

The first step may be a shocker; we exponentiate both sides of the inequality in the probability by c. Since the new inequality describes the same event as the old, the probabilities are the same. The second step uses Markov's Theorem.

Recall that Markov's Theorem sometimes gives weak bounds and sometimes gives tight bounds. The motivation for the first step is to alter the distribution of the random variable T to hit the "sweet spot" of Markov's Theorem. That is, Markov's Theorem gives a tighter bound on the random variable c^T than on the original variable T. We used the same trick in Chebyshev's theorem: we looked at the expectation of T^2 instead of that of T, because that gave us more powerful results.

All that remains is to evaluate $\mathrm{E}\left[c^T\right]$. To do this we need a Lemma:

Lemma 16.4. *If R and S are independent random variables, and f and g are* any *real-valued functions on the reals, then $f(R)$ and $g(S)$ are independent random variables.*

We leave the proof of Lemma 16.4 as a routine exercise.

We begin by calculating $\mathrm{E}\left[c^{T_i}\right]$:

$$\begin{aligned}
\mathrm{E}\left[c^{T_i}\right] &::= c^1 \Pr\{T_i = 1\} + c^0 \Pr\{T_i = 0\} \\
&= c\Pr\{T_i = 1\} + (1 - \Pr\{T_i = 1\}) && \text{(complement rule, since } T_i = 0 \text{ iff } T_i \neq 1) \\
&= 1 + (c-1)\Pr\{T_i = 1\} \\
&\leq e^{(c-1)\Pr\{T_i=1\}} && (1 + x \leq e^x) \\
&= e^{(c-1)\,\mathrm{E}[T_i]} && \text{(expectation of indicator variable).}
\end{aligned} \tag{38}$$

So now we have

$$\begin{aligned}
\mathrm{E}\left[c^T\right] &= \mathrm{E}\left[c^{T_1+T_2+\cdots+T_n}\right] && \text{(def of } T) \\
&= \mathrm{E}\left[c^{T_1} \cdot c^{T_2} \cdots c^{T_n}\right] \\
&= \mathrm{E}\left[c^{T_1}\right] \cdot \mathrm{E}\left[c^{T_2}\right] \cdots \mathrm{E}\left[c^{T_n}\right] && \text{(independence of } T_i\text{'s and Lemma 16.4)} \\
&\leq e^{(c-1)\,\mathrm{E}[T_1]} \cdot e^{(c-1)\,\mathrm{E}[T_2]} \cdots e^{(c-1)\,\mathrm{E}[T_n]} && \text{(by (38))} \\
&= e^{(c-1)\,\mathrm{E}[T_1]+(c-1)\,\mathrm{E}[T_2]+\cdots+(c-1)\,\mathrm{E}[T_n]} \\
&= e^{(c-1)\,\mathrm{E}[T_1+\cdots+T_n]} && \text{(linearity of expectation)} \\
&= e^{(c-1)\,\mathrm{E}[T]} && \text{(def of } T).
\end{aligned} \tag{39}$$

Now we can substitute into the Markov inequality we started with to complete the proof of the

Chernoff bound (36):

$$\begin{aligned}
\Pr\{T \geq c\,\mathrm{E}[T]\} &\leq \frac{\mathrm{E}\left[c^T\right]}{c^{c\,\mathrm{E}[T]}} && \text{(by (37))} \\
&\leq \frac{e^{(c-1)\,\mathrm{E}[T]}}{c^{c\,\mathrm{E}[T]}} && \text{(by (39))} \\
&= \frac{e^{(c-1)\,\mathrm{E}[T]}}{(e^{\ln c})^{c\,\mathrm{E}[T]}} \\
&= e^{(c-1)\,\mathrm{E}[T]-c\ln c\,\mathrm{E}[T]} \\
&= e^{-(c\ln c-c+1)\,\mathrm{E}[T]}.
\end{aligned}$$

16.6 Example: A Phone Network Problem

Suppose that there is a phone network that handles a billion calls a day. Some of these call are routed through a certain switch. The exact number of calls passing through this switch is somewhat random and fluctuates over time, but on average the switch handles a million calls a day.

Our problem is to set the capacity of the switch; that is, we must determine the number of calls that a switch is able to handle in a day. If we make the capacity too small, then some phone calls will not go through. On the other hand, if we make the capacity too large, then we are wasting money. Of course, we cannot rule out a freak situation in which a huge number of calls are all coincidentally routed through the same switch, thus overloading it. However, we would like to guarantee that a switch is rarely overloaded.

Assume that each call has some probability of passing through a particular switch. In particular, let T_i be an indicator variable for the event that the ith call passes through the switch. That is, $T_i = 1$ if the call is routed through the switch, and $T_i = 0$ if the call does not pass through the switch. Then the total call load on the switch is $T = T_1 + T_2 + \cdots + T_n$. We do not know the exact probability that the switch handles the ith call, but we are given that the switch handles an average of a million calls a day; that is, $\mathrm{E}[T] = 1,000,000$.

We will make the crucial assumption that the random variables T_i are mutually independent; that is, calls do or do not pass through the switch mutually independently.

16.6.1 How to Build One Switch

We can now compute the probability that the load on a switch fluctuates upwards by 1% due to the randomness of calling patterns. Substituting $c = 1.01$ and $\mathrm{E}[T] = 1,000,000$ into the Chernoff Bound gives:

$$\begin{aligned}
\Pr\{\text{particular switch overloaded}\} &= \Pr\{T \geq 1.01 \cdot 1,000,000\} \\
&\leq e^{-(1.01\ln 1.01-1.01+1)\cdot 1,000,000} \\
&< e^{-1.01(0.00004934)\cdot 1,000,000} \\
&< 2.3 \cdot 10^{-22}.
\end{aligned}$$

The probability that the load on the switch ever rises by even 1% is unbelievably small! (A June blizzard during an earthquake in Cambridge is far more likely.) If we build the switch with capacity only 1% above the average load, then the switch will (almost) never be overloaded.

The strength of this result relies on the huge expected number of calls. For example, suppose that the average number of calls through the switch were 100 per day instead of 1,000,000. Then every million in the above calculation would be replaced by a hundred; no other numbers change. The final probability of overloading the switch would then be bounded above not by $2.3 \cdot 10^{-22}$, but by 0.995! If the switch handles only 100 calls on an average day, then the call load can very often fluctuate upward by 1% to 101 or more.

16.6.2 How to Build the Network

We now know that building 1% excess capacity into a switch ensures that it is effectively never overloaded. The next problem is to guarantee that no switch in the entire network is overloaded. Suppose that are 1000 switches, and every switch handles an average of a million calls a day.

Previously, we saw that the probability that some event occurs is at most the sum of the event probabilities. In particular, the probability that some switch is overloaded is at most the sum of the probabilities that each of the 1000 switches is overloaded. Therefore, we have:

$$\Pr\{\text{some switch overloaded}\} \leq 1000 \cdot \Pr\{\text{particular switch overloaded}\} < 2.3 \cdot 10^{-19}.$$

This means that building 1% excess capacity into every switch is sufficient to ensure that no switch is ever overloaded.

The above results are of limited practical value, because calls typically do not pass through a switch mutually independently. For example, after an earthquake on Albany Street, everyone would call through the eastern Cambridge switchboard to check on friends and family. Furthermore, there are many more phone calls on certain dates like Mother's Day. On such occasions, 1% excess capacity is insufficient.

17 A Generalized Chernoff Bound

Chernoff's Theorem applies only to sums of Bernoulli (0-1-valued) random variables. It can, however, be extended to apply to sums of random variables with considerably more arbitrary distributions. We state one such generalization in Theorem 40 below. We omit its proof, noting only that the proof is similar to that of the Chernoff bound of Theorem 16.1. The bound of Theorem 40 is not quite as good as the Chernoff bound, but it is more general in what it covers.

Theorem 17.1. *Let $R_1, \dots, R_n$ be independent random variables with $0 \leq R_i \leq 1$. Let $R = \sum_{i=1}^{n} R_i$. Then*

$$\Pr\left\{R - \mathrm{E}[R] \geq c\sqrt{n}\right\} \leq e^{-c^2/2}. \tag{40}$$

Example 17.2. Load balancing.

A set of n jobs have to be scheduled on a set of m equally fast processing machines. The length (processing time) of the ith job is some number L_i in the range $[0, 1]$. We would like to schedule the jobs on the machines so that the loadtime on the machines is reasonably balanced. This means we would like the loadtime on every machine to be not much more than the average loadtime

$L ::= \sum_{i=1}^{n} L_i/m$ per machine. Finding an optimally balanced assignment is a notoriously time-consuming task even when we know all the processing times. But commonly, successive jobs have to be assigned to machines without knowing how long the later jobs will take, and in that case it is impossible to guarantee a balanced load.

We will approach this problem of load balancing using the simplest random strategy: we independently assign each job to a randomly selected machine, with each machine equally likely to be selected. It turns out that for many job scheduling problems, this strategy is almost certain to do very well, even though it does not even take into account the number of jobs nor their processing times.

To see why the simple random strategy does well, notice that, since each job has probability $1/m$ of being assigned to any given machine, the expected loadtime on a machine is precisely $1/m$th of the total time required by the jobs. That is, the *expected* loadtime per machine is precisely the *average* loadtime, L. Now consider any machine, M, and define R_i to be the time M spends processing the ith job. That is, $R_i = L_i$ if the ith job gets assigned to machine M, otherwise $R_i = 0$. So the total loadtime on machine M is $\sum_{i=1}^{n} R_i$. From the generalized Chernoff bound (40), we conclude

$$\Pr\left\{(\text{loadtime on machine } M) - L \geq c\sqrt{n}\right\} \leq e^{-c^2/2}.$$

Now by Boole's inequality,

$$\begin{aligned}
&\Pr\left\{\text{the loadtime on some machine is } \geq L + c\sqrt{n}\right\} \\
&\leq \sum_{k=1}^{m} \Pr\left\{(\text{loadtime on machine } k) \geq L + c\sqrt{n}\right\} \\
&\leq me^{-c^2/2}.
\end{aligned}$$

If we choose $c = \sqrt{2\ln m} + 6$, say, so that $c^2/2 \geq \ln m + 18$, then

$$\begin{aligned}
&\Pr\left\{(\text{loadtime on each machine}) \leq L + c\sqrt{n}\right\} \\
&\geq 1 - me^{-\ln m - 18} \\
&= 1 - e^{\ln m}e^{-\ln m - 18} \\
&= 1 - e^{-18} = 0.99999998\ldots.
\end{aligned}$$

Hence, we can be 99.999998% sure that every machine will have load at most $L + (\sqrt{2\ln m} + 6)\sqrt{n}$. For many values of n and m this is comes very close to balancing the loads on all the machines. For example, if $m = 10$ machines, $n = 5000$ jobs, and average load length $L = 300$ per machine, the maximum load on any machine will almost surely not exceed 332. (In fact it is likely to be even less—we have been fairly crude in our bounds.)

18 Review of Markov, Chebyshev, Chernoff and Binomial bounds

Let us review the methods we have for bounding deviation from the mean via the following example. Assume that I.Q. is made up of thinkatons; each thinkaton fires independently with a 10% chance. We have 1000 thinkatons in all, and I.Q. is the number of thinkatons that fire. What is the probability of having Marilyn's IQ of 228?

So the I.Q. is a Binomial distribution with $n = 1000, p = 0.1$. Hence, $\mathrm{E}\left[\text{I.Q.}\right] = 100, \sigma_{\text{I.Q.}} = \sqrt{0.09 \times 1000} = 9.48$.

An I.Q. of 228 is $128/9.48 > 13.5$ standard deviations away.

Let us compare the methods we have for bounding the probability of this I.Q..

1. Markov:

$$\Pr\left\{\text{I.Q.} \geq 228\right\} \leq \frac{100}{228} < 0.44$$

2. Chebyshev:

$$\Pr\left\{\text{I.Q.} - 100 \geq 128\right\} \leq \frac{1}{13.5^2 + 1} < \frac{1}{183}$$

3. Chernoff:

$$\Pr\left\{\text{I.Q.} \geq 2.28 \times 100\right\} \leq e^{-(2.28 \ln 2.28 - 2.28 + 1)100} \leq e^{-59.9}$$

4. Binomial tails:

$$\Pr\left\{\text{I.Q.} \geq 228\right\} = \Pr\left\{1000 - \text{I.Q.} \leq 772\right\} = F_{0.9,1000}(772) \leq e^{-72.5}$$

   Here we used the formula for the binomial tail from Notes 11-12 with $p = 0.9, n = 1000, \alpha = 0.772$.

Note that the more we know about the distribution the better are the bounds we can obtain on the probability.

Milestones of Probability Theory

1 Introduction

Many probabilistic processes can be understood as limits of processes with binomial densities. That is, their densities can be approximated arbitrarily closely by a binomial density $f_{n,p}$ for suitable choices of n and p. In these Notes, we consider two important results of this kind. First, we consider Poisson processes, which are limits of binomial processes where $n \to \infty$ and $p \to 0$ while np remains constant. Second, by fixing p and letting n approach infinity, we arrive at a profound result of probability theory: the Central Limit Theorem.

We also consider another fundamental result called the *Strong* Law of Large Numbers. The Strong Law provides important information about how the average of independent trials may vary during the course of the trials. The Weak Law we considered in Notes 13-14 is implied by the Strong Law in most circumstances. The Weak Law is also a simple Corollary of the Central Limit Theorem. However, neither the Central Limit Theorem nor the Strong Law imply each other.

2 The Poisson Approximation

We’ve worked with the binomial distribution, which measures the probability of k successful outcomes occur in a sequence of n independent trials. In this section we’ll consider a closely related and widely applicable distribution known as the *Poisson distribution*. The Poisson distribution arises when n is much larger than the expected number of successful outcomes.

2.1 Poisson Random Variables

Let’s consider a particular example. Suppose that we want to model the arrival of packets at an internet router. We know that on average the router handles $\lambda = 10^7$ packets per second. Given this expected value, how can we model the actual distribution of packet arrivals in a second? One possible model is to break up each second into tiny intervals of size $\delta > 0$ seconds, so there are a large number, $n = 1/\delta$, of tiny intervals. Then we declare that in each tiny interval, a packet arrives with probability $\lambda\delta$ (this gives the right expected number of arrivals). Under this model, the number, X, of intervals in which a packet actually arrives has a binomial distribution:

$$\Pr\{X = k\} = \binom{1/\delta}{k} (\lambda\delta)^k (1 - \lambda\delta)^{1/\delta - k}. \tag{1}$$

Note that this is not quite the same as counting the number of arrivals, since more than one packet may arrive in a given interval. But if the interval is tiny, this is so unlikely that we can ignore the possiblity.

Now we let δ become infinitesimally small (while holding k fixed) and make use of three approximations:

$$\begin{aligned}\binom{1/\delta}{k} &\approx \frac{(1/\delta)^k}{k!}\\(1-\lambda\delta)^{1/\delta} &\approx e^{-\lambda}\\1-\delta k &\approx 1.\end{aligned}$$

Plugging these approximations into (1) yields

$$\begin{aligned}\Pr\{X=k\} &= \binom{1/\delta}{k}(\lambda\delta)^k(1-\lambda\delta)^{1/\delta-k}\\&= \binom{1/\delta}{k}(\lambda\delta)^k(1-\lambda\delta)^{(1-\delta k)/\delta}\\&\approx \frac{(1/\delta)^k}{k!}(\lambda\delta)^k(1-\lambda\delta)^{1/\delta}\\&= \frac{\lambda^k}{k!}(1-\lambda\delta)^{1/\delta}\\&\approx \frac{\lambda^k}{k!}e^{-\lambda} \qquad (2)\end{aligned}$$

The probability distribution (2) is known as the *Poisson distribution*. When system events appear according to a Poisson density, the system is called a *Poisson process*.

Another example where a Poisson distribution fits the facts is in observing the number of misprints per page in a book. In a well edited book, there may be an average of one misprint on every three pages. That is, there is an average of $\lambda = 1/3$ misprints per page. An average page has about 40 lines of a dozen words, or about $n = 480$ words. If we suppose that each word has an independent probability of $1/(3 \cdot 480)$ of containing an uncorrected misprint, then the density function of errors per page would be $f_{480,1/1440}$, which will be approximated to three decimal places by the Poisson density with $\lambda = 1/3$.

Further examples of random variables which generally obey a Poisson distribution include:

- the number of decaying particles in a radioactive sample in a given time interval,
- the distribution of the number of failures per day of a system,
- the number of people in a community who are older than 100 years,
- the number of vacancies occurring per year on the Supreme Court,
- the number of wrong telephone numbers dialed in Boston per day.

2.2 Properties of the Poisson Distribution

As a sanity check on our distribution, the probability values (2) had better sum to 1. Using the Taylor expansion for e^{λ}, we can verify that they do:

$$\sum_{k \in \mathbb{N}} \Pr\{X = k\} = \sum e^{-\lambda} \frac{\lambda^k}{k!} = e^{-\lambda} \sum \frac{\lambda^k}{k!} = e^{-\lambda} e^{\lambda} = 1.$$

A further sanity check is that the expected number of arrivals in a second is indeed λ, namely,

$$\mathrm{E}[X] = \lambda. \tag{3}$$

Similarly, the binomial distribution $f_{n,p}$ has variance $np(1-p)$. Since the Poisson distribution is the limit of $f_{1/\delta,\lambda\delta}$ as δ vanishes, it ought to have variance

$$(1/\delta)(\lambda\delta)(1-\lambda\delta) = \lambda(1-\lambda\delta) \approx \lambda.$$

The final approximation holds since $1 - \lambda\delta \approx 1$ for vanishing δ. In other words,

$$\mathrm{Var}[X] = \lambda. \tag{4}$$

Also, suppose we have two independent Poisson processes X_1, X_2 contributing arrival events at the respective rates λ_1, λ_2. Intuitively, this ought to be the same as having a single process producing independent arrivals at the rate $\lambda_1 + \lambda_2$. This explains another useful property of the Poisson distribution:

Lemma 2.1. *If X_1 are X_2 are Poisson processes, then so is $X_1 + X_2$.*

Both equations (3) and (4), and Lemma 2.1, are easy to verify formally from the definition (2) of the Poisson distribution and the Taylor series for e.

[Optional]

Finally, we can develop a Chernoff style bound for the probability that a Poisson process deviates from its mean. As in the proof of the Chernoff bound, we have for *any* random variable, $R \geq 0$, and constants $c, t \geq 0$, that

$$R \geq c \text{ iff } e^{tR} \geq e^{tc}.$$

So by Markov's inequality

$$\Pr\{R \geq c\} \leq \frac{\mathrm{E}\left[e^{tR}\right]}{e^{tc}} \tag{5}$$

For a Poisson process, X, we have

$$\begin{aligned}
\mathrm{E}\left[e^{tX}\right] &= \sum_{k \in \mathbb{N}} \frac{e^{tk} e^{-\lambda} \lambda^k}{k!} \\
&= e^{-\lambda} \sum \frac{(\lambda e^t)^k}{k!} \\
&= e^{-\lambda} e^{\lambda e^t} \\
&= e^{\lambda(e^t - 1)}.
\end{aligned}$$

So if X is a Poisson process,

$$\Pr\{X \geq c\} \leq e^{\lambda(e^t-1)}e^{-tc} = e^{\lambda(e^t-1)-tc}. \tag{6}$$

To minimize the exponent, we choose $t = \ln(c/\lambda)$, which will be positive as long as $c > \lambda$. Substituting this value for t into (6), we conclude

$$\Pr\{X \geq c\} \leq e^{c\ln(\lambda/c)+c-\lambda}. \tag{7}$$

Now letting, $c = c'\lambda$ in (7) and using (3) yields

$$\begin{aligned}\Pr\left\{X \geq c' \operatorname{E}[X]\right\} &\leq e^{c'\lambda\ln(1/c')+c'\lambda-\lambda}\\ &= e^{-c'\lambda\ln c'+c'\lambda-\lambda}\\ &= e^{-(c'\ln c'-c'+1)\lambda}\\ &= e^{-(c'\ln c'-c'+1)\operatorname{E}[X]}.\end{aligned}$$

Notice that this is exactly the same as the Chernoff bound. So we have yet another way in which a Poisson process behaves like a sum of independent indicator variables.

3 The Central Limit Theorem

In the Weak Law of Large Numbers we had $S_n ::= \sum_{i=1}^n G_i$ where $G_1, \ldots, G_i, \ldots$ were mutually independent variables with same mean, μ, and deviation σ. The Weak Law said that the probability that S_n/n was outside an interval of fixed size $\epsilon > 0$ around μ approached 0 as n approached infinity.

The Central Limit Theorem describes not just the limiting behavior of deviation from the mean of S_n/n, but actually describes a limiting shape of the entire distribution for S_n/n. So this theorem substantially refines the Weak Law.

Definition 3.1. For any random variable R with finite mean, μ_R, and deviation, σ_R, let R^* be the random variable

$$R^* ::= \frac{R - \mu_R}{\sigma_R}.$$

R^* is called the "normalized" version of R.

Note that R^* has mean 0 and deviation 1. In other words, R^* is just R shifted and scaled so that its mean is 0 and its deviation and variance are 1.

The Central Limit Theorem says that *regardless* of the underlying distribution of the variables G_i, so long as they are independent, the distribution of S_n^* converges to the same, *normal*, distribution. It is not surprising that this normal distribution—also known as a *Gaussian* distribution—plays a fundamental role in the study of probability and statistics: as long as you are summing enough random variables, you can pretend that the result is Gaussian.

Definition 3.2. The *normal density function* is the function

$$\eta(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2},$$

and the *normal distribution function* is its integral

$$N(y) = \int_{-\infty}^{y} \eta(x)dx = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{y} e^{-x^2/2}dx.$$

The function $\eta(x)$ defines the standard *Bell curve*, centered about the origin with height $1/\sqrt{2\pi}$ and about two-thirds of its area within unit distance of the origin. The normal distribution function $N(y)$ approaches 0 as $y \to -\infty$. As y approaches zero from below, $N(y)$ grows rapidly towards $1/2$. Then as y continues to increase beyond zero, $N(y)$ rapidly approaches 1.

Theorem (Central Limit). *Let $S_n = \sum_{i=1}^{n} G_i$ where $G_1, \ldots, G_i, \ldots$ are mutually independent variables with the same mean, μ, and deviation, σ. Let $\mu_n ::= \mathrm{E}[S_n] = n\mu$, and $\sigma_n ::= \sigma_{S_n} = n\sigma$. Now let $S_n^* ::= (S_n - \mu_n)/\sigma_n$ be the normalized version of S_n. Then*

$$\lim_{n\to\infty} \Pr\{S_n^* \le \beta\} = N(\beta)$$

for any real number β.

To understand the Central Limit Theorem, it helps to see how it implies the Weak Law of Large Numbers.

Note first that $\mu_{S_n} = n\mu$, $\mathrm{Var}[S_n] = n\sigma^2$, and so $\sigma_{S_n} = \sigma\sqrt{n}$. Now,

$$\begin{aligned}\left|\frac{S_n}{n} - \mu\right| > \epsilon \quad &\text{iff} \quad |S_n - n\mu| > n\epsilon \\ &\text{iff} \quad \left|\frac{S_n - n\mu}{\sigma_{S_n}}\right| > \frac{n\epsilon}{\sigma_{S_n}} \\ &\text{iff} \quad |S_n^*| > \frac{\sqrt{n}\epsilon}{\sigma}.\end{aligned}$$

But for any real number $\beta > 0$,

$$\frac{\sqrt{n}\epsilon}{\sigma} > \beta$$

will hold for all large n. Hence, for any $\beta > 0$ and all large n,

$$\Pr\left\{\left|\frac{S_n}{n} - \mu\right| > \epsilon\right\} = \Pr\left\{|S_n^*| > \frac{\sqrt{n}\epsilon}{\sigma}\right\} \le \Pr\{|S_n^*| > \beta\}. \tag{8}$$

So

$$\begin{aligned}\lim_{n\to\infty} \Pr\left\{\left|\frac{S_n}{n} - \mu\right| > \epsilon\right\} &\le \lim_{n\to\infty} \Pr\{|S_n^*| > \beta\} && \text{(by (8))} \\ &= \lim_{n\to\infty} \Pr\{S_n^* > \beta\} + \Pr\{S_n^* < -\beta\} \\ &= 1 - N(\beta) + N(-\beta), && \text{(by the Central Limit Thm (3))}\end{aligned}$$

for all real numbers $\beta > 0$. By choosing β large enough, we can ensure that $N(\beta)$ is arbitrarily close to 1 and $N(-\beta)$ is arbitrarily close to 0, so that final term above is arbitrarily close to 1-1+0 = 0. Hence,

$$\lim_{n\to\infty} \Pr\left\{\left|\frac{S_n}{n} - \mu\right| > \epsilon\right\} = 0,$$

which is the Weak Law of Large Numbers.

We will not prove the Central Limit Theorem, but will only note that a standard proof rests on extending ideas we have already used in deriving Chernoff bounds, in particular properties of $\mathrm{E}\left[e^{tX}\right]$. Regarded as a function of t, $\mathrm{E}\left[e^{tX}\right]$ is called the *moment generating function* of the random variable, X. The Central Limit Theorem can be proved using a more complete development of the properties of moment generating functions, more than we have time for in 6.042.

Like the Weak Law of Large Numbers, the Central Limit Theorem as stated cannot be applied to actual problems because the necessary information about the rate of convergence is missing, that is, we need to know the accuracy with which the limit $N(\beta)$ approximates the probability that $S_n^* < \beta$. For variables $G_1, G_2, \ldots$ whose absolute value is bounded by about 5, or are themselves are normal, a rule of thumb is that (3) holds to one or two decimal places when $|\beta| < 3$ and $n > 30$. But in situations such as those we have seen for designing overload tolerance into systems, and also for ensuring the quality of the solution to an optimization problem by a probabilistic algorithm, we are typically more concerned with events that differ from the mean by many standard deviations. For estimating probabilities at such distribution tails, Chernoff bounds are more accurate than those based on normal distributions. For this reason, Chernoff bounds play a more prominent role in Computer Science than the Central Limit Theorem.

4 Strong Law of Large Numbers [Optional]

[Optional]

We described the *Weak* Law of Large Numbers in previous notes, begging the question of what *strong* law of large numbers we might prove. Roughly speaking, the strong law says that with probability 1, the bound of the weak law will hold for all but a finite number of the S_n *simultaneously*—there will only be finitely many exceptions to it.

Theorem 4.1. *[The Strong Law of Large Numbers]* [1] *Let* $S_n ::= \sum_{i=1}^n X_i$ *where* $X_1, \ldots, X_i, \ldots$ *are mutually independent, identically distributed random variables with finite expectation,* μ*. Then*

$$\Pr\left\{\lim_{n\to\infty} \frac{S_n}{n} = \mu\right\} = 1.$$

Although Theorem 4.1 can be proven without this assumption, we will assume for simplicity that the random variables X_i have a finite fourth moment. That is, we will suppose that

$$\mathrm{E}\left[X_i^4\right] = K < \infty. \tag{9}$$

Proof. To begin, assume that μ, the mean of the X_i, is equal to 0. As usual, let $S_n ::= \sum_{i=1}^n X_i$ and consider

$$\mathrm{E}\left[S_n^4\right] = \mathrm{E}\left[(X_1 + \cdots + X_n) \times (X_1 + \cdots + X_n) \times (X_1 + \cdots + X_n) \times (X_1 + \cdots + X_n)\right]. \tag{10}$$

Expanding the righthand side of (10) results in terms of the forms

$$X_i^4, \qquad X_i^3 X_j, \qquad X_i^2 X_j^2, \qquad X_i^2 X_j X_k, \qquad X_i X_j X_k X_l$$

where i, j, k, l are all different. As all the X_i have mean 0, it follows by independence that

$$\begin{aligned}
\mathrm{E}\left[X_i^3 X_j\right] &= \mathrm{E}\left[X_i^3\right]\mathrm{E}\left[X_j\right] = 0\\
\mathrm{E}\left[X_i^2 X_j X_k\right] &= \mathrm{E}\left[X_i^2\right]\mathrm{E}\left[X_j\right]\mathrm{E}\left[X_k\right] = 0\\
\mathrm{E}\left[X_i X_j X_k X_l\right] &= 0.
\end{aligned}$$

[1] This section taken from Ross, *A First Course in Probability Theory*.

Now, for a given pair i and j there will be $\binom{4}{2} = 6$ terms in the expansion that will equal $X_i^2 X_j^2$. Hence, after expanding the righthand side (10), we have

$$\begin{aligned} \mathrm{E}\left[S_n^4\right] &= n\,\mathrm{E}\left[X_i^4\right] + 6\binom{n}{2}\mathrm{E}\left[X_i^2 X_j^2\right] && \text{(linearity of expectation)} \quad (11) \\ &= nK + 3n(n-1)\,\mathrm{E}\left[X_i^2\right]\mathrm{E}\left[X_j^2\right]. && \text{(by (9) and independence)} \quad (12) \end{aligned}$$

Now, since

$$0 \leq \mathrm{Var}\left[X_i^2\right] = \mathrm{E}\left[X_i^4\right] - \mathrm{E}^2\left[X_i^2\right]$$

we see that

$$\mathrm{E}^2\left[X_i^2\right] \leq \mathrm{E}\left[X_i^4\right] = K.$$

Therefore, from (12) we have that

$$\mathrm{E}\left[S_n^4\right] \leq nK + 3n(n-1)K$$

which implies that

$$\mathrm{E}\left[\frac{S_n^4}{n^4}\right] \leq \frac{K}{n^3} + \frac{3K}{n^2},$$

and so

$$\mathrm{E}\left[\sum_{i=1}^{\infty}\frac{S_n^4}{n^4}\right] = \sum_{i=1}^{\infty}\mathrm{E}\left[\frac{S_n^4}{n^4}\right] \leq K\sum_{i=1}^{\infty}\frac{1}{n^3} + \frac{3}{n^2} < \infty.$$

But since the expected value is finite, the probability that $\sum_{i=1}^{\infty} S_n^4/n^4$ is finite must be one. (If there was a positive probability that the sum is infinite, then its expected value would be infinite.) Now the convergence of a series implies that its nth term goes to 0; so we can conclude that $\lim_{n\to\infty} S_n^4/n^4 = 0$ with probability 1. But if $S_n^4/n^4 = (S_n/n)^4$ goes to 0, then so must S_n/n; so we have completed the proof that with probability 1,

$$\frac{S_n}{n} \to 0 \qquad \text{as } n \to \infty.$$

When μ, the mean of the X_i, is not equal to 0, we can apply the preceding argument to the random variables $X_i - \mu$ to obtain that with probability 1,

$$\lim_{n\to\infty}\sum_{i=1}^{n}\frac{X_i - \mu}{n} = 0$$

or, equivalently,

$$\lim_{n\to\infty}\sum_{i=1}^{n}\frac{X_i}{n} = \mu$$

which proves the result. □

We remark that as in the Weak Law, full mutual independence of $\{X_i\}$ is not necessary. The proof above only requires that $\{X_i\}$ are 4-way independent.

4.1 A Failure of the Strong Law

To clarify the somewhat subtle difference between the Weak and Strong Laws of Large Numbers, we will construct an example of a sequence $X_1, X_2, \ldots$ of mutually independent random variables that satisfies the Weak Law of Large

Numbers, but not the Strong Law. The distribution of X_i will have to depend on i, because otherwise both laws would be satisfied.[2]

In particular, let $X_1, X_2, \ldots$ be the sequence of mutually independent random variables such that $X_1 = 0$, and for each integer $i > 1$,

$$\Pr\{X_i = i\} = \frac{1}{2i\log i}, \quad \Pr\{X_i = -i\} = \frac{1}{2i\log i}, \quad \Pr\{X_i = 0\} = 1 - \frac{1}{i\log i}.$$

Note that $\mu = \mathrm{E}[X_i] = 0$ for all i.

Problem. (a) Show that $\mathrm{Var}[S_n] = \Theta(n^2/\log n)$. *Hint:* $n/\log n > i/\log i$ for $2 \le i \le n$.

(b) Show that the sequence $X_1, X_2, \ldots$ satisfies the Weak Law of Large Numbers, *i.e.*, prove that for any $\epsilon > 0$

$$\lim_{n\to\infty} \Pr\left\{\left|\frac{S_n}{n}\right| \ge \epsilon\right\} = 0.$$

We now show that the sequence $X_1, X_2, \ldots$ does not satisfy the Strong Law of Large Numbers.

(c) (The first Borel-Cantelli lemma.) Let $A_1, A_2, \ldots$ be any infinite sequence of mutually independent events such that

$$\sum_{i=1}^{\infty} \Pr\{A_i\} = \infty. \tag{13}$$

Prove that

$$\Pr\{\text{infinitely many } A_i \text{ occur}\} = 1.$$

Hint: We know that the probability that no A_i with $i \ge r$ occurs is

$$\le e^{-\mathrm{E}[T_r]} \tag{14}$$

where $T_r ::= \sum_{i=r}^{\infty} I_{A_i}$ is the number of events A_i with $i \ge r$ that occur.

(d) Show that $\sum_{i=1}^{\infty} \Pr\{|X_i| \ge i\}$ diverges. *Hint:* $\int dx/(xlogx) = \log\log x$.

(e) Conclude that

$$\Pr\left\{\lim_{n\to\infty} \frac{S_n}{n} = \mu\right\} = 0. \tag{15}$$

and hence that the Strong Law of Large Numbers *completely* fails for the sequence $X_1, X_2, \ldots$.
Hint:

$$\frac{X_n}{n} = \frac{S_n}{n} - \frac{n-1}{n}\frac{S_{n-1}}{n-1},$$

so if $\lim_{n\to\infty} S_n/n = 0$, then also $\lim_{n\to\infty} X_n/n = 0$.

[2] This problem is adapted from Grinstead & Snell, *Intro. to Probability*, Ch.8, exercise 16, pp314–315, where it is credited to David Maslen.