1

Introduction

Purpose of the Course

The purpose of this course is to study mathematically the behaviour of stochastic systems.

Examples

1. A CPU with jobs arriving to it in a random fashion.
2. A communications network multiplexor with “packets” arriving to it randomly.
3. A store with random demands on the inventory.

Often part of the reason for studying the system is to be able to predict how it will behave depending on how we design or control it.

Examples

1. A CPU could process jobs
    - one at a time as they arrive (FIFO).
    - one at a time but according to some predefined priority scheme.
    - all at the "same time" (processor sharing).
2. A network multiplexor could transmit packets from different connections
    - as they arrive (statistical multiplexing).
    - in a round-robin fashion (time-division multiplexing).
3. We could replenish a store's inventory
    - when it drops below a predefined level (dynamic).
    - once a month (static).

   Also, how much should we stock each time we replenish the inventory?

Over the last several decades stochastic process models have become important tools in many disciplines and for many applications (e.g. the spread of cancer cells, growth of epidemics, naturally occuring genetic mutation, social and economic mobility, population growth, fluctuation of financial instruments).

Our goal is to study models of "real" systems that evolve in a random fashion and for which we would like to quantify the likelihood of various outcomes.

2

Stochastic Processes and Some Probability Review

Stochastic Processes

Stochastic means the same thing as *random* and probability models that describe a quantity that evolves randomly in time are called *stochastic processes.*

A stochastic process is a sequence of random variables $\{X_u, u \in I\}$, which we will sometimes denote by $\boldsymbol{X}$ for shorthand (a bold X).

- u is called the *index*, and most commonly (and always in this course) it denotes time. Thus we say 'X_u is the state of the system at time u.'
- I is the *index set*. This is the set of all times we wish to define for the particular process under consideration.

The index set I will be either a discrete or a continuous set. If it is discrete (e.g. $I = \{0, 1, 2, \ldots\}$) then we say that $\boldsymbol{X}$ is a *discrete-time* stochastic process. If it is continuous (e.g. $I = [0, \infty)$) then we say $\boldsymbol{X}$ is a *continuous-time* stochastic process.

Whether the index set I is discrete or continuous is important in determining how we mathematically study the process. Chapter 4 of the text deals exclusively with a class of discrete-time processes (Markov Chains) while chapters 5 and 6 deal with the analogous class of processes in continuous time (Poisson Processes in chapter 5 and Continuous Time Markov Processes in chapter 6).

Notation: We won't always use $\boldsymbol{X}$ or $\{X_u, u \in I\}$ to denote a stochastic process. The default letter to use for a stochastic process will be (a captital) X, but we'll use other letters too (like Y, Z, W, S, N, M and sometimes lower case letters like x, y, w, etc., and sometimes even Greek letters like ξ, α or β), like for example when we want notation for two or more different processes, but we'll try to use the X notation whenever possible. Also, the index won't always be u (actually it will usually be something different like n or t), but we'll (almost) always use a lower-case Roman letter for the index (unless we're referring to a specific time in which case we use the value of the index, like 1, 2, etc.)

By convention we use certain letters for discrete time indexes and other letters for continuous time indexes. For discrete time indexes we'll usually use the letter n for the index, as in X_n, where n usually will represent a nonnegative integer, and we'll also use the letters i, j, k, l, m. For continuous time indexes we'll usually use the letter t, as in X_t, where t usually will represent a nonnegative real number, and we'll also use the letters s, r, u, h, ϵ. The letters h and ϵ by convention will be reserved for small numbers, as in X_{t+h}, where we mean the process at a time point just after time t.

We'll never use the same letter for the process and the index, as in s_s. That's bad and confusing notation, and really an incorrect use of notation.

State Space

The other fundamental component (besides the index set I) in defining the structure of a stochastic process is the *state space*. This is the set of all values that the process can take on, much like the concept of the sample space of a random variable (in fact, it is the sample space of each of the random variables X_u in the sequence making up the process), and we usually denote the state space by S.

Examples

1. If the system under study is a CPU with randomly arriving jobs, we might let $I = [0, \infty)$ and $S = \{0, 1, 2, \ldots\}$, where the state represents the number of jobs at the CPU either waiting for processing or being processed. That is, X_t is the number of jobs at the CPU waiting for processing or being processed at time t.
2. When studying the levels of inventory at a store we might let $I = \{0, 1, 2, \ldots\}$ and $S = \{0, 1, \ldots, B\}$, where the state represents the number of units of the inventory item currently in the inventory, up to a maximum of B units. That is, X_n is the number of units in the inventory at time n.

The units of the time index are completely up to us to specify. So in the inventory example the time index n could mean week n, month n, or just time period n if we want to leave it more unspecified. But we can only choose the unit to represent one thing; the time unit can't represent, for example, both days and weeks.

Like the index set I, the state space S can be either discrete or continuous. However, dealing mathematically with a continuous state space involves some technical details that are beyond the scope of this course, as they say, and are not particularly instructive. Moreover, most real world systems can be adequately described using a discrete state space. In this course we'll always assume the state space is discrete.

By discrete we mean that the state space is either finite or countable. It's pretty obvious what we mean by *finite* (e.g. $S = \{0, 1, 2, \ldots, B\}$). In case you don't know, *countable* means that there is a one-to-one correspondence between the elements of S and the natural numbers $\{1, 2, 3, \ldots\}$. So we could count all the elements (if we had an infinite amount of time) and not miss any elements. Examples are the set of all integers, the set of all multiples of 0.25, or the set of all rational numbers. This is in contrast to *uncountable*, or continuous, sets, like the interval $[0, 1]$. We could never devise a counting scheme to count all the elements in this set even if we had an infinite amount of time.

The index set I and the state space S are enough to define the basic structure of the stochastic process. But we also need to specify how the process evolves randomly in time. We'll come back to this when we start studying chapter 4. Before that, we'll review some probability theory and study the concept of *conditioning* as a useful technique to evaluate complex probabilities and expectations.

Some Probability Review

Until chapter 5 in the text we'll be dealing almost exclusively with discrete probabilities and random variables. You are expected to know about continuous random variables and density functions and using integration to calculate things like probabilities and expectations, but it may help you to organize any time you spend reviewing basic probability concepts to know that we won't be using continuous random variables regularly until chapter 5.

Fundamental concepts in probability are things like *sample spaces*, *events*, *the axioms of probability*, *random variables*, *distribution functions*, and *expectation*.

Another fundamental concept is *conditioning*. We'll devote most of the first two weeks of the course on this valuable idea.

Let's start with a simple example in which we calculate a probability. This example is meant to cover some basic concepts of probability modeling.

Example: In an election candidate A receives n votes while candidate B receives just 1 vote. What is the probability that A was always ahead in the vote count assuming that every ordering in the vote count is equally likely?

Solution: First we define an appropriate sample space. Since vote count orderings are equally likely we can set this up as a counting problem if we make our sample space the set of all possible vote count orderings of $n+1$ votes that have n votes for candidate A and 1 vote for candidate B. If we do this we need to be able to count the total number of such orderings and also the number of such orderings in which A is always ahead in the vote count. We think we can do the counting so we proceed. In fact there are $n+1$ possible vote count orderings (in the ith ordering B received the ith vote). Moreover, A had to receive the first two votes to be always ahead in the vote count. There are $n-1$ orderings in which A received the first two votes. So our desired probability is $\frac{n-1}{n+1}$. □

In this example we set up a sample space and defined an event of interest in terms of outcomes in the sample space. We then computed the probability of the event by counting the number of outcomes in the event. This can be done if the outcomes in the sample space are all equally likely.

Was that simple? Try this one.

Example: A fair coin is tossed repeatedly. Show that a heads is eventually tossed with probability 1.

Solution: The event {heads eventually} is the same as the union of all events of the form {heads flipped for the first time on toss n}, for $n \geq 1$. That is,

$$\{\text{Heads eventually}\} = \{H\} \bigcup \{TH\} \bigcup \{TTH\} \bigcup \{TTTH\} \bigcup \ldots$$

The events in the union above are all *mutually exclusive* and, because we are implicitly assuming that the outcomes of different tosses of the coin are independent, the probability that a heads is flipped for the first time on the nth toss is $1/2^n$, so

$$\begin{aligned} P(\text{Heads eventually}) &= P(H) + P(TH) + P(TTH) + \ldots \\ &= \sum_{n=1}^{\infty} \frac{1}{2^n} \\ &= \frac{1}{2} \sum_{n=0}^{\infty} \frac{1}{2^n} \\ &= \frac{1}{2} \times 2 = 1, \end{aligned}$$

as desired. □

Concepts to review here are mutually exclusive events, independent events, and the geometric series.

Another solution to the previous problem, one which utilizes the very useful basic probability rule that $P(A^c) = 1 - P(A)$ for any event A, is to determine the probability of the complement of the event. The logical opposite of the event that a heads eventually occurs is the event that it never occurs. As is sometimes the case, the complement event is easier to work with. Here the event that a heads never occurs is just the event that a tails is flipped forever:

$$\{\text{head eventually}\}^c = \{TTTTTT\ldots\}.$$

How do we show this event has probability 0?
If you were to write the following

$$P(TTTTTT\ldots) = \left(\frac{1}{2}\right)^{\infty} = 0$$

I wouldn't mark it wrong, but one must always be careful when working with ∞. This is because ∞ is technically not a number. The symbol ∞ is really the mathematician's shorthand way of saying "the limit as n goes to ∞". That is

$$\left(\frac{1}{2}\right)^{\infty} \quad \text{really means} \quad \lim_{n\to\infty}\left(\frac{1}{2}\right)^{n}$$

and also the validity of the statement that $P(TTTTT\ldots) = \left(\frac{1}{2}\right)^{\infty}$ relies on the fact that

$$\lim_{n\to\infty} P(TTT\ldots T) = P(\lim_{n\to\infty} TTT\ldots T),$$

where the number of T's in each of the above events is n.

Another important point to note is that the event $\{TTTT\ldots\}$ of flipping tails forever is not a logically impossible event (in terms of sets this means it's not the empty set). However, it has probability 0. There's a difference between *impossible* events and events *of probability 0*.

Here's a more extreme example.

Example: *Monkey Typing Shakespeare.* A monkey hits keys on a typewriter randomly and forever. Show that he eventually types the complete works of Shakespeare with probability 1.

Solution: Let N be the number of characters in the complete works of Shakespeare and let T be the number of different keys on the keypad of the typewriter. Let A be the event that the monkey never types the complete works of Shakespeare, and we'll show that $P(A) = 0$. To do this we'll use an important technique in mathematics, that of *bounding*. Specifically, divide up the sequence of typed characters into blocks of N characters, starting with the first typed character. Let B be the event that the monkey never types the complete works of Shakespeare in one of the blocks. We'll show that B has probability 0 and that A is contained in B. This will show that A also has probability 0. We work with the event B because it's actually rather trivial to show that it has probability 0. Let B_n be the event that the monkey doesn't type the complete works of Shakespeare in the the nth block. Because the blocks are disjoint the outcomes in different blocks are independent, and also

$$B = B_1 \bigcap B_2 \bigcap B_3 \bigcap \ldots$$

so that

$$P(B) = P(B_1)P(B_2)P(B_3)\ldots$$

But $P(B_n)$ is in fact is the same for all n because all blocks are identically distributed, so that

$$P(B) = P(B_1)^\infty.$$

So to show $P(B) = 0$ all we need is that $P(B_1) < 1$, but this is clearly so since, even though it's small, the probability that the monkey does type the complete works of Shakespeare in the first N keystrokes is nonetheless positive (as an exercise calculate exactly $P(B_1)$). Therefore, $P(B) = 0$. Finally, it can be seen that event A logically implies event B but event B does not imply event A, because B could occur even though our monkey did type the complete works of Shakespeare (just not in one of the blocks). Therefore, $A \subset B$, which implies $P(A) \leq P(B)$, or $P(A) = 0$. □

Note that both events A and B have infinitely many outcomes, for there are infinitely many infinitely long sequences of characters that do not contain any subsequence of length N that types out the complete works of Shakespeare, so that both A and B are clearly not logically impossible events. Yet both are events of zero probability.

One final point I would like to make in this example is again concerning the notion of infinity. Our monkey may indeed eventually write the complete works of Shakespeare given enough time, but probably not before our galaxy has been sucked into a black hole. So the knowledge that it will eventually happen has no practical use here because it would take too long for it to be of any use.
That's not the point I'm trying to make, though. The point is that statisticians regularly let "things go to infinity" because it often is the case that "infinity" happens really fast, at least in a practical sense. The best example is perhaps the Central Limit Theorem, which says that

$$\frac{\sum_{i=1}^{n}(X_i - \mu)}{\sigma\sqrt{n}} \Rightarrow N(0,1),$$

where the X_i are independent random variables each with mean μ and variance σ^2, $N(0,1)$ denotes the standard normal distribution, and $\Rightarrow$ denotes convergence in distribution. You may have learned a rough rule of thumb that if $n \geq 30$ then the limiting $N(0,1)$ distribution provides a good approximation (i.e. 30 is effectively equal to ∞). This is what makes the Central Limit Theorem so useful.

Similarly, when we study a stochastic process in discrete time, say $\{X_n, n = 0, 1, 2 \ldots\}$, one of the important things that we'll be trying to do is get the limiting distribution of X_n as n goes to infinity. We do this because this limiting distribution is often a good approximation to the distribution of X_n for small n. What this means is that if we let a system operate "for a little while", then we expect that the probability of it being in a given state should follow the probability of that state given by the limiting distribution.

3

Some Expectation Examples

Expectation

Let X be a discrete random variable defined on a sample space S with probability mass function $f_X(\cdot)$. The *expected value* of X, also called the *mean* of X and denoted $E[X]$, is

$$E[X] := \sum_{x \in S} x f_X(x) = \sum_{x \in S} x P(X = x),$$

if the sum is absolutely convergent. Note that the sample space of a random variable is always a subset of $\mathcal{R}$, the real line.

Law of the Unconscious Statistician

Let $g(x)$ be an arbitrary function from S to $\mathcal{R}$. Then the expected value of the random variable $g(X)$ is

$$E[g(X)] = \sum_{x \in S} g(x) f_X(x) = \sum_{x \in S} g(x) P(X = x)$$

If $g(X) = X^2$ then its mean is called the *second moment* of X. In general, $E[X^k]$ is called the kth moment of X. The first moment is the same as the mean of X. The first and second moments are the most important moments. If $g(X) = (X - E[X])^2$ then its mean is called the *second central moment*. $E[(X - E[X])^2]$ is also commonly called the *variance* of X.

Example: Find the mean of the Geometric(p) distribution.

Solution: The Geometric(p) distribution has probability mass function

$$f(k) = p(1-p)^{k-1} \quad \text{for } k = 1, 2, 3, \ldots$$

so if X is a random variable with the Geometric(p) distribution,

$$\begin{aligned} E[X] &= \sum_{k=1}^{\infty} kp(1-p)^{k-1} \\ &= p \sum_{k=1}^{\infty} k(1-p)^{k-1}. \end{aligned}$$

There is a standard way to evaluate this infinite sum. Let

$$g(p) = \sum_{k=0}^{\infty} (1-p)^k.$$

This is just a Geometric series so we know that

$$g(p) = \frac{1}{1-(1-p)} = \frac{1}{p}.$$

The derivative is $g'(p) = -1/p^2$, which has the following form based on its infinite sum representation:

$$g'(p) = -\sum_{k=1}^{\infty} k(1-p)^{k-1}.$$

In fact we've evaluated the negative of the infinite sum in $E[X]$:

$$E[X] = p\frac{1}{p^2} = \frac{1}{p}.$$

Next week we'll see how we can evaluate $E[X]$ much more simply and naturally by using a conditioning argument. □

We can also find the second moment, $E[X^2]$, of the Geometric(p) distribution in a similar fashion by using the Law of the Unconscious Statistician, which allows us to write

$$E[X^2] = p\sum_{k=1}^{\infty} k^2(1-p)^{k-1}.$$

One might consider trying to take the second derivative of $g(p)$. When this is done, one gets

$$g''(p) = \sum_{k=2}^{\infty} k(k-1)(1-p)^{k-2}.$$

This is not quite what we want, but it is close. Actually,

$$\begin{aligned} p(1-p)g''(p) &= p\sum_{k=2}^{\infty} k(k-1)(1-p)^{k-1} \\ &= p\sum_{k=1}^{\infty} k(k-1)(1-p)^{k-1} \\ &= E[X(X-1)]. \end{aligned}$$

Since we know $g'(p) = -1/p^2$ we have that $g''(p) = 2/p^3$ and so

$$E[X(X-1)] = \frac{2p(1-p)}{p^3} = \frac{2(1-p)}{p^2}.$$

To finish it off we can write $E[X(X-1)] = E[X^2 - X] = E[X^2] - E[X]$ so that

$$\begin{aligned} E[X^2] &= E[X(X-1)] + E[X] \\ &= \frac{2(1-p)}{p^2} + \frac{1}{p} \\ &= \frac{2}{p^2} - \frac{1}{p}. \end{aligned}$$

Expectation is a very important quantity when evaluating a stochastic system.

- In financial markets, expected return is often used to determine a "fair price" for financial derivatives and other equities (based on the notion of a fair game, for which a fair price to enter the game is the expected return from the game, so that your expected net return is zero).

- When designing or controlling a system which provides a service (such as a CPU or a communications multiplexor) which experiences random demands on the resources, it is often the average system behaviour that one is trying to optimize (such as expected delay, average rate of denial of service, etc.)

- When devising strategies for investment, expected profit is often used as a guide for developing optimal strategies (e.g. a financial portfolio).

- In the inventory example, we might use the expected number of unfilled orders to determine how best to schedule the replenishment of the inventory. Of course, you can quickly see that this doesn't work because according to this criterion we should just stock the inventory with an infinite number of units. We're ignoring a crucial element: cost. This would lead us to develop a *cost function*, which would reflect components such as lost orders, cost of storage, cost of the stock, and possibly other factors, and then try to develop a schedule that minimizes the expected "cost".

Example: Suppose you enter into a game in which you roll a die repeatedly and when you stop you receive k dollars if your last roll showed a k, except that you must stop if you roll a 1. A reasonable type of strategy would be to stop as soon as the die shows m or greater. What's the best m?

Solution: We will use as our criterion for deciding what m is best the expected prize. Firstly, if $m = 1$ or $m = 2$ then this corresponds to the strategy in which you stop after the first roll of the dice. The expected prize for this strategy is just the expected value of one roll of the dice. Assuming the dice is fair (each outcome is equally likely), we have
For $m = 1$ or $m = 2$:

$$\text{Expected prize} = \frac{1}{6}(1+2+3+4+5+6) = \frac{21}{6} = \frac{7}{2}.$$

Let's try $m = 6$. In this strategy we stop as soon as we roll a 1 or a 6. Let X denote the number on the dice when we stop. Then X is a random variable that takes on the values 1 or 6 and

$$\text{Expected prize} = (1)P(X = 1) + (6)P(X = 6).$$

Now we need to determine $P(X = 1)$ and $P(X = 6)$. Note that we are writing the expectation above in terms of the distribution of X. We could have invoked the Law of the Unconscious Statistician and written the expectation as a sum over all possible outcomes in the underlying experiment of the value of X corresponding to that outcome times the probability of that outcome (outcomes in the underlying experiment are sequences of dice rolls of finite length that end in a 1 or a 6). However, there are an infinite number of possible outcomes in the underlying experiment and trying to evaluate the sum over all of the outcomes would be more complex than necessary (it actually wouldn't be that hard in this case and you might try it as an exercise). It's unnecessary in this case because we will be able to determine the distribution of X without too much trouble, and in this case it's better to compute our desired expectation directly from the distribution of X. So what is $P(X = 6)$? An intuitive argument would be as follows. When we stop we roll either a 1 or a 6 but on that last roll we are equally likely to roll either a 1 or a 6, so $P(X = 6) = 1/2$ which also gives $P(X = 1) = 1/2$. This informal argument turns out to be correct, but one should usually be careful when using intuitive arguments and try to check the correctness of the answer more rigorously. We'll do a more rigorous argument here. Let T denote the roll number of the last roll. Then T is a random variable that takes on the values $1, 2, 3, \ldots$. Let A_n be the event $\{T = n\}$, for $n \geq 1$, and let $A = A_1 \bigcup A_2 \bigcup A_3 \bigcup \ldots$. Then the set of events $\{A_1, A_2, A_3, \ldots\}$ is what we call a *partition* of the sample space because it is a collection of mutually exclusive events and their union is the whole sample space (every outcome in our underlying experiment corresponds to exactly one of the A_i). In particular, the event A is the whole sample space, so that $\{X = 6\} \bigcap A = \{X = 6\}$, and

$$\begin{aligned}
P(\{X = 6\}) &= P(\{X = 6\} \bigcap A) \\
&= P\left(\{X = 6\} \bigcap \left(A_1 \bigcup A_2 \bigcup A_3 \bigcup \ldots\right)\right) \\
&= P\left((\{X = 6\} \bigcap A_1) \bigcup (\{X = 6\} \bigcap A_2) \bigcup \ldots\right) \\
&= P\left(\{X = 6\} \bigcap A_1\right) + P\left(\{X = 6\} \bigcap A_2\right) + \ldots
\end{aligned}$$

because the events $\{X = 6\} \bigcap A_n$ in the union in the third equality above are mutually disjoint (because the A_n are mutually disjoint). We've gone to some pain to go through all the formal steps to show that

$$P(X = 6) = P(X = 6, T = 1) + P(X = 6, T = 2) + \dots$$

partly because intersecting an event with the union of the events of a partition is a fairly important and useful technique for computing the probability of the event, as long as we choose a useful partition, where a partition is "useful" if it provides "information" such that calculating the probability of the intersection of the event and any member of the partition is easier to do than calculating the probability of the event itself. Here the event $\{X = 6\} \bigcap \{T = k\}$ can only happen if the first $k - 1$ rolls of the dice were not a 1 or a 6 and the kth roll was a 6. Since rolls of the dice are independent, it's easy to see that the probability of this is

$$P(X = 6, T = k) = \left(\frac{4}{6}\right)^{k-1} \frac{1}{6}.$$

Thus,

$$\begin{aligned} P(X = 6) &= \sum_{k=1}^{\infty} \left(\frac{4}{6}\right)^{k-1} \frac{1}{6} \\ &= \frac{1}{6} \times \frac{1}{1 - 4/6} = \frac{1}{6} \times \frac{6}{2} = \frac{1}{2}, \end{aligned}$$

confirming our earlier intuitive argument. Next week we'll look at how we would calculate $P(X = 6)$ using a conditioning argument.

Going back to our original calculation, we have

For $m = 6$:

$$\text{Expected prize} = \frac{1}{2}(1 + 6) = \frac{7}{2},$$

which is the same as the expected prize for the $m = 1$ and $m = 2$ strategies. Moving on (a little more quickly this time), let's calculate

the expected prize for the $m = 5$ strategy. Again let X denote the number on the dice when we finish rolling. This time the possible values of X are 1, 5 or 6. We'll just appeal to the informal argument for the distribution of X because it's faster and happens to be correct. When we stop we roll either a 1, 5 or 6 and we are equally likely to roll any of these numbers, so $P(X = 1) = P(X = 5) = P(X = 6) = 1/3$, giving
For $m = 5$:

$$\text{Expected prize} = \frac{1}{3}(1 + 5 + 6) = \frac{12}{3} = 4.$$

Similar arguments yield
For $m = 4$:

$$\text{Expected prize} = \frac{1}{4}(1 + 4 + 5 + 6) = \frac{16}{4} = 4.$$

For $m = 3$:

$$\text{Expected prize} = \frac{1}{5}(1 + 3 + 4 + 5 + 6) = \frac{19}{5}.$$

So we see that the strategies corresponding to $m = 4$ or $m = 5$ yield the highest expected prize, and these strategies are optimal in this sense. □

Digression on Examples:

You may have been noticing in the examples we've looked at so far in the course, that they are not all straightforward (e.g. the monkey example). If this is supposed to be probability review this week, you might be asking yourself "Am I expected to already know how to do all these examples?" The answer is no, at least not all of them. All of the examples involve only basic probability concepts but you're probably starting to realize that a problem can be difficult not because you don't know the concepts but because there's a certain level of sophistication in the solution method. The solutions are not always simple or direct applications of the concepts.

Much of mathematics, including much of the theory of probability, was developed in response to people posing, usually simply stated, problems that they were genuinely interested in. For example, the early development of probability theory was motivated by games of chance. The methods of applied probability, in particular, continue to be vigorously challenged by both old and new problems that people pose in our uncertain world. It's fair to say that the bulk of the work that goes on in probability is not in developing new general theory or new concepts. The language of probability has more or less already been sufficiently developed. Most of the work goes on in trying to solve particular problems originating in a wide variety of applications. So it's in the problems, and the solutions to those problems, that much of the learning and studying is to be done, and this fact is certainly reflected in our textbook.

So don't feel that there is something wrong if a solution to a problem is not obvious to you or if it takes some time and thought to follow even when it's given. That's supposed to happen. As you read the text you'll notice that the examples are not like typical examples in an introductory probability book. Many of the examples (and problems) required a lot of thought on the part of the author and other people before a solution was obtained. You get the benefit of all that work that went into the examples, but keep in mind that the examples and the problems are lessons in themselves over and above the main exposition of the text.

4

An Expectation Example

This week we'll start studying conditional expectation arguments in Section 3.4 of the text. Before doing so, we'll do one more example calculating an expectation. The problem and solution in this example is of the type alluded to in the previous lecture, meriting careful thought and study on its own. The solution also utilizes a useful quantity known as an *indicator function*.

Indicator Functions:

Indicator functions are very useful. For any set A the indicator function of the set A is

$$I_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{otherwise} \end{cases} .$$

One important property of the indicator function of A is that if A is an event on a sample space and $I_A(x)$ is a function on that sample space and there is a probability distribution P defined on that sample space, then I_A is a random variable with expectation

$$E[I_A] = (1)P(A) + (0)P(A^c) = P(A).$$

Example: *Matching Problem.* If n people throw their hats in a pile and then randomly pick up a hat from the pile, what is the probability that exactly r people retrieve their own hat?

Solution: First consider $r = n$, so that everyone retrieves their own hat. This case is relatively easy. The problem is equivalent to saying that we take a random permutation of the integers $1, \ldots, n$ and asking what is the probability that we choose one particular permutation (the one corresponding to all persons retrieving their own hat). There are a total of $n!$ possible permutations and only one corresponding to all persons retrieving their own hat, so the probability that everyone retrieves their own hat is $1/n!$.

Secondly, consider $r = n - 1$. This case is also easy, because it's logically impossible for exactly $n-1$ persons to retrieve their own hat, so the probability of this is 0.

For $0 \leq r \leq n-2$ the solution is not as trivial. There is more than one way to approach the problem, and we'll consider a solution that uses conditioning later in the week. Here we'll consider a more direct approach.

Let A_i be the event that person i retrieves his/her own hat. Note that $P(A_i) = 1/n$ for all i. We can see this because asking that the event A_i occur is logically the same thing as asking for a permutation of the integers $\{1, 2, \ldots, n\}$ that leaves the integer i in the ith position (i.e. i doesn't move). But we're allowed to permute the other $n-1$ integers any way we want, so we see that there are $(n-1)!$ permutations that leave integer i alone. So if we assume that all permutations are equally likely we see that $P(A_i) = (n-1)!/n! = 1/n$. In fact, using exact the same type of argument we can get the probability that any particular set of persons retrieves their own hat. Let $\{i_1, \ldots, i_s\}$ be a particular set of s persons (e.g. $s = 4$ and $\{2, 4, 5, 7\}$ are the persons).

Then leaving positions $i_1, \ldots, i_s$ alone, we can permute the remaining $n-s$ positions any way we want, for a total of $(n-s)!$ permutations that leave positions $i_1, \ldots, i_s$ alone. Therefore, the probability that persons $i_1, \ldots, i_s$ all retrieve their own hats is $(n-s)!/n!$. That is,

$$P(A_{i_1} \bigcap \ldots \bigcap A_{i_s}) = \frac{(n-s)!}{n!}.$$

However, these are not quite the probabilities that we're asking about (though we'll want to use them eventually, so remember them), because if we take $s = r$ and consider the above event that persons $i_1, \ldots, i_r$ all retrieved their own hat, this event doesn't preclude the possiblity that other persons (or even everyone) also retrieved their own hat.

What we want are the probabilities of events like the following:

$$E_{(i_1,\ldots,i_n)} = A_{i_1} \bigcap \ldots \bigcap A_{i_r} \bigcap A^c_{i_{r+1}} \bigcap \ldots \bigcap A^c_{i_n},$$

where $(i_1, \ldots, i_n)$ is some permutation of $(1, \ldots, n)$. Event $E_{(i_1,\ldots,i_n)}$ says that persons $i_1, \ldots, i_r$ retrieved their own hat but that persons $i_{r+1}, \ldots, i_n$ did not. For a particular $(i_1, \ldots, i_n)$, that would be one way for the event of r persons retrieving their own hat to occur. These events $E_{(i_1,\ldots,i_n)}$ are the right events to be considering, because as we let $(i_1, \ldots, i_n)$ vary over all possible permutations, we get all the possible ways for exactly r persons to retrieve their own hat. However, here we need to be careful in our counting, because as we vary $(i_1, \ldots, i_n)$ over all possible permutations we are doing some multiple counting of the same event. For example, suppose $n = 5$ and $r = 3$. Then, if you go examine the way we've defined the event $E_{(i_1,i_2,i_3,i_4,i_5)}$ in general, you'll see that the event $E_{(1,2,3,4,5)}$ is the same as the event $E_{(3,2,1,4,5)}$ or the event $E_{(3,2,1,5,4)}$.

In general, if we have a particular permutation $(i_1, \ldots, i_n)$ and consider the event $E_{(i_1,\ldots,i_n)}$, we can permute the first r positions any way we want and also permute the last $n-r$ positions any way we want and we'll still end up with the same event. Since there are $r!$ ways to permute the first r positions and for each of these ways there are $(n-r)!$ ways to permute the last $n-r$ positions, in total there are $r!(n-r)!$ permutations of $(i_1,\ldots,i_n)$ that lead to the same event $E_{(i_1,\ldots,i_n)}$. So that means if we sum up $P(E_{(i_1,\ldots,i_n)})$ over all $n!$ possible permutations of all n positions, then we should divide that sum by $r!(n-r)!$ and we should end up with the right answer.

The next step is to realize that the events $E_{(i_1,\ldots,i_n)}$ all have the same probability no matter what the permutation $(i_1,\ldots,i_n)$ is. This is so because all permutations are equally likely. This sort of symmetry reasoning is a very valuable method for simplifying calculations in problems of this sort and its important to get a feel for when you can apply this kind of reasoning by getting practice applying it in problems. This symmetry immediately simplifies our calculation, because when we sum $P(E_{(i_1,\ldots,i_n)})$ over all possible permutations, the answer can be given in terms of any particular permutation, and in particular

$$\sum_{(i_1,\ldots,i_n)} P(E_{(i_1,\ldots,i_n)}) = n!P(E_{(1,\ldots,n)}).$$

So now if we divide this by $r!(n-r)!$ we should get the right answer:

$$P(\text{exactly } r \text{ persons retrieve their own hat}) = \frac{n!}{r!(n-r)!}P(E_{(1,\ldots,n)}),$$

and now the problem is to figure what is the probability of $E_{(1,\ldots,n)}$, which is the event that persons $1,\ldots,r$ retrieve their own hat and persons $r+1,\ldots,n$ do not.

Now (you were probably wondering when we would get to them) we'll introduce the use of indicator functions. In the interest of a more compact notation, let I_j denote I_{A_j}, the indicator of the event that person j retrieves his/her own hat. Two useful properties of indicator functions are the following. If I_A and I_B are two indicator functions for the events A and B, respectively, then

$$\begin{aligned} I_A I_B &= I_{A\bigcap B} = \text{the indicator of the intersection of } A \text{ and } B\text{, and} \\ 1 - I_A &= I_{A^c} = \text{the indicator of the complement of } A. \end{aligned}$$

Using these two properties repeatedly, we get that the indicator of the event $E_{(1,\ldots,n)}$ (go back and look at the definition of this event) is

$$I_{E_{(1,\ldots,n)}} = I_1 \ldots I_r(1 - I_{r+1}) \ldots (1 - I_n).$$

Now you may be wondering how this helps? Well, it helps because we've converted a set expression $A_1 \bigcap \ldots \bigcap A_r \bigcap A^c_{r+1} \bigcap \ldots \bigcap A^c_n$ into an arithmetic expression $I_1 \ldots I_r(1 - I_{r+1}) \ldots (1 - I_n)$ (containing random variables), related by the fact the probability of the set expression is equal to the expected value of the arithmetic expression. But this is useful because now we can apply ordinary arithmetic operations to the arithmetic expression. In particular, we will expand out $(1 - I_{r+1}) \ldots (1 - I_n)$. Can you see why we might want to do this?

So how do we do this? I think we'll just have to do some multiplying and see what we get. Let's simplify notation a little bit first. Suppose $a_1, \ldots, a_k$ are any k numbers and let's ask how we expand out the product $(1 - a_1) \ldots (1 - a_k)$. Starting out easy, suppose $k = 2$. Then we get

$$(1 - a_1)(1 - a_2) = 1 - a_1 - a_2 + a_1 a_2.$$

What if we now multiply that by $(1 - a_3)$? Well, we get

$$1 - a_1 - a_2 + a_1 a_2 - a_3 + a_1 a_3 + a_2 a_3 - a_1 a_2 a_3.$$

What if we now multiply this by $(1 - a_4)$? No, I don't want to write it out either. It's time we looked for a pattern. This is the kind of thing that turns up on IQ tests. I claim that the pattern is

$$\prod_{i=1}^{k}(1-a_i)=\sum_{s=0}^{k}(-1)^s\sum_{1\le i_1<\ldots<i_s\le k}a_{i_1}\ldots a_{i_s},$$

where the first term (when $s=0$) is meant to be a 1 and in the remaining terms (for $s>0$) the s indices $i_1,\ldots,i_s$ are to run from 1 to k, with the constraint that $i_1<\ldots<i_s$. But is the above equality true? In fact it is. One way to prove it would be to use induction. You can check that it's true for the cases when $k=2$ and $k=3$. Then we assume the expression is correct for any fixed k, then multiply the expression by $(1-a_{k+1})$ and see that it's true for the $k+1$ case. Then logically it must be true for all k. I'll leave that for you to do as an exercise. Right now it's probably better to go back to our original problem, which is to multiply out $(1-I_{r+1})\ldots(1-I_n)$. If we get all the indices straight, we get that

$$(1-I_{r+1})\ldots(1-I_n)=\sum_{s=0}^{n-r}(-1)^s\sum_{r+1\le i_1<\ldots i_s\le n}I_{i_1}\ldots I_{i_s}.$$

Now you can see why we wanted to expand out the above product. It's because each of the terms in the sum corresponds to an intersection of the some of the events A_i directly, with no complements in the intersection. And this will still be true when we multiply it all by $I_1\ldots I_r$. We want this because when we take the expectation, what we'll want to know is the probability that a given set of persons retrieved their own hats. But we know how to do this. Remember?

So we'll take the above expression, multiply it by $I_1\ldots I_r$, and take the expectation, and what we end up with is $P(E_{(1,\ldots,n)})$, and from there our final answer is just one multiplication away.

$$\begin{aligned} P(E_{(1,\ldots,n)}) &= E[I_1 \ldots I_r(1-I_{r+1})\ldots(1-I_n)] \\ &= E\Big[I_1\ldots I_r\sum_{s=0}^{n-r}(-1)^s \sum_{r+1\le i_1<\ldots<i_s\le n} I_{i_1}\ldots I_{i_s}\Big] \\ &= \sum_{s=0}^{n-r}(-1)^s \sum_{r+1\le i_1<\ldots<i_s\le n} E[I_1\ldots I_r I_{i_1}\ldots I_{i_s}]. \end{aligned}$$

Let's pause here for a moment. The final line above comes about by taking the expectation inside both summations. We can always take expectations inside summations because of the basic linearity property of expectation (but don't make the mistake that you can always take expectation inside of products). Now we also know the value of each of the expectations above, from way back near the beginning of this solution. For a given s, there are $r+s$ indicator functions inside the expectation. The expectation in a given term in the sum is just the probability that a particular set of $r+s$ persons (persons $1,\ldots,r,i_1,\ldots,i_s$) retrieved their own hat, and we've already calculated that this probability is given by $(n-(r+s))!/n!$. So we know that

$$P(E_{(1,\ldots,n)}) = \sum_{s=0}^{n-r}(-1)^s \sum_{r+1\le i_1<\ldots<i_s\le n} \frac{(n-r-s)!}{n!}.$$

We can certainly simplify this, because we can take the term $(n-r-s)!/n!$ outside of the inner sum because this term depends only on s, not on the particular indices $i_1,\ldots,i_s$. So now the question is: for a given s how many terms are there in the inner sum? This is a counting question again. The question is how many ways are there to pick s integers from the integers $r+1,\ldots,n$. There are $n-r$ integers in the set $\{r+1,\ldots,n\}$ so there are $\binom{n-r}{s}$ ways to pick s integers from these. So let's use this and simplify further:

$$\begin{aligned}
P(E_{(1,\ldots,n)}) &= \sum_{s=0}^{n-r}(-1)^s\frac{(n-r-s)!}{n!}\binom{n-r}{s} \\
&= \sum_{s=0}^{n-r}(-1)^s\frac{(n-r-s)!}{n!}\frac{(n-r)!}{(n-r-s)!s!} \\
&= \sum_{s=0}^{n-r}(-1)^s\frac{(n-r)!}{n!s!}.
\end{aligned}$$

Now we are basically done except to write the final answer. Recall (from a few pages ago) that

$$P(\text{exactly } r \text{ persons retrieve their own hat}) = \frac{n!}{r!(n-r)!}P(E_{(1,\ldots,n)})$$

so that now we can write

$$\begin{aligned}
&P\,(\text{exactly } r \text{ persons retrieve their own hat}) \\
&\quad = \frac{n!}{r!(n-r)!}\sum_{s=0}^{n-r}(-1)^s\frac{(n-r)!}{n!s!} = \frac{1}{r!}\sum_{s=0}^{n-r}(-1)^s\frac{1}{s!}.
\end{aligned}$$

Finally, we can tweak the answer just a little bit more, because the first two terms in the sum above are 1 and -1, so they cancel, and so

$$\begin{gathered}
P\,(\text{exactly } r \text{ persons retrieve their own hat}) \\
= \frac{1}{r!}\left(\frac{1}{2!}-\frac{1}{3!}+\ldots+\frac{(-1)^{n-r}}{(n-r)!}\right).
\end{gathered}$$

This answer is valid for $r \le n-2$. This answer has an interesting form when $r = 0$. When n is large, the probability that nobody retrieves their own hat is approximately e^{-1}. Intuitive? Surprising?

5

Conditional Expectation

Conditional Expectation

Recall that given two events A and B with $P(B) > 0$, the conditional probability of A given B, written $P(A|B)$, is defined as

$$P(A|B) = \frac{P(A \bigcap B)}{P(B)}.$$

Given two (discrete) random variables X and Y the conditional expectation of X given $Y = y$, where $P(Y = y) > 0$, is defined to be

$$E[X|Y = y] = \sum_x xP(X = x|Y = y).$$

Note that this is just the mean of the conditional distribution of X given $Y = y$.

Conditioning on an event has the interpretation of information, in that knowing the event $\{Y = y\}$ occured gives us information about the likelihood of the outcomes of X. There is a certain art or intuitiveness to knowing when conditioning on an event will give us useful information or not, that one can develop with practice. Sometimes though it's fairly obvious when conditioning will be helpful.

Example: Suppose that in 1 week a chicken will lay N eggs, where

N is a random variable with a Poisson(λ) distribution. Also each egg has probability p of hatching a chick. Let X be the number of chicks hatched in a week. What is $E[X]$?

The point being illustrated here is that the distribution of X is not quite obvious but if we condition on the event $\{N = n\}$ for some fixed n, then the distribution of X is clearly Binomial(n, p), so that while $E[X]$ may not be obvious, $E[X|N = n]$ is easily seen to be np.

In $E[X|Y = y]$, the quantity y is used to denote any particular value that the random variable Y takes on, but it's in lowercase because it is meant to be a fixed value, just some fixed number. Similarly, $E[X|Y = y]$ is also just some number, and as we change the value of y the conditional expectation $E[X|Y = y]$ changes also. A very useful thing happens when we take the weighted average of all these numbers, weighted by the probability distribution of Y. That is, when we consider

$$\sum_y E[X|Y = y]P(Y = y).$$

When we plug in the definition of $E[X|Y = y]$ and work out the sum we get

$$\begin{aligned}
\sum_y E[X|Y = y]P(Y = y) &= \sum_y \sum_x xP(X = x|Y = y)P(Y = y) \\
&= \sum_y \sum_x x\frac{P(X = x, Y = y)}{P(Y = y)}P(Y = y) \\
&= \sum_x x \sum_y P(X = x, Y = y) \\
&= \sum_x xP(X = x) = E[X].
\end{aligned}$$

In words, when we average the conditional expectation of X given $Y = y$ over the possible values of y weighted by their marginal probabilities, we get the marginal expectation of X. This result is so useful let's write it again:

$$E[X] = \sum_y E[X|Y = y]P(Y = y).$$

It's also important enough to have a name:

The Law of Total Expectation.

Note that the quantity $\sum_y E[X|Y = y]P(Y = y)$ is also an expectation (with respect to the distribution of Y). It's the expected value of a function of Y, which we'll just call $g(Y)$ for now, whose value when $Y = y$ is $g(y) = E[X|Y = y]$. What we usually do, even though it's sometimes confusing for students, is to use $E[X|Y]$ to denote this function of Y, rather than using something like $g(Y)$. The notation $E[X|Y]$ is actually quite natural, but it can be confusing because it relies rather explicitly on the usual convention that upper case letters denote random variables while lower case letters denote fixed values that the random variable might assume. In particular, $E[X|Y]$ is not a number, it's a function of Y as we said, and as such is itself a random variable. It's also potentially confusing because we refer to the quantity $E[X|Y]$ as the conditional expectation of X given Y, and this is almost the same way we refer to the quantity $E[X|Y = y]$. Also, it may take some getting used to thinking of a conditional expectation as a random variable, and you may be alarmed when we first write something like $E[E[X|Y]]$. The more compact and efficient way to write the Law of Total Expectation is

$$E[X] = E[E[X|Y]],$$

where the inner expectation is taken with respect to the conditional distribution of X given $Y = y$ and the outer expectation is taken with respect to the marginal distribution of Y.

Example: *Chicken and Eggs.* Since we know $E[X|N = n] = np$ we have that $E[X|N] = Np$ and so

$$E[X] = E[E[X|N]] = E[Np] = pE[N] = p\lambda.$$

Now let's continue with looking at more examples of computing expectations using conditioning arguments via the Law of Total Expectation. Of course to use a conditioning argument you have to have two random variables, the one whose expectation you want and the one you are conditioning on. Note though that the quantity you are conditioning on can be more general than a random variable. It can for example be a random vector or it could be the outcome of a general sample space, not necessarily even numeric. In addition, conditioning arguments are often used in the context of a whole sequence of random variables (such as a stochastic process, hint hint), but the use of the Law of Total Expectation is only part of the argument. The conditioning doesn't give us the answer directly, but it does give us an equation or set of equations involving expectation(s) of interest that we can solve. Here's an example of this kind of argument which takes us back to one of our earlier expectation examples.

Example: Use a conditional expectation argument to find the mean of the Geometric(p) distribution.

Solution: Let X be a random variable with the Geometric(p) distribution. To use conditioning recall that the distribution of X has a description in terms of a sequence of independent Bernoulli trials. Namely, each trial has probability p of "success" and X is the time (the trial number) of the first success. Now condition on the outcome of the first trial. The Law of Total Expectation then gives us that

$$\begin{aligned}
E[X] &= E[E[X|\text{outcome of first trial}]] \\
&= E[X|S \text{ on 1st trial}]P(S \text{ on 1st trial}) \\
&\qquad\quad + E[X|F \text{ on 1st trial}]P(F \text{ on 1st trial}) \\
&= E[X|S \text{ on 1st trial}]p + E[X|F \text{ on 1st trial}](1-p).
\end{aligned}$$

Now the argument proceeds as follows. Given that there is a success on the first trial, X is identically equal to 1. Therefore,

$$E[X|S \text{ on 1st trial}] = 1.$$

The crucial part of the argument is recognizing what happens when we condition on there being a failure in the first trial. If this happens, then X is equal to 1 (for the first trial) *plus* the number of additional trials needed for the first success. But the number of additional trials required for the first success *has the same distribution* as X, namely a Geometric(p) distribution. This is so because all trials are identically distributed and independent. To write out this argument more formally and mathematically we might write the following. Let Y be defined as the first trial index, *starting from the second trial*, that we have a success, minus 1. Then the distribution of Y is the same as the distribution of X and, in fact, Y is independent of the outcome of the first trial. However, given that the first trial is a failure, the conditional distribution of X is the same as the distribution of $1 + Y$. Therefore,

$$E[X|F \text{ on 1st trial}] \; = \; E[1 + Y] = 1 + E[Y] = 1 + E[X],$$

where $E[Y] = E[X]$ because X and Y have the same distribution. We've just connected a circle, relating the conditional expectation $E[X|F \text{ on 1st trial}]$ back to the original unconditional expectation of interest, $E[X]$. Putting this back into our original equation for $E[X]$, we have

$$E[X] \; = \; (1)p + (1 + E[X])(1 - p).$$

Now it's easy to solve for $E[X]$, giving $E[X] = 1/p$.

I claim that the preceding method for evaluating $E[X]$ is more elegant than the way we calculated it last week using directly the definition of expectation and evaluating the resulting infinite sum, because the conditioning argument is more natural and intuitive and is a purely probabilistic argument. We can do a similar calculation to calculate $E[X^2]$ (without any words this time to clutter the elegance):

$$\begin{aligned}
E[X^2] &= E[X^2|S \text{ on 1st trial}]P(S \text{ on 1st trial}) \\
&\qquad\qquad + E[X^2|F \text{ on 1st trial}]P(F \text{ on 1st trial}) \\
&= E[X^2|S \text{ on 1st trial}]p + E[X^2|F \text{ on 1st trial}](1-p) \\
&= (1)^2(p) + E[(1+Y)^2](1-p) \\
&= p + E[(1+X)^2](1-p) \\
&= p + E[1+2X+X^2](1-p) \\
&= p + (1+2E[X]+E[X^2])(1-p) \\
&= 1 + \frac{2(1-p)}{p} + E[X^2](1-p).
\end{aligned}$$

Solving for $E[X^2]$ gives

$$E[X^2] = \frac{1}{p} + \frac{2}{p^2} - \frac{2}{p} = \frac{2}{p^2} - \frac{1}{p}.$$

(Check that this is the same answer we obtained by direct calculation last week).

Finding a Conditional Expectation by Conditioning

Note that the Law of Total Expectation can be used to find, at least in principle, the mean of any distribution. This includes the *conditional* distribution of X given $Y = y$. That is,

$$E[X|Y = y]$$

is the mean of a distribution, just like $E[X]$ is, so it too can be computed by conditioning on another random variable, say Z. However, *we must use the conditional distribution of Z given $Y = y$, and not the marginal distribution of Z when we do the weighted averaging:*

$$E[X|Y = y] = \sum_z E[X|Y = y, Z = z]P(Z = z|Y = y).$$

Example: Suppose we roll a fair die and then flip a fair coin the number of times showing on our die roll. If any heads are flipped when we've finished flipping the coin we stop. Otherwise we keep repeating the above experiment until we've flipped at least one heads. What's the expected number of flips we make before we stop?

Solution: Let X be the number of flips before we stop and let Y be the outcome of the first roll of the die. Then

$$E[X] = \sum_{k=1}^{6} E[X|Y = k] \times \frac{1}{6}$$

Now we compute $E[X|Y = k]$ by conditioning on whether or not there are any heads when we flip the coin k times. Let $Z = 1$ if there is at least one heads in our first set of coin flips and $Z = 0$ if there are no heads in our first set of coin flips. Then

$$\begin{aligned} E[X|Y = k] &= E[X|Y = k, Z = 1]P(Z = 1|Y = k) \\ &\quad + E[X|Y = k, Z = 0]P(Z = 0|Y = k). \end{aligned}$$

Now $E[X|Y = k, Z = 1] = k$ because we will stop the experiment after the k flips since we've had at least one heads. But

$$E[X|Y = k, Z = 0] = k + E[X]$$

because we flip the coin k times and then probabilistically restart the experiment over again. So we have

$$\begin{aligned} E[X|Y = k] &= kP(Z = 1|Y = k) + (k + E[X])P(Z = 0|Y = k) \\ &= k + E[X]P(Z = 0|Y = k) \\ &= k + E[X]\left(\frac{1}{2}\right)^k, \end{aligned}$$

since $P(Z = 0|Y = k)$ is the probability of k consecutive tails. Plugging this back in to our original expression for $E[X]$, we have

$$\begin{aligned} E[X] &= \frac{1}{6}\sum_{k=1}^{6}\left[k + E[X]\left(\frac{1}{2}\right)^k\right] \\ &= \frac{21}{6} + \frac{E[X]}{6} \times \frac{63}{64}. \end{aligned}$$

Solving for $E[X]$ then gives

$$E[X] = \frac{21}{6} \times \frac{(6)(64)}{63} = \frac{64}{3}.$$

6

Quicksort Algorithm Example

In the previous examples we saw that conditioning allowed us to derive an equation for the particular expectation of interest. In more elaborate situations, we may want know about more than one unknown expectation, possibly an infinite number of unknown expectations. In such situations the unknown expectations are usually "of the same kind", and sometimes conditioning allows us to derive not just one, but multiple equations that allow us to solve for all the unknown expectations. The following is a nice example from the text which analyzes the computational complexity of what is probably the most common sorting algorithm, the *Quicksort Algorithm*.

Example: *The Quicksort Algorithm.* Given n values, a classical programming problem is to efficiently sort these values in increasing order. One of the most efficient and widely used sorting algorithms for doing this is called the Quicksort Algorithm, which is described as follows.

```
Procedure QUICKSORT, Inputs $n$ and $\{x_1, \ldots, x_n\}$.
If $n = 0$ or $n = 1$ Stop. Return "no sorting needed" flag.
Else {
    • Choose one of the values at random.
    • Compare each of the remaining values to the value chosen and
      divide up the remaining values into two sets, $L$ and $H$, where $L$
      is the set of values less than the chosen value and $H$ is the set
      of values higher than the chosen value (if one of the remaining
      values is equal to the chosen value then assign it to either $L$ or
      $H$ arbitrarily).
    • Apply procedure QUICKSORT to $L$.
    • Apply procedure QUICKSORT to $H$.
    • Return with sorted list.
}
```

Note that the procedure QUICKSORT is applied *recursively* to the sets L and H. As an example, suppose we are given the 7 values $\{6, 2, 3, 9, 1, 3, 5\}$ and we wish to sort them in increasing order. First we choose one of the values at random. Suppose we choose the first 3. When we compare every other value to 3 we get the sets $L = \{2, 1, 3\}$ and $H = \{6, 9, 5\}$ (where we arbitrarily have assigned the second 3 to the L set). So far the numbers have been sorted to the following extent:

$$\{2, 1, 3\}, 3, \{6, 9, 5\}.$$

Let's also keep track of the number of comparisons we make as we go. So far we've made 6 comparisons. Now we apply QUICKSORT to the set $L = \{2, 1, 3\}$. Suppose we (randomly) pick the value 1. Then we divide up L into the two sets $\{\}$ (the empty set) and $\{2, 3\}$. This required 2 more comparisons (for a total of 8 so far), and the set L has so far been sorted into

$$\{\}, 1, \{2, 3\}.$$

Now the empty set $\{\}$ is passed to QUICKSORT and it immediately returns with no comparisons made. Then the set $\{2, 3\}$ is passed to QUICKSORT. One can see that in this call to QUICKSORT 1 more comparison will be made (for a total of 9 so far), then from this call QUICKSORT will be called two more times and immediately return both times, and the call to QUICKSORT with the set L will have finished with L sorted. Now control gets passed to the top level and QUICKSORT is called with the set $H = \{6, 9, 5\}$. Now suppose the value if 5 is picked. Without going into the details again, QUICKSORT will be called 4 more times and 3 more comparisons will be made (for a total now of 12 comparisons). At this point H will be sorted and the entire original list will be sorted.

A natural way to measure the efficiency of the algorithm is to count how many comparisons it must make to sort n values. However, for the QUICKSORT algorithm, note that the number of comparisons is a random variable, because of the randomness involved in selecting the value which will separate the low and high sets. The worst case occurs if we always select the lowest value in the set (or the highest value). For example if our original set has 7 values then initially we make 6 comparisons. But if we picked the lowest value to compare to, then the set H will have 6 values, and when we call QUICKSORT again, we'll need to make 5 more comparisons. If we again choose the lowest value in H then QUICKSORT will be called with a set containing 5 values and 4 more comparisons will need to be made. If we always (by bad luck) choose the lowest value, then in total we'll end up making $6 + 5 + 4 + 3 + 2 + 1 = 21$ comparisons. The best case occurs when we (by good luck) always choose the "middle" value in whatever set is passed to QUICKSORT.

So the number of comparisons QUICKSORT makes to sort a list of n values is a random variable. Let X_n denote the number of comparisons required to sort a list of n values and let $M_n = E[X_n]$. As we noted last week, we may use the expected value, M_n, as a measure of the performance or efficiency of the QUICKSORT algorithm.

In this example we pose the problem of determining $M_n = E[X_n]$ for $n = 0, 1, 2, \ldots$. To simplify things a little bit, we'll assume that the list of numbers to sort have no tied values, so there is no ambiguity about how the algorithm proceeds. We have an infinite number of unknown expectations to determine, but they are all "of the same kind". Furthermore, the recursive nature of the QUICKSORT algorithm suggests that a conditioning argument may be useful. But what should we condition on?

Solution: First we note that no comparisons are required to sort a set of 0 or 1 elements, because the QUICKSORT procedure will return immediately in these cases. Thus, X_0 and X_1 are both equal to 0, and so $M_0 = E[X_0]$ and $M_1 = E[X_1]$ are both equal to 0 as well. We may also note that $X_2 = 1$ so that $M_2 = E[X_2] = 1$ (though it will turn out that we won't need to know this). For $n \geq 3$, X_n is indeed random. If we are going to use a conditioning argument to compute $M_n = E[X_n]$ in general, it is natural to consider how the QUICKSORT algorithm proceeds. The first thing it does is randomly pick one of the n numbers that are to be sorted. Let Y denote the *rank* of the number that is picked. Thus $Y = 1$ if we pick the smallest value, $Y = 2$ if we pick the second smallest value, and so on. Since we select the number at random, all ranks are equally likely. That is, $P(Y = j) = 1/n$ for $j = 1, \ldots, n$.

Why might we want to condition on the rank of the chosen value? What information do we get by doing so? If we know the rank of the chosen value then we'll know the size of each of the sets L and H and so we'll be able to express the expected total number of comparisons in terms of the expected number of comparisons it takes to sort the elements in L and in H. Let's proceed.

First we condition on Y and use the Law of Total Expectation to write

$$M_n = E[X_n] = E[E[X_n|Y]] = \sum_{j=1}^{n} E[X_n|Y = j]\frac{1}{n}.$$

Now consider $E[X_n|Y = j]$. Firstly, no matter what the value of Y is, we will make $n - 1$ comparisons in order to form our sets L and H. But if $Y = j$, then L will have $j - 1$ elements and H will have $n - j$ elements. Then we apply the QUICKSORT procedure to L then to H. Since L has $j - 1$ elements, the number of comparisons to sort the elements in L is X_{j-1} and since the number of elements in H is $n-j$, the number of comparisons to sort the elements in H is X_{n-j}. Thus given $Y=j$, the distribution of X_n is the same as the distribution of $n-1+X_{j-1}+X_{n-j}$, and so

$$\begin{aligned} E[X_n|Y=j] &= n-1+E[X_{j-1}]+E[X_{n-j}] \\ &= n-1+M_{j-1}+M_{n-j}. \end{aligned}$$

Plugging this back into our expression for M_n we have

$$M_n = \frac{1}{n}\sum_{j=1}^{n}(n-1+M_{j-1}+M_{n-j}).$$

At this point we can see we are getting somewhere and we probably did the right thing in conditioning on Y, because we have derived a set of (linear) equations for the quantities M_n. They don't look too bad and we have some hope of solving them for the unknown quantities M_n. In fact we can solve them and the rest of the solution is now solving a linear algebra problem. All the probability arguments in the solution are now over. While the probability argument is the main thing I want you to learn from this example, we still have a linear system to solve. Indeed, throughout this course, we'll see that some conditioning argument will lead to a system of linear equations that needs to be solved, so now is a good time to get some practice simplifying linear systems.

The first thing to do is take the sum through in the expression for M_n, which gives

$$M_n = \frac{n-1}{n}\sum_{j=1}^{n}(1) + \frac{1}{n}\sum_{j=1}^{n}M_{j-1} + \frac{1}{n}\sum_{j=1}^{n}M_{n-j}.$$

This simplifies somewhat. In the first sum, when we sum the value one n times we get n, and this cancels with the n in the denominator, so the first term above is $n-1$. Next, in the second sum, the values of $j-1$ range from 0 to $n-1$, and in the third sum the values of $n-j$ also range from 0 to $n-1$. So in fact the second and third sums are the same quantity, and we have

$$M_n = n-1+\frac{2}{n}\sum_{j=0}^{n-1} M_j.$$

We see that M_n can be obtained recursively in terms of M_j's with smaller indices j. Moreover, the M_j's appear in the recursion by simply summing them up. The usual way to simplify such a recursive equation is to first get rid of the sum by taking two successive equations (for n and $n+1$) and then subtracting one from the other. But first we need to isolate the sum so that it is not multiplied by any term containing n. So we multiply through by n to obtain

$$nM_n = n(n-1)+2\sum_{j=0}^{n-1} M_j.$$

Now we write the above equation with n replaced by $n+1$:

$$(n+1)M_{n+1} = (n+1)n+2\sum_{j=0}^{n} M_j$$

and then take the difference of the two equations above:

$$\begin{aligned}(n+1)M_{n+1}-nM_n &= (n+1)n-n(n-1)+2M_n\\ &= 2n+2M_n,\end{aligned}$$

or

$$(n+1)M_{n+1} = 2n+(n+2)M_n.$$

So we have simplified the equations quite a bit, and now we can think about actually solving them. If you stare at the above for a little while you may see that we can make it simpler still. If we divide by $(n+1)(n+2)$ we get

$$\frac{M_{n+1}}{n+2} = \frac{2n}{(n+1)(n+2)} + \frac{M_n}{n+1}.$$

Why is this simpler? It's because the quantities involving M_n and M_{n+1} are now of the same *form*. If we define

$$R_n = \frac{M_n}{n+1}$$

then the equation is equal to

$$R_{n+1} = \frac{2n}{(n+1)(n+2)} + R_n.$$

Now we can successively replace R_n with a similar expression involving R_{n-1}, then replace R_{n-1} with a similar expression involving R_{n-2}, and so on, as follows:

$$\begin{aligned} R_{n+1} &= \frac{2n}{(n+1)(n+2)} + R_n \\ &= \frac{2n}{(n+1)(n+2)} + \frac{2(n-1)}{n(n+1)} + R_{n-2} \\ &\vdots \\ &= \sum_{j=0}^{n-1} \frac{2(n-j)}{(n+1-j)(n+2-j)} + R_1 \\ &= \sum_{j=0}^{n-1} \frac{2(n-j)}{(n+1-j)(n+2-j)}, \end{aligned}$$

because $R_1 = M_1/2 = 0$ as we discussed earlier. We're basically done, but when you have a final answer its good practice to present the final answer in as readable a form as possible. Here, we can make the expression more readable if we make the substitution $i = n - j$. As j ranges between 0 and $n - 1$ i will range between 1 and n, and the expression now becomes

$$R_{n+1} = \sum_{i=1}^{n} \frac{2i}{(i+1)(i+2)}.$$

Finally, we should replace R_{n+1} by its definition in terms of M_{n+1} to obtain

$$\frac{M_{n+1}}{n+2} = \sum_{i=1}^{n} \frac{2i}{(i+1)(i+2)},$$

or

$$M_{n+1} = (n+2) \sum_{i=1}^{n} \frac{2i}{(i+1)(i+2)}.$$

So after a bit of tweaking we have our final answer. □

There is another important thing we haven't done in this example. The quantity M_{n+1} is the expected number of comparisons that the QUICKSORT algorithm makes to sort a list of $n+1$ (distinct) numbers, and so is a measure of how long the algorithm will take to complete its task. In computer science, we often talk about the *complexity* of an algorithm. But rather than a specific equation like the one for M_{n+1} above, what we are usually interested in is the *order of complexity* of the algorithm. By this we mean that we want to know how fast does the complexity grow with n? For example, if it grows exponentially with n (like $\exp(an)$) then that's usually bad news. Complexity that grows like n (the complexity is like $an + b$) is usually very good, and complexity that grows like n^2 is usually tolerable.

So what does the complexity of the QUICKSORT algorithm grow like? Let's write it out again for easy reference:

$$M_{n+1} = (n+2) \sum_{i=1}^{n} \frac{2i}{(i+1)(i+2)}.$$

It certainly seems to grow faster than n, because the expression is like n times a quantity (the sum) which has n terms. So perhaps it grows like n^2? Not quite. In fact it grows like $n \log n$, which is somewhere in between linear and quadratic complexity. This is quite good, which makes the QUICKSORT algorithm a very commonly used sorting algorithm. To see that it grows like $n \log n$ please refer to the text on p.116. Consider this a reading assignment. The important point for now is to introduce you to the concept of the *order* of an expression. We'll define what we mean by that more precisely in the coming weeks.

7

The List Model

We'll start today with one more example calculating an expectation using a conditional expectation argument. Then we'll look more closely at the Law of Total Probability, which is a special case of the Law of Total Expectation.

Example: *The List Model (Section 3.6.1).*
In information retrieval systems, items are often stored in a list. Suppose a list has n items $e_1, \ldots, e_n$ and that we know that item i will be requested with probability P_i, for $i = 1, \ldots, n$. The time it takes to retrieve an item from the list will be proportional to the position of the item in the list. It will take longer to retrieve items that are further down in the list. Ideally, we would like any requested item to be at the top of the list, but since we don't know in advance what item will be requested, it's not possible to ensure this ideal case. However, we may devise heuristic schemes to dynamically update the ordering of the list items depending on what items have been requested in the past.

Suppose, for example, that we use the *move to the front* rule. In this scheme when an item is requested it is moved to the front of the list, while the remaining items maintain their same relative ordering.

Other schemes might include the *move one forward* rule, where the position of a requested item is swapped with the position of the item immediately in front of it, or a *static ordering*, in which the ordering of the items in the list never changes once it is initially set. This static ordering scheme might be good if we knew what the request probabilities P_i were. For example, we could put the item most likely to be requested first in the list, the item second most likely to be requested second in the list, and so on. The advantage of the *move to the front* or *move one forward* rules is that we don't need to know the request probabilities P_i ahead of time in order to implement these schemes. In this example we'll analyze the *move to the front* scheme. Later, when we study Markov Chains in Chapter 4, we'll come back to this example and consider the *move one forward* scheme.

Initially, the items in the list are in some order. We'll allow it to be any arbitrary ordering. Let X_n be the position of the nth requested item. It's clear that if we know what the initial ordering is, then we can easily determine the distribution of X_1. What we would like to imagine though is that requests have been coming for a long time, in fact forever. We'll see in Chapter 4 that the distribution of X_n will approach some limiting distribution. If we say that X is a random variable with this limiting distribution, then the way we write this is

$$X_n \Rightarrow X \quad \text{in distribution.}$$

What this means is that

$$\lim_{n\to\infty} P(X_n = j) = P(X = j) \quad \text{for } j = 1, \ldots, n.$$

What we are interested in is $E[X]$.

Though we'll consider questions involving such limiting distributions more rigorously in Chapter 4, for now we'll approach the question more intuitively. Intuitively, the process has been running for a long time. Now some request for an item comes. The current position in the list of the item requested will be a random variable. We would like to know the expected position of the requested item.

Solution: First we'll condition on which item was requested. Since the item requested will be item i with probability P_i, we have

$$\begin{aligned} E[\text{position}] &= \sum_{i=1}^{n} E[\text{position}|e_i \text{ is requested}]P_i \\ &= \sum_{i=1}^{n} E[\text{position of } e_i]P_i. \end{aligned}$$

So we would like to know what is the expected position of item e_i at some time point far in the future. Conditioning has allowed us to focus on a specific item e_i, but have we really made the problem any simpler? We have, but we need to decompose the problem still further into quantities we might know how to compute. Further conditioning doesn't really help us much here. Conditioning is a good tool, but part of the effective use of conditioning is to know when not to use it.

Here we find the use of indicator functions to be helpful once again. Let I_j be the indicator of the event that item e_j is ahead of item e_i in the list. Since the position of e_i is just 1 plus the number of items that are ahead of it in the list, we can decompose the quantity "position of e_i" into

$$\text{position of } e_i = 1 + \sum_{j \neq i} I_j.$$

Therefore, when we take expectation, we have that

$$E[\text{position of } e_i] = 1 + \sum_{j \neq i} P(\text{item } e_j \text{ is ahead of item } e_i)$$

So we've decomposed our calculation somewhat. Now, for any particular i and j, we need to compute the probability that item e_j is ahead of item e_i. It may not be immediately apparent, but this we can do. This is because the only times items e_j and e_i ever change their relative ordering are the times when either e_j or e_i is requested. Whenever any other item is requested, items e_j and e_i do not change their relative ordering. So imagine that out of the sequence of all requests up to our current time, we only look at the requests that were for item e_j or for item e_i. Item e_j will currently be ahead of item e_i if and only if the last time item e_j or e_i was requested, it was item e_j that was requested. So the question is what is the probability that item e_j was requested the last time either item e_j or e_i was requested. In other words, given that item e_j or item e_i is requested, what is the probability that item e_j is requested. That is,

$$\begin{aligned}
&P\left(\text{item } e_j \text{ is ahead of item } e_i\right) \\
&\quad = P(\text{item } e_j \text{ is requested} \mid \text{item } e_j \text{ or } e_i \text{ is requested}) \\
&\quad = \frac{P(\{e_j \text{ requested}\} \bigcap \{e_j \text{ or } e_i \text{ requested}\})}{P(\{e_j \text{ or } e_i \text{ requested}\})} \\
&\quad = \frac{P(\{e_j \text{ requested}\})}{P(\{e_j \text{ or } e_i \text{ requested}\})} \\
&\quad = \frac{P_j}{P_j + P_i}.
\end{aligned}$$

So, plugging everything back in we get our final answer

$$\begin{aligned}
E[\text{position}] &= \sum_{i=1}^{n} E[\text{position of } e_i]P_i \\
&= \sum_{i=1}^{n}\Big(1+\sum_{j\neq i} P(\text{item } e_j \text{ is ahead of item } e_i)\Big)P_i \\
&= \sum_{i=1}^{n}\Big(1+\sum_{j\neq i}\frac{P_j}{P_i+P_j}\Big)P_i \\
&= 1+\sum_{i=1}^{n} P_i\sum_{j\neq i}\frac{P_j}{P_i+P_j}.
\end{aligned}$$

Note that in developing this solution, we are assuming that we are really far out into the future. One (intuitive) consequence of this is that when we consider the last time before our current time that either items e_i or e_j was requested, we are assuming that there was a last time. "Far out into the future" means far *enough* out (basically infinitely far out) that we are sure that items e_i and e_j had been requested many times (indeed infinitely many times) prior to the current time. □

One of the homework problems asks you to consider some fixed time t which is not necessarily far into the future. You are asked to compute the expected position of an item that is requested at time t. The solution proceeds much along the same lines as our current solution. However, for a specific time t you must allow for the possibility that neither items e_i or e_j was ever requested before time t. If you allow for this possibility, then in order to proceed with a solution, you must know what the initial ordering of the items is, or at least know the probabilities of the different possible orderings. In the homework problem it is assumed that all initial orderings are equally likely.

Calculating Probabilities by Conditioning (Section 3.5):

As noted last week, a special case of the Law of Total Expectation

$$E[X] = E[E[X|Y]] = \sum_y E[X|Y = y]P(Y = y)$$

is when the random variable X is the indicator of some event A. This special case is called the *Law of Total Probability*.

Since $E[I_A] = P(A)$ and $E[I_A|Y = y] = P(A|Y = y)$, we have

$$P(A) = \sum_y P(A|Y = y)P(Y = y).$$

Digression: Even though I promised not to look at continuous random variables for a while (till Chapter 5), I'd like to cheat a bit here and ask what the Law of Total Expectation looks like when Y is a continuous random variable. As you might expect, it looks like

$$E[X] = \int_y E[X|Y = y]f_Y(y)dy$$

where $f_Y(y)$ is the *probability density function* of Y. Similarly, the Law of Total Probability looks like

$$P(A) = \int_y P(A|Y = y)f_Y(y)dy$$

But wait! If Y is a continuous random variable, doesn't the event $\{Y = y\}$ have probability 0? So isn't the conditional probability $P(A|Y = y)$ undefined? Actually, it is defined, and textbooks (including this one, see Section 1.4) that tell you otherwise are not being quite truthful. Of course the definition is *not*

$$P(A|Y = y) = \frac{P(A, Y = y)}{P(Y = y)}.$$

Let's illustrate the Law of Total Probability (for discrete Y) with a fairly straightforward example.

Example: Let X_1 and X_2 be independent Geometric random variables with respective parameters p_1 and p_2. Find $P(|X_1 - X_2| \le 1)$.

Solution: We condition on either X_1 or X_2 (it doesn't matter which). Say we condition on X_2. Then note that

$$P(|X_1 - X_2| \le 1 \mid X_2 = j) = P(X_1 = j-1,\ j,\text{ or } j+1).$$

Thus,

$$\begin{aligned} P\left(|X_1 - X_2| \le 1\right) &= \sum_{j=1}^{\infty} P(|X_1 - X_2| \le 1 \mid X_2 = j) P(X_2 = j) \\ &= \sum_{j=1}^{\infty} P(X_1 = j-1,\ j,\text{ or } j+1) P(X_2 = j) \\ &= \sum_{j=1}^{\infty} \big[P(X_1 = j-1) + P(X_1 = j) + P(X_1 = j+1)\big] P(X_2 = j) \\ &= \sum_{j=2}^{\infty} p_1(1-p_1)^{j-2} p_2(1-p_2)^{j-1} + \sum_{j=1}^{\infty} p_1(1-p_1)^{j-1} p_2(1-p_2)^{j-1} \\ &\qquad + \sum_{j=1}^{\infty} p_1(1-p_1)^{j} p_2(1-p_2)^{j-1} \\ &= \frac{p_1p_2(1-p_2) + p_1p_2 + p_1p_2(1-p_1)}{1-(1-p_1)(1-p_2)}. \end{aligned}$$

How might you do this problem without using conditioning? □

8

Matching Problem Revisited

We'll do one more example of calculating a probability using conditioning by redoing the matching problem.

Example: *Matching Problem Revisited (Example 3.23).* Recall that we want to calculate the probability that exactly r persons retrieve their own hats when n persons throw their hats into the middle of a room and randomly retrieve them.

Solution: We start out by considering the case $r = 0$. That is, what is the probability that no one retrieves their own hat when there are n persons? As you search for a way to proceed with a conditioning argument, you start out by wondering what information would decompose the problem into conditional probabilities that might be simpler to compute? One of the first things that might dawn on you is that if you knew person 1 retrieved his own hat then the event that no one retrieved their own hat could not have happened. So there is some useful information there.

Next you would need to ask if you can determine the probability that person 1 retrieved his own hat. Yes, you can. Clearly, person 1 is equally likely to retrieve any of the n hats, so the probability that he retrieves his own hat is $1/n$. Next you need to wonder if you can calculate the probability that no one retrieved their own hat given that person 1 did not retrieve his own hat. This one is unclear perhaps. But in fact we can determine it, and here is how we proceed.

First we'll set up some notation. Let

$$E_n = \{\text{no one retrieves their own hat when there are } n \text{ persons}\}$$

and

$$P_n = P(E_n).$$

Also, let

$$Y = \text{The number of the hat picked up by person 1.}$$

Then

$$P(Y = j) = \frac{1}{n} \quad \text{for } j = 1, \ldots, n.$$

Conditioning on Y, we have

$$P_n = P(E_n) = \sum_{j=1}^{n} P(E_n | Y = j) \frac{1}{n}.$$

Now, $P(E_n | Y = 1) = 0$ as noted earlier, so

$$P_n = \sum_{j=2}^{n} P(E_n | Y = j) \frac{1}{n}.$$

At this point it's important to be able to see that $P(E_n|Y = 2)$ is the same as $P(E_n|Y = 3)$, and in fact $P(E_n|Y = j)$ is the same for all $j = 2, \ldots, n$. This is an example of spotting symmetry in an experiment and using it to simplify an expression. Symmetries in experiments are extremely useful because the human mind is somehow quite good at spotting them. The ability to recognize patterns is in fact one thing that humans are good at that even the most powerful supercomputers are perhaps only now getting the hang of. In probability the recognition of symmetries in experiments allows us to deduce that two or more expressions must be equal even if we have no clue as to what the actual value of the expressions is. We can spot the symmetry here because, as far as the event E_n is concerned, the indices $2, \ldots, n$ are just interchangeable labels. The only thing that matters as far as the event E_n is concerned is that person 1 picked up a *different* hat than his own.

With this symmetry, we can write, for example

$$P_n = \frac{n-1}{n} P(E_n|Y = 2),$$

Now let's consider $P(E_n|Y = 2)$. To determine this we might consider that it would help to know what hat person 2 picked up. We know person 2 didn't pick up his own hat because we know person 1 picked it up. Suppose we knew that person 2 picked up person 1's hat. Then we can see that persons 1 and 2 have formed a pair (they have picked up one another's hats) and the event E_n will occur now if persons 3 to n do not pick up their own hats, where these hats actually do belong to persons 3 to n. We have reduced the problem to one that is exactly of the same form as our original problem, but now involving only $n - 2$ persons (persons 3 through n).

So let's define Z to be the number of the hat picked up by person 2 and do one more level of conditioning to write

$$P(E_n|Y=2) = \sum_{\substack{k=1\\k\neq 2}}^{n} P(E_n|Y=2, Z=k)P(Z=k|Y=2).$$

Now given $Y = 2$ (person 1 picked up person 2's hat), person 2 is equally likely to have picked up any of remaining $n-1$ hats, so

$$P(Z=k|Y=2) = \frac{1}{n-1}$$

and so

$$P(E_n|Y=2) = \sum_{\substack{k=1\\k\neq 2}}^{n} P(E_n|Y=2, Z=k)\frac{1}{n-1}.$$

Furthermore, as we discussed on the previous page, if $Z = 1$ (and $Y = 2$), then we have reduced the problem to one involving $n-2$ persons, and so

$$P(E_n|Y=2, Z=1) = P(E_{n-2}) = P_{n-2}.$$

Plugging this back in we have

$$P(E_n|Y=2) = \frac{1}{n-1}P_{n-2} + \sum_{k=3}^{n} P(E_n|Y=2, Z=k)\frac{1}{n-1}.$$

Now we can argue once again by symmetry that $P(E_n|Y=2, Z=k)$ is the same for all $k = 3, \ldots, n$ because for these k the index is just an arbitrary label as far as the event E_n is concerned. So we have, for example

$$P(E_n|Y=2) = \frac{1}{n-1}P_{n-2} + \frac{n-2}{n-1}P(E_n|Y=2, Z=3).$$

Plugging this back into our expression for P_n we get

$$\begin{aligned} P_n &= \frac{n-1}{n} P(E_n|Y=2) \\ &= \frac{1}{n} P_{n-2} + \frac{n-2}{n} P(E_n|Y=2, Z=3). \end{aligned}$$

Now you might see that we can follow a similar line of argument to decompose $P(E_n|Y=2, Z=3)$ by conditioning on what hat person 3 picked up given that person 1 picked up person 2's hat and person 2 picked up person 3's hat. You can see it only matters whether person 3 picked up person 1's hat, in which case the problem is reduced to one involving $n-3$ people, or person 3 picked up person l's hat, for $l = 4, \ldots, n$. Indeed, this line of argument would lead to a correct answer, and is in fact equivalent to the the argument involving the notion of cycles in the Remark on p.120 of the text (also reproduced in the statement of Problem 3 on Homework #2).

You would proceed to find that

$$P(E_n|Y=2, Z=3) = \frac{1}{n-2} P_{n-3} + \frac{n-3}{n-2} P(E_n|Y=2, Z=3, Z'=4),$$

where Z' is the hat picked by person 3, so that

$$P_n = \frac{1}{n} P_{n-2} + \frac{1}{n} P_{n-3} + \frac{n-3}{n} P(E_n|Y=2, Z=3, Z'=4).$$

Continuing in this way you would end up with

$$\begin{aligned} P_n &= \frac{1}{n} P_{n-2} + \frac{1}{n} P_{n-3} + \ldots + \frac{1}{n} P_2 \\ &= \frac{1}{n} \sum_{k=2}^{n-2} P_k. \end{aligned}$$

However, I claim we could have stopped conditioning after our initial conditioning on Y because the probability $P(E_n|Y = 2)$ can be expressed in terms of the event E_{n-1} and E_{n-2} more directly. Here's why. Person 1 picked up person 2's hat. Suppose we relabel person 1's hat and pretend that it belongs to person 2 (but with the understanding that if person 2 picks up his "own" hat it's really person 1's hat). Then the event E_n will occur if either of the events

$$\{\text{persons 2 to } n \text{ do not pick up their own hat}\}$$

or

$$\{\text{person 2 picks up his "own" hat and persons 3 to } n \text{ do not}\}$$

occurs, and these two events are mutually disjoint. The probability of the first event (given $Y = 2$) is

$$\begin{aligned} & P\,(\text{persons 2 to } n \text{ do no pick up their own hat}|Y = 2) \\ = \ & P(E_{n-1}) = P_{n-1} \end{aligned}$$

while the probability of the second event given $Y = 2$ we can write as

$$\begin{aligned} & P\,(\text{person 2 picks up his "own" hat and persons 3 to } n \text{ do not}|Y = 2) \\ = \ & P(\text{persons 3 to } n \text{ do not}|\text{person 2 does}, Y = 2) \\ & \qquad \times P(\text{person 2 does}|Y = 2) \\ = \ & P(E_{n-2})\frac{1}{n-1}. \end{aligned}$$

So we see that

$$P_n = \frac{n-1}{n}P(E_n|Y = 2) = \frac{n-1}{n}P_{n-1} + \frac{1}{n}P_{n-2}$$

Let's finish this off now by solving these equations.

We have

$$P_n = \frac{n-1}{n} P_{n-1} + \frac{1}{n} P_{n-2},$$

which is equivalent to

$$P_n - P_{n-1} = -\frac{1}{n}(P_{n-1} - P_{n-2}).$$

This gives us a direct recursion for $P_n - P_{n-1}$. If we keep following it down we get

$$P_n - P_{n-1} = (-1)^{n-2} \frac{1}{n} \times \frac{1}{n-1} \times \ldots \times \frac{1}{3}(P_2 - P_1),$$

but since $P_2 = 1/2$ and $P_1 = 0$, $P_2 - P_1 = 1/2$ and

$$\begin{aligned} P_n - P_{n-1} &= (-1)^{n-2} \frac{1}{n} \times \frac{1}{n-1} \times \ldots \times \frac{1}{3} \times \frac{1}{2} \\ &= (-1)^{n-2} \frac{1}{n!}. \end{aligned}$$

So starting with $P_2 = 1/2$, this gives

$$\begin{aligned} P_3 &= P_2 - \frac{1}{3!} = \frac{1}{2} - \frac{1}{3!} \\ P_4 &= P_3 + \frac{1}{4!} = \frac{1}{2} - \frac{1}{3!} + \frac{1}{4!}, \end{aligned}$$

and so on. In general we would have

$$P_n = \frac{1}{2} - \frac{1}{3!} + \ldots + (-1)^n \frac{1}{n!}.$$

For $r = 0$, please check that this is the same answer we got last week.

The case $r = 0$ was really the hard part of the calculation. For $1 \leq r \leq n-2$ (recall that the probability that all n persons pick up their own hat is $1/n!$ and the probability that exactly $n-1$ persons pick up their own hat is 0), we can use a straightforward counting argument to express the answer in terms of P_{n-r}, the probability that exactly $n-r$ persons do not pick up their own hat.

In fact, recall that

$$\begin{aligned} & P\,(\text{exactly } r \text{ persons pick up their own hat}) \\ &= \binom{n}{r} P(\text{persons } 1, \ldots, r \text{ do and persons } r+1 \ldots, n \text{ don't}), \end{aligned}$$

since we can select the particular set of r people who pick up their own hat in exactly $\binom{n}{r}$ ways and, for each set of persons, the probability that they pick up their own hats while the other persons do not is the same no matter what subset of people we pick. However,

$$\begin{aligned} & P\,(\text{persons } 1, \ldots, r \text{ do and persons } r+1, \ldots, n \text{ don't}) \\ &= P(\text{persons } r+1, \ldots, n \text{ don't} | \text{persons } 1, \ldots, r \text{ do}) \\ &\qquad \times P(\text{persons } 1, \ldots, r \text{ do}) \\ &= P_{n-r} \frac{(n-r)!}{n!}, \end{aligned}$$

and so

$$\begin{aligned} & P\,(\text{exactly } r \text{ persons pick up their own hat}) \\ &= \binom{n}{r} \frac{(n-r)!}{n!} P_{n-r} = \frac{1}{r!} P_{n-r} \\ &= \frac{1}{r!} \left(\frac{1}{2} - \frac{1}{3!} + \ldots + (-1)^{n-r} \frac{1}{(n-r)!} \right). \end{aligned}$$

Summary of Chapter 3:

In Chapter 3 we've seen several useful techniques for solving problems. Conditioning is a very important tool that we'll be using throughout the course. Using conditioning arguments can decompose a probability or an expectation into simpler conditional expectations or probabilities, but the problem can still be difficult, and we still often need to be able to simplify and evaluate fairly complex events in a direct way, for example using counting or symmetry arguments.

Getting good at solving problems using conditioning takes practice. It's not a matter of just knowing the Law of Total Expectation. It's like saying that because I read a book on Visual Programming I now expect that I can say I'm a programmer. The best way to get the process of solving a problem into your minds is to do problems. I strongly recommend looking at problems in Chapter 3 in addition to the homework problems. At least look at some of them and try to work out how you would approach the problem in your head. I'll be glad to answer any questions you may have that arises out of this process.

Having stressed the importance of reading examples and doing problems, we should note that this course is also about learning some general theory for stochastic processes. Up to now we've been mostly looking at examples of applying theory that we either already knew or took a very short time to state (such as the Law of Total Expectation). We'll continue to look at plenty of examples, but it's time to consider some general theory now as we start looking at Markov Chains.