Lecture Notes

1 The Expected Value of a Product

This lecture continues the discussion of expected value.

In the last lecture, we looked at the expected value of a *sum* of random variables. The main result, called linearity of expectation, says that the expected value of a sum is the sum of the expected values; that is, $\mathrm{Ex}(R_1 + R_2) = \mathrm{Ex}(R_1) + \mathrm{Ex}(R_2)$. The wonderful thing about linearity of expectation is that R_1 and R_2 need not be independent.

In this lecture, we will look at the expected value of a *product* of random variables. If R_1 and R_2 are independent, then an analogous result holds. Namely, the expected value of a product is the product of the expected values.

Theorem 1.1 For any two *independent* random variables R_1 and R_2

$$\mathrm{Ex}(R_1 \cdot R_2) = \mathrm{Ex}(R_1) \cdot \mathrm{Ex}(R_2)$$

Proof. We will transform the left side into the right side. We first use the definition of expected value.

$$Ex(R_1 \cdot R_2) = \sum^{x \in \mathrm{Range}(R_1 \cdot R_2)} x \cdot \Pr(R_1 \cdot R_2 = x)$$

Next, we break the summation over the product $R_1 \cdot R_2$ into separate sums over R_1 and R_2.

$$= \sum^{x_2 \in \mathrm{Range}(R_2)} \sum^{x_1 \in \mathrm{Range}(R_1)} x_1 x_2 \cdot \Pr((R_1 = x_1) \cap (R_2 = x_2))$$

We now use the fact that R_1 and R_2 are independent. The probability of the intersection of the events $R_1 = x_1$ and $R_2 = x_2$ is the product of the probabilities of the two events.

$$= \sum^{x_2 \in \mathrm{Range}(R_2)} \sum^{x_1 \in \mathrm{Range}(R_1)} x_1 x_2 \cdot \Pr(R_1 = x_1) \cdot \Pr(R_2 = x_2)$$

In the remaining four steps, we alternate pulling a term out of a summation and applying the definition of expected value.

$$
\begin{aligned}
&= \sum^{x_2 \in \text{Range}(R_2)} x_2 \cdot \Pr(R_2 = x_2) \sum^{x_1 \in \text{Range}(R_1)} x_1 \cdot \Pr(R_1 = x_1) \\
&= \sum^{x_2 \in \text{Range}(R_2)} x_2 \cdot \Pr(R_2 = x_2) \cdot \text{Ex}(R_1) \\
&= \text{Ex}(R_1) \sum^{x_2 \in \text{Range}(R_2)} x_2 \cdot \Pr(R_2 = x_2) \\
&= \text{Ex}(R_1) \cdot \text{Ex}(R_2)
\end{aligned}
$$

■

1.1 The Product of Two Independent 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:

$$
\begin{aligned}
\text{Ex}(R_1 \cdot R_2) &= \text{Ex}(R_1) \cdot \text{Ex}(R_2) \\
&= 3\frac{1}{2} \cdot 3\frac{1}{2} \\
&= 12\frac{1}{4}
\end{aligned}
$$

The first step uses Theorem 1.1 and the fact that the dice are independent. In the second step, we use the result from last lecture that the expected value of one die is $3\frac{1}{2}$. The last step is simplification. The expected value of the product is $12\frac{1}{4}$.

1.2 The Product of Two Dependent Dice

Suppose that the two dice are not independent; in fact, assume that the second die is always the same as the first. Now what is the expected value of the product?

Because the dice are not independent, we can no longer apply Theorem 1.1. However, is the expected value of the product really changed by this loss of independence? As before, 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 1.1 as follows:

$$\begin{aligned}
\mathrm{Ex}(R_1 \cdot R_2) &= \mathrm{Ex}(R_1^2) \\
&= \sum_{i=1}^{6} i^2 \cdot \Pr(R_1 = i) \\
&= \frac{1^2}{6} + \frac{2^2}{6} + \frac{3^2}{6} + \frac{4^2}{6} + \frac{5^2}{6} + \frac{6^2}{6} \\
&= 15\frac{1}{6}
\end{aligned}$$

The first step uses the fact that the outcome of the second die is always the same as the first. In the second step, we expand $\mathrm{Ex}(R_1^2)$ according to the definition of expected value. In the third step, we use the fact that the probability of each number coming up is $\frac{1}{6}$, since the dice are fair. The last step is simplification. The expected value of the product has changed to $15\frac{1}{6}$.

The preceding two examples show that for *dependent* variables, the expected value of a product need not be the product of the expected values; that is, $\mathrm{Ex}(R_1 \cdot R_2) \neq \mathrm{Ex}(R_1) \cdot \mathrm{Ex}(R_2)$. The general rules for expected values of sums and products are summarized below.

For random variables R_1 and R_2:

- Regardless of whether R_1 and R_2 are independent, $\mathrm{Ex}(R_1 + R_2) = \mathrm{Ex}(R_1) + \mathrm{Ex}(R_2)$.
- If R_1 and R_2 are independent, then $\mathrm{Ex}(R_1 \cdot R_2) = \mathrm{Ex}(R_1) \cdot \mathrm{Ex}(R_2)$.

1.3 Corollaries

Theorem 1.1 extends to a collection of mutually independent variables.

Corollary 1.2 If random variables $R_1, R_2, \ldots, R_n$ are mutually independent, then

$$\mathrm{Ex}(R_1 \cdot R_2 \cdots R_n) = \mathrm{Ex}(R_1) \cdot \mathrm{Ex}(R_2) \cdots \mathrm{Ex}(R_n)$$

The proof of the corollary is an induction argument using Theorem 1.1 and the definition of mutual independence. Details are omitted here.

Adjusting a random variable by an additive or multiplicative constant adjusts the expected value in the same way.

Corollary 1.3 If R is a random variable and a and b are constants, then

$$\mathrm{Ex}(aR + b) = a\mathrm{Ex}(R) + b$$

This corollary is proved by regarding a and b as random variables that take on a particular value with probability 1. We can prove that a is always independent of R. The result then follows from linearity of expectation and Theorem 1.1.

We now know the expected value of a sum or product of random variables. Unfortunately, the expected value of a reciprocal is not so easy to characterize. Here is a flawed attempt.

False Corollary 1.4 If R is a random variable, then

$$\mathrm{Ex}\left(\frac{1}{R}\right) = \frac{1}{\mathrm{Ex}(R)}$$

As a counterexample, suppose the random variable R is 1 with probability $\frac{1}{2}$ and is 2 with probability $\frac{1}{2}$. Then we have:

$$\begin{aligned}
\frac{1}{\mathrm{Ex}(R)} &= \frac{1}{1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{2}} = \frac{2}{3} \\
\mathrm{Ex}(\frac{1}{R}) &= \frac{1}{1} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}
\end{aligned}$$

The two quantities are not equal, so the corollary is false. Here is another false corollary, which we will actually "prove".

False Corollary 1.5 Let Z and R be random variables, and let c be a real number.

$$\mathrm{Ex}\left(\frac{Z}{R}\right) = c \quad \Rightarrow \quad \mathrm{Ex}(Z) = c \cdot \mathrm{Ex}(R)$$

Proof.

$$\begin{aligned}
\mathrm{Ex}\left(\frac{Z}{R}\right) &= c \\
\Rightarrow \quad \mathrm{Ex}\left(\frac{Z}{R}\right) \cdot \mathrm{Ex}(R) &= c \cdot \mathrm{Ex}(R) \\
\Rightarrow \quad \mathrm{Ex}\left(\frac{Z}{R} \cdot R\right) &= c \cdot \mathrm{Ex}(R) \\
\Rightarrow \quad \mathrm{Ex}(Z) &= c \cdot \mathrm{Ex}(R)
\end{aligned}$$

In the first step, we multiply both sides by $\mathrm{Ex}(R)$. In the second step, we apply Theorem 1.1, and the last step is simplification. ■

This proof is bogus! We can not apply Theorem 1.1 in the second step, because $\frac{Z}{R}$ and R are not necessarily independent.

2 A RISC Paradox

Unfortunately, the fact that Corollary 1.5 is false, does not mean that it is never used! 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 made a table of program lengths for some benchmark problems:

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 first two columns are program lengths for each type of processor. The third 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".

This reasoning is wrong! One problem is that the total amount of RISC code for all benchmarks is 3220 bytes versus 2000 bytes for the CISC processor. In other words, the RISC code is much *longer* overall. One could argue, however, that this is a bad calculation, since one large, atypical benchmark (e.g. Rec 2-sort) could skew the whole result.

However, there is a second problem that can not be waved away. Suppose we 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?

2.1 A Probabilistic Interpretation

To shed some light on this paradox, 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 RISC program, and let the random variable C be the length of the CISC program. We would like to compare the average length of a RISC program, Ex(R), to the average length of a CISC program, Ex(C).

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. We can now compute Ex(R) and Ex(C) as follows:

$$\begin{aligned}
\text{Ex}(R) &= \sum^{i \in \text{Range(R)}} i \cdot \Pr(R = i) \\
&= \frac{150}{4} + \frac{120}{4} + \frac{150}{4} + \frac{2800}{4} \\
&= 805
\end{aligned}$$

$$\begin{aligned}
\text{Ex}(C) &= \sum^{i \in \text{Range(C)}} i \cdot \Pr(C = i) \\
&= \frac{120}{4} + \frac{180}{4} + \frac{300}{4} + \frac{1400}{4} \\
&= 500
\end{aligned}$$

Since $\text{Ex}(R)/\text{Ex}(C) = 1.61$, 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. In fact, both of the earlier results are wrong, and this one is correct.

In terms of our probability model, the paper computes C/R for each sample point, and then averages to obtain $\text{Ex}(C/R) = 1.2$. This much is correct. However, the authors then conclude that "CISC programs are 20% longer on average"; that is, they conclude that $\text{Ex}(C) = 1.2\ \text{Ex}(R)$. The authors have applied Corollary 1.5, which we know to be false!

Similarly, our calculation correctly showed that $\text{Ex}(R/C) = 1.1$. However, we then applied the false corollary to conclude that $\text{Ex}(R) = 1.1\ \text{Ex}(C)$; that is, a RISC program is 10% longer than a CISC program on average. Since both arguments are false, it should come as no surprise that they give contradictory conclusions!

2.2 A Simpler Example

The source of the problem is more clear 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. Do not take an average of ratios blindly!

3 Wald's Theorem

Let's try to compute an expected value in a more complicated experiment. Suppose that we flip 3 fair coins. For each head that comes up, we roll a fair die. For example, if we flipped HTH, then we would roll two dice. Assume that the coin tosses and dice rolls are all mutually independent. What is the expected value of the sum of the dice rolls?

This experiment is more complicated, because there is a random number of random variables; the outcome of the coin tosses determines the number of dice rolls. Still, we could solve this problem by drawing a tree diagram showing all possible outcomes, computing the probability of each outcome, and then finding the desired expected value. However, we can also bash this problem with a big hammer called Wald's Theorem. The idea of Wald's Theorem is simple, but the statement is swamped by technical restrictions.

Theorem 3.1 (Wald's Theorem) Let Q be a random variable that takes on only non-negative, integer values such that $\text{Ex}(Q)$ is finite. Let $R, R_1, R_2, \ldots$ be a sequence of random variables, all with the same distribution and with $\text{Ex}(|R|)$ finite. Let T be the sum $R_1 + R_2 + \ldots + R_Q$. Suppose that every variable R_k is independent of the event that it is included in the sum; that is, for all $k \geq 1$, R_k is independent of an indicator variable for the event that $Q \geq k$. Then:

$$\text{Ex}(T) = \text{Ex}(Q) \cdot \text{Ex}(R_1)$$

Intuitively, Wald's Theorem says that the sum of a random number of random variables is equal to the expected number of terms times the expected value of each term. To correctly apply the theorem, however, we must always check three conditions:

1. The random variable Q takes on only non-negative, integer values, and $\text{Ex}(Q)$ is finite.
2. All random variables R and R_k have the same distribution, and $\text{Ex}(|R|)$ is finite.
3. Every variable R_k is independent of the event that it is included in the sum; that is, for all $k \geq 1$, R_k is independent of an indicator variable for the event that $Q \geq k$.

While a bit tricky to follow, the third condition eliminates some nasty cases that would otherwise break the theorem. For example, we could have a collection of random variables R_k, each of which is zero whenever it is included in the sum; that is, $R_k = 0$ precisely when $Q \geq k$. Then the left side of the equation in the theorem would be zero and the right side would be non-zero.

3.1 Solution to the Coins-and-Dice Problem

Before proving Wald's Theorem, let's use it to solve the coins-and-dice problem. In this case, Q is the number of heads we flip, $R_1, R_2, \ldots$ are the outcomes of die rolls, and T is the sum of the first Q rolls. Our goal is to compute the expected value of the sum of the dice rolls, $\text{Ex}(T)$.

To apply Wald's Theorem, we must check the three conditions listed above. Since we are tossing 3 coins, Q takes on only non-negative integer values, and $\text{Ex}(Q) = \frac{3}{2}$ is finite. All of the dice are fair, so all random variables R_k have the same distribution, and $\text{Ex}(|R|) = 3\frac{1}{2}$ is finite.

Finally, every variable R_k (a function of the dice) is independent of the event that it is included in the sum (a function of the coins), since we assumed that all the dice rolls and coin tosses were mutually independent. Since all three conditions are satisfied, we can apply Wald's theorem as follows:

$$\begin{aligned}
\text{Ex}(T) &= \text{Ex}(Q) \cdot \text{Ex}(R_1) \\
&= \frac{3}{2} \cdot 3\frac{1}{2} \\
&= \frac{21}{4}
\end{aligned}$$

The first step uses Wald's Theorem. In the second step, we substitute the expected number of heads ($\frac{3}{2}$) and the expected value of a die roll ($3\frac{1}{2}$). The final step is simplification. We conclude that the expected value for the sum of the dice is $\frac{21}{4}$.

3.2 Proof of Wald's Theorem

We will prove a slightly weaker form of Wald's Theorem than the one stated above. This conveys the intuition behind the result without involving complicated math. In particular, we will assume that *Q is bounded above by some constant c*; that is, we always sum at most c random variables. For example, in the coins-and-dice problem we always sum at most $c = 3$ random variables, since we can flip at most three heads on three coins. Later in this lecture, we will see a problem requiring the full version of Wald's Theorem.

Proof. Let I_k be an indicator variable for the event that $Q \geq k$. That is, define I_k as follows:

$$I_k = \begin{cases} 1 & \text{if } Q \geq k \\ 0 & \text{if } Q < k \end{cases}$$

Intuitively, I_k indicates whether R_k is included the sum $R_1 + R_2 + \ldots + R_Q$. Using these indicator variables, we can rewrite T as shown below:

$$\begin{aligned}
T &= R_1 + R_2 + \ldots + R_Q \\
&= \sum_{k=1}^{\infty} R_k I_k
\end{aligned}$$

The effect of the indicator variables is to cancel out all but the first Q terms in the infinite sum. We can now compute the expected value of T as follows:

$$\text{Ex}(T) = \text{Ex}\left(\sum_{k=1}^{\infty} R_k I_k\right)$$

On the right, we have the expectation of an infinite sum. We would like to apply linearity of expectation, but we only know that this rule holds for *finite* sums. This is the motivation for our assumption that Q is bounded above by some constant c. This means that $I_k = 0$ for all $k > c$, and so all but the first c terms of the infinite sum are zero. Now we can rewrite the infinite sum as a finite sum and apply linearity of expectation.

$$\begin{aligned} &= \text{Ex}\left(\sum_{k=1}^{c} R_k I_k\right) \\ &= \sum_{k=1}^{c} \text{Ex}\left(R_k I_k\right) \end{aligned}$$

The third condition of Wald's Theorem is precisely that R_k and I_k are independent. Therefore, the expectation of the product is the product of the expectations:

$$= \sum_{k=1}^{c} \text{Ex}(R_k)\text{Ex}(I_k)$$

The expected value of R_k is equal to the expected value of R by the second condition of Wald's Theorem. The expected value of an indicator variable like I_k is the probability that it is 1, which occurs when $Q \geq k$.

$$= \sum_{k=1}^{c} \text{Ex}(R)\,\Pr(Q \geq k)$$

Next, we pull the term $\text{Ex}(R)$ outside the summation. The remaining sum is equal to $\text{Ex}(Q)$, since by the first condition of Wald's Theorem, Q is a non-negative, integer-valued random variable and since $\Pr(Q > c) = 0$ by assumption.

$$\begin{aligned} &= \text{Ex}(R) \sum_{k=1}^{c} \Pr(Q \geq k) \\ &= \text{Ex}(R) \cdot \text{Ex}(Q) \end{aligned}$$

■

4 Building a System

Wald's Theorem turns out to be useful in analyzing algorithms and systems. The following example was taught incorrectly in 6.033 for several years. The problem– and erroneous solution– originated in a paper by Herbert Simon, who won the Nobel Prize in economics.

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?

4.1 The Sample Space

We can regard the sample points in this experiment as infinite strings of S's and F's. An S in the i-th position indicates that a component is successfully added in the i-th step. An F in the i-th position indicates that the system falls apart in the i-th 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 $SSFSFF\underline{SSS}FSF\ldots$, the system is completed successfully after 9 steps, since after 9 steps we have encountered a string of three consecutive S's.

4.2 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 k-th try, and 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 $SSFSFFSSSFSF\ldots$into tries as shown below:

$$\underbrace{S\ \ S\ \ F}_{\substack{R_1 = 3 \\ \text{failure}}}\ \underbrace{S\ \ F}_{\substack{R_2 = 2 \\ \text{failure}}}\ \underbrace{F}_{\substack{R_3 = 1 \\ \text{failure}}}\ \underbrace{S\ \ S\ \ S}_{\substack{R_4 = 3 \\ \text{success!}}}\ \underbrace{F}_{\substack{R_5 = 1 \\ \text{failure}}}\ \underbrace{S\ \ F}_{\substack{R_6 = 2 \\ \text{failure}}}\ \cdots$$

(Note that we regard an outcome as an infinite string of S's and F's, and so an outcome formally consists of an infinite number of tries. However, the remainder of the string after the first successful try is not relevant.)

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:

$$T = \sum_{k=1}^{Q} R_k = R_1 + R_2 + R_3 + R_4 = 3 + 2 + 1 + 3 = 9$$

4.3 Applying Wald's Theorem

Our goal is to determine $\text{Ex}(T)$, the expected number of steps needed to complete the system. The first step is to apply Wald's Theorem:

$$\begin{aligned} \text{Ex}(T) &= \text{Ex}\left(\sum_{k=1}^{Q} R_k\right) \\ &= \text{Ex}(Q) \cdot \text{Ex}(R) \end{aligned}$$

Here R is a random variable indicating the number of steps in a try.

To verify the second step, we must check the three conditions in Wald's Theorem. First, the number of tries, Q, is always a non-negative integer. We will show that $\text{Ex}(Q)$ is finite later. Second, all tries are symmetric, so all random variables R_k have the same distribution. Furthermore, $\text{Ex}(|R|)$ is finite because every try lasts at most n steps. Third, the number of steps in try k is independent of the event that k or more tries are required; that is, R_k is independent of an indicator variable for the event that $Q \geq k$.

(Note that we are using the general form of Wald's Theorem, not the restricted form that we proved. The number of tries Q is not bounded above by any constant c.)

To solve the problem, we must still find $\text{Ex}(Q)$ and $\text{Ex}(R_1)$.

4.4 Computing the Expected Number of Tries

Let's compute $\text{Ex}(Q)$, 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 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 encountered this question in another guise during the last lecture. 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 $\frac{1}{q}$. 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 $\frac{1}{(1-p)^n}$. Therefore, we have:

$$\mathrm{Ex}(Q) = \frac{1}{(1-p)^n}$$

This also shows that Q is finite, provided $p \neq 1$.

4.5 Computing Expected Length of a Try

Now let's compute $\mathrm{Ex}(R)$, the expected number of steps in a try. Since the number of steps in a try is always a non-negative integer, we can write:

$$\begin{aligned} \mathrm{Ex}(R) &= \sum_{i=0}^{\infty} \Pr(R > i) \\ &= \sum_{i=0}^{n-1} \Pr(R > i) \end{aligned}$$

The second equality holds because a try never lasts for more that n steps; if the system does not collapse within n steps, then the system is completed, and the try is done anyway. Therefore, $\Pr(R > n) = 0$, and so only the first n terms of the infinite sum can be non-zero.

Now we must evaluate $\Pr(R > i)$, 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, $\Pr(R > i) = (1-p)^i$. Substituting this into the equation above and summing the resulting geometric series gives the expected number of steps in a try:

$$\begin{aligned} &= \sum_{i=0}^{n-1} (1-p)^i \\ &= \frac{1-(1-p)^n}{1-(1-p)} \\ &= \frac{1-(1-p)^n}{p} \end{aligned}$$

Now we can compute the expected number of steps needed to complete the system.

$$
\begin{aligned}
\mathrm{Ex}(T) &= \mathrm{Ex}(Q) \cdot \mathrm{Ex}(R) \\
&= \frac{1}{(1-p)^n} \cdot \frac{1-(1-p)^n}{p} \\
&= \frac{1-(1-p)^n}{p(1-p)^n} \\
&= \frac{1}{p(1-p)^n} - \frac{1}{p}
\end{aligned}
$$

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 completed a system with $n = 10$ components is about 10.57. For $n = 100$ components, the number of steps is about 173.20. 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!

4.6 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 time 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 time 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 $111 \cdot 10.57 \ldots \approx 1173$ steps. This compares very favorably with the 2.3 million steps required in the direct method!