Appendix V. Randomness and Probability

One of the excellent advantages of computers and calculators is the opportunity they provide to model real world activities and to draw conclusions from the results of that modeling. An important tool for this kind of application is the calculator's random number generator. Although the examples of this appendix are largely drawn from games of chance, the real world applications of random numbers are extensive and important.

Here is an example of how this powerful tool works. Suppose we want to roll a pair of dice over and over but we are either too lazy to do so or we don't happen to have a pair of dice handy. We can do this with the following brief program:

```
PROGRAM DICE
: Lbl 1
: int(6*rand)+1 STO> R
: int(6*rand)+1 STO> S
: R+S STO> T
: Disp T
: Pause
: Goto 1
```

I suggest that you enter that program now and run it, copying down some of your results. You will find the command `rand` at `MATH Prob`.

When you run that program, you will indeed display a series of "dice toss" totals one at a time, something like 5, 7, 3, 11, 8, 8, 2,...

Is there an android in your calculator rolling dice? Of course not.

Here is what is going on. Your calculator includes what is called a random number generator. Every time you enter the command `rand`, the calculator returns a new ten digit decimal value I will designate as r in the range 0 < r < 1. For example, I called up that command and pressed ENTER. My calculator responded: .9990836386. Then when I pressed ENTER again and again and again, it gave me .3503353354, .0669258334, and .5770724903. Your calculator has an internal program that provides numbers like this as nearly randomly as the engineers can make them.

Now let's see what is happening in those three key program lines:

```
: int(6*rand)+1 STO> R
: int(6*rand)+1 STO> S
: R+S STO> T
```

The instruction `int(6*rand)` takes one of those random numbers, multiplies it by 6 and then takes the integer value of the result. Recalling that `rand` gave us a number in the range 0 < r < 1, when we multiply it by 6 we obtain 0 < 6r < 6. In other words, when we multiply by 6 we find a value somewhere in that range 0 to 6. Take the integer value of that number (remember that is like rounding down) and we have 0, 1, 2, 3, 4 or 5. Now, to complete `int(6*rand)+1`, 1 is added to this number and we have one of the values: 1, 2, 3, 4, 5 or 6. Here are some examples of this sequence using the random numbers my calculator gave me:
Inside Your Calculator

<table>
<thead>
<tr>
<th>rand</th>
<th>6*rand</th>
<th>int(6*rand)</th>
<th>int(6*rand)+1</th>
</tr>
</thead>
<tbody>
<tr>
<td>.9990836386</td>
<td>5.994501832</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>.3503353354</td>
<td>2.102012012</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>.0669258334</td>
<td>.4015550004</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>.5770724903</td>
<td>3.462434942</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

When the first of those three program lines is processed, a value of a single die is stored in R. In the second another value is stored in S and in the third those two are added to give T. Thus the first time the steps in my example are run, T would become 9, adding R=6 and S=3; in the second pass T would become 5.

You might be tempted to combine those three steps into one because the first two look so similar. Be sure you see why this line could NOT replace them:

\[2 \times (\text{int}(6 \times \text{rand}) + 1) \text{ STO> T}\]

The problem with that instruction is that, since only one random value is obtained, it would make both dice give the same number and thus would produce only even sums, in the case of our example values: 12, 6, 2 and 8.

Because we seek random integers so often, there is a shortcut for \(\text{int}(6 \times \text{rand}) + 1\). It is \textbf{RandInt(1,6)}, also found at \textbf{MATH Prob}. What are entered between the parentheses are the lower and upper bounds of the numbers which have an equal chance of appearing.

Be sure that you remember the "equal chance" in that statement. You might be tempted to think of modeling the roll of two dice by using the instruction line:

\[\text{RandInt}(2,12) \text{ STO> T}\]

That command would indeed give you integers from 2 to 12, each of them occurring approximately equally. The totals produced by rolling two dice are, however, not all equal as shown by the following summary:

<table>
<thead>
<tr>
<th>Dice Total</th>
<th>Ways of Achieving This</th>
<th>Number of Ways</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1 1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1 2, 2 1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>1 3, 2 2, 3 1</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>1 4, 2 3, 3 2, 4 1</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>1 5, 2 4, 3 3, 4 2, 5 1</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>1 6, 2 5, 3 4, 4 3, 5 2, 6 1</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>2 6, 3 5, 4 4, 5 3, 6 2</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>3 6, 4 5, 5 4, 6 3</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>4 6, 5 5, 6 4</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>5 6, 6 5</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>6 6</td>
<td>1</td>
</tr>
</tbody>
</table>
Since there are 36 possible ordered pairs of dice, you roll, for example, 2 only 1/36 of the time and 7, 6/36 or 1/6 of the time. Clearly the totals do not appear equally often. To simulate throwing two dice you would need to use the lines given in our example which could have been combined into the single program line

\[
\text{RandInt}(1,6) + \text{RandInt}(1,6) \; \text{STO} \; T
\]

for each throw.

**Probability**

Probability and statistics are very serious and important academic studies; universities offer doctoral degrees in this field. It is not, therefore, my intent to offer more than a slight introduction (or more likely today, reintroduction, since this subject is regularly taught in schools) to this field. I will offer here then only a few basic tools of this subject.

Most of us today have a good idea of what we mean when we write that a given event has probability \( p = a/b \); we mean that \( a \) times out of \( b \), that event "should" happen. Of course, that word "should" needs interpretation. We will be talking about probabilities, which we can often justify through the mathematics underlying the activity. For example, when we throw a die, its symmetries indicate that each face will generally occur equally as often; thus we have for the throw of a single die, \( p = 1/6 \). There are many situations when those probabilities are less sure. When we say that the probability of it raining tomorrow is \( p = 3/5 \), we are far less sure. Our examples in this appendix will fit the more certain category.

The first and most important thing to consider when dealing with probability is that, insofar as possible, you want to work with basic events that occur with equal chance. However, just as we know that a single fair die offers equal chance of obtaining 1, 2, 3, 4, 5 or 6, as we have just seen, rolling two dice does not produce equal chance of rolling the various sums.

The second thing to consider when dealing with probability problems involving more than one event is independence. If one event affects the next, you will need to take that into account. For example, suppose you have a standard deck of 52 cards: 13 spades, 13 hearts, 13 diamonds and 13 clubs. If you drew a card from such a deck, you should have a 13/52 or 1/4 chance of drawing a spade. But what is the chance of drawing two spades? Unless you replace the first card you drew, the second draw is dependent on the first. If your first card was a spade, the chance now is 12/51; if not, it is 13/51.

You should keep those two considerations in mind. They play an important role in what follows.

But here we need to back up a bit. Where did that 13/52 come from when drawing a spade from a deck of cards in the first place? That was the fraction:

\[
\text{number of ways of drawing a spade} \over \text{total number of ways of drawing a card}
\]

In our example of the deck of cards, there were 13 spades so the numerator was 13, out of a deck of 52 cards, so the denominator was 52. Similarly, the chance of rolling a 5 with a single die is
Inside Your Calculator

1/6, one way in the numerator, out of 6 possibilities in the denominator. Remember once again that the numbers in that fraction are numbers of things that have equal chances.

Now we can ask a simplistic question, given a deck of cards and a single die, what is the chance of drawing a spade from the deck and then rolling a five with the die? It turns out that, when two events are independent, as these are, and you want both to happen, as you do these, you can multiply their probabilities. In this case we have (using the notation \( P_{\text{event}} \) to represent the probability of that event happening):

\[
P_{\text{spade}} \times P_{\text{five}} = \frac{1}{4} \times \frac{1}{6} = \frac{1}{24}
\]

If, on the other hand, you wanted to know the chance of drawing a spade or rolling that five, you would add the chances:

\[
P_{\text{spade}} + P_{\text{five}} = \frac{1}{4} + \frac{1}{6} = \frac{5}{12}
\]

A useful way of deciding whether to multiply or add, try to reword your situation so that the words "and then" or "or" occur. "And then" leads to multiplication, "or" to addition.

We can see this in our earlier example of the two dice. Suppose we concentrate on the chance of throwing a 10 with two dice. To achieve this total we could:

- throw a 4 and then a 6
- or
- throw a 5 and then a 5
- or
- throw a 6 and then a 4

Now we need only recall that the chance of throwing any individual die is 1/6. Translating this series of events line by line gives us:

\[
\frac{1}{6} \times \frac{1}{6} + \frac{1}{6} \times \frac{1}{6} + \frac{1}{6} \times \frac{1}{6} = \frac{3}{36} = \frac{1}{12}
\]

which sums to 3/36 or 1/12. Thus we see that we have one chance out of twelve to throw a 10 with two dice.

---

1 In many probability applications, estimates are used. You have seen this, for example, in weather projections. "There is a 30% chance of rain tomorrow," the weatherman says. What he means is that past experience indicates that the present conditions will lead to rain tomorrow on 3 days out of ten. We all know how often such reports and our subsequent experience fail to satisfy us with such predictions. Despite our dissatisfaction, however, such scientifically based projections serve us far better than such seasonal guides as The Farmer's Almanac, regardless of claims to the contrary.
Why Do We Need Random Numbers?

You might think that you could make up a series of numbers at random yourself. A good exercise is to try this. To test yourself to see if you pass a simple test of randomness, before you read ahead, write down 100 single digit numbers, 0 to 9. If you do this in an orderly way, say 10 numbers in each row, you will only need 10 rows.

Once you have finished, check your list. Did each digit appear about the same number of times, about 10 times for this list? (Note that you would have passed this test if you simply wrote the digits in order 10 times.) But now check to see how many pairs of digits you produced, like 5 5. You should have had about ten such pairs. And how many triplets, like 4 4 4. You should have had at least one triplet on your list. And one such list out of ten might have had quadruplets, like 1 1 1 1.

A good random number generator should give you not only the numbers occurring equally but also such strings occurring as expected by probability theory.

How often should these events occur? We can use those simple probability rules to predict how many. Assume any digit, d, appears. The chance of the next digit also being d is 1/10. And if that happens, that is you have the sequence d d, the chance of another d occurring is again 1/10. Since these events are independent, the chance of d d is 1/10 * 1/10 or 1/100. Continuing in the same way we have:

<table>
<thead>
<tr>
<th>digits alike</th>
<th>chance</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 (pair)</td>
<td>0.1</td>
</tr>
<tr>
<td>3 (triplet)</td>
<td>0.01</td>
</tr>
<tr>
<td>4 (quadruplet)</td>
<td>0.001</td>
</tr>
<tr>
<td>5 (quintuplet)</td>
<td>0.0001</td>
</tr>
<tr>
<td>6 (hextuplet)</td>
<td>0.00001</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Now, if we check our list of 100 numbers, we should expect the number of pairs to be about $0.1 \times 100 = 10$ and the number of triplets to be about $0.01 \times 100 = 1$. How closely does your list conform to these tests? You should not expect it to be exact, but most people come nowhere near these values. (They usually have far too few pairs and triplets.)

Here is a rather complicated program that gathers information about such digit distribution for the random numbers your calculator provides. You can use it to compare with your results, especially with regard to repeated digits and to see as well just how reasonable is your calculator's random number generator.

---

2 The word "odds" is often applied to such statements, especially when gambling is involved. If we have $p = m/n$, then the odds for this event to happen are $m:n-1$ and the odds against it happening are $n-1:m$. Thus the odds for throwing a 10 with two dice ($p = 1/12$) are $1:11$ and against $11:1$. If you were betting fairly that you could throw a 10, the person you are betting against should put up $11$ for each $1$ you bet.
PROGRAM: DIGCOUNT
:{1,3} STO> dim([B])
:For (R,1,3)
  :For (C,1,10)
    : 0 STO> [B](R,C)
  :End
:End
:Prompt N
:For(I,1,N)
  :randInt(0,9) STO> T
  :[B](3,T+1)+1 STO> [B](3,T+1)
  :For(J,6,2,-1)
    : [B](2,J-1) STO> [B](2,J)
  :End
  :T STO> [B](2,1)
  :If [B](2,1)=[B](2,2)
    :Then
      : [B](1,1)+1 STO> [B](1,1)
      :If [B](2,2)=[B](2,3)
        :Then
          : [B](1,2)+1 STO> [B](1,2)
          :If [B](2,3)=[B](2,4)
            :Then
              : [B](1,3)+1 STO> [B](1,3)
              :If [B](2,4)=[B](2,5)
                :Then
                  : [B](1,4)+1 STO> [B](1,4)
                  :If [B](2,5)=[B](2,6)
                    :[B](1,5)+1 STO> [B](1,5)
                    :End
                :End
            :End
        :End
    :End
  :End
:End

Instead of describing how that program works, I will describe how the output is placed in the matrix [B]. Here, for example, is the matrix I obtained when I ran the program with N = 100:

\[
\begin{bmatrix}
7 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
6 & 5 & 9 & 3 & 8 & 2 & 0 & 0 & 0 & 0 \\
5 & 6 & 14 & 10 & 11 & 8 & 10 & 16 & 11 & 9 \\
\end{bmatrix}
\]

(To view the matrix after you have run the program, key the MATRIX menu, choose Edit then [B].)

Here is what that matrix shows you:
1. The bottom line: These are the totals for each digit, 0-9. In this case there were 5 0s, 6 1s, 14 2s, etc.

2. The second line simply records the last six random digits in reverse order. The last random digit produced by the program was 6, the second last 5, etc. (In order to shorten program running time the last four digits of this line remained unused and are therefore still filled with zeros.) This line provides the data needed to find the values of the first row.

3. The top line records the number of pairs, triplets, quadruplets, quintuplets and hextuplets, the remainder of the line again not used.

Now that you know what the matrix tells us, consider the following results. (For clarity I have left the unused entries blank.) First, here is a matrix resulting from running the program with N = 10,000. The calculator took almost 20 minutes to complete this process:

\[
\begin{bmatrix}
1015 & 95 & 11 & 1 & 0 \\
3 & 4 & 2 & 8 & 2 & 6 \\
975 & 972 & 1039 & 1008 & 1038 & 1037 & 965 & 979 & 1018 & 969 \\
\end{bmatrix}
\]

Notice that the occurrence of the digits along the bottom line are each about 0.1*10,000 = 1000. Also notice that the number of pairs is about this same number, the number of triplets about 0.01*10,000 = 100, etc.

Here too are the results of running the program with N = 100,000, with running time over three hours:

\[
\begin{bmatrix}
10036 & 1024 & 95 & 9 & 1 \\
4 & 1 & 2 & 0 & 9 & 5 \\
10028 & 10180 & 9973 & 9999 & 9985 & 10064 & 9938 & 9919 & 10110 & 9794 \\
\end{bmatrix}
\]

If this program produced exactly the expected values, the top and bottom lines of the resulting matrix would have looked like this:

\[
\begin{bmatrix}
10000 & 1000 & 100 & 10 & 1 \\
\cdots \\
10000 & 10000 & 10000 & 10000 & 10000 & 10000 & 10000 & 10000 & 10000 & 10000 \\
\end{bmatrix}
\]

The fact that the results I obtained are close to these but not exactly the same should be reassuring. If they were exact, we should suspect that we had too much regularity.

What should also be apparent from the running time of this program is how willing your calculator is to work for you for long periods of time carrying out routine operations over and over again without complaint.
Inside Your Calculator

Just How Random are Random Numbers

This is not the setting in which to address the many problems, both mathematical and philosophical, associated with the concept of randomness. As just one indication of how complex is this topic, Donald Knuth devotes 160 pages of his monumental text series, *The Art of Computer Programming*,\(^3\) to the mathematician's search for ways of representing randomness. It is important to stress that your calculator does NOT provide truly random numbers. Because of this some mathematicians call the numbers it produces pseudo-random numbers. Somewhere in your calculator's hardware is a program that takes you from one of a series of numbers to the next and, if you are not careful, you will get the same series of results from one experiment to the next.

I have not tested this but I suspect that two brand new calculators of the same type would generate the same list of random numbers when first used. Your calculator does not do this because, each time you use it, it continues the series of numbers that it produced earlier.

The Monte Carlo Method\(^4\)

Mathematicians use Monte Carlo simulations to solve a wide range of real world problems. These simulations employ random numbers as their basic data source. Knuth suggests the range of these uses as from "the study of nuclear physics (where particles are subject to random collisions) to system engineering (where people come into, say, a bank at random intervals)." Other areas he considers are sampling, numerical analysis, checking computer programs for errors and executive decision making,\(^5\) as well as recreational activities.

An interesting example of so-called "recreational activities" is the time-honored slot machine. Formerly mechanical, individual machines now are essentially self-contained computers which can be programmed to provide a given income for the "house", the machine owner. In casinos these machines may also be linked to larger computers. This is today a billion dollar industry.

Exercises 22, 23 and 24 of the set of Additional Programming Exercises will give you an opportunity to use random numbers to explore these ideas.

---

\(^3\) Volume 2, Chapter 3. At the outset he quotes John von Neuman's ironic comment, "Anyone who considers arithmetical methods of producing random numbers is, of course, in a state of sin."

\(^4\) This name was chosen by the famous Los Alamos mathematician Stanislaw Ulam in recognition of his uncle, who regularly borrowed money from the family to gamble at the Monte Carlo gaming tables.

\(^5\) Knuth notes in this regard, "It is rumored that some college professors assign grades on such a basis."