SIMULATION

Statisticians use simulations built from random variables to model various situations. In an educational context, we can use it to develop a deeper understanding of the way that data collection can vary across samples and get a better sense of statistical distributions. This handout will examine statistical simulations using Excel and Minitab. We will use Excel primarily to see behind the scenes for how the simulations work, and then we'll use Minitab to help us automate the process.

Excel has two kinds of random numbers that we can rely on for simulations. The RANDBETWEEN function gives us uniformly distributed integer values. The RAND() function gives a uniformly continuous random number between 0 and 1. We can use the RANDBETWEEN function when we can express the probability as a ratio of integers. The RAND() function can be used regardless of the value of the probability.

	A	B	C	D
1	Toss 1	Toss 2	Sum	
2	=RANDBETWEEN(1,6)	=RANDBETWEEN(1,6)	=SUM(A2:B2)	
3				
4				

Suppose that we want to simulate the toss of two dice, and then calculate the sum. The RANDBETWEEN function takes two values: the smallest possible integer value the random variable can take, and the larges possible integer value the random variable can take. So, for a standard die, the values are 1 and 6. Since we are tossing two dice, enter the same formula in the second column, and then add the values.

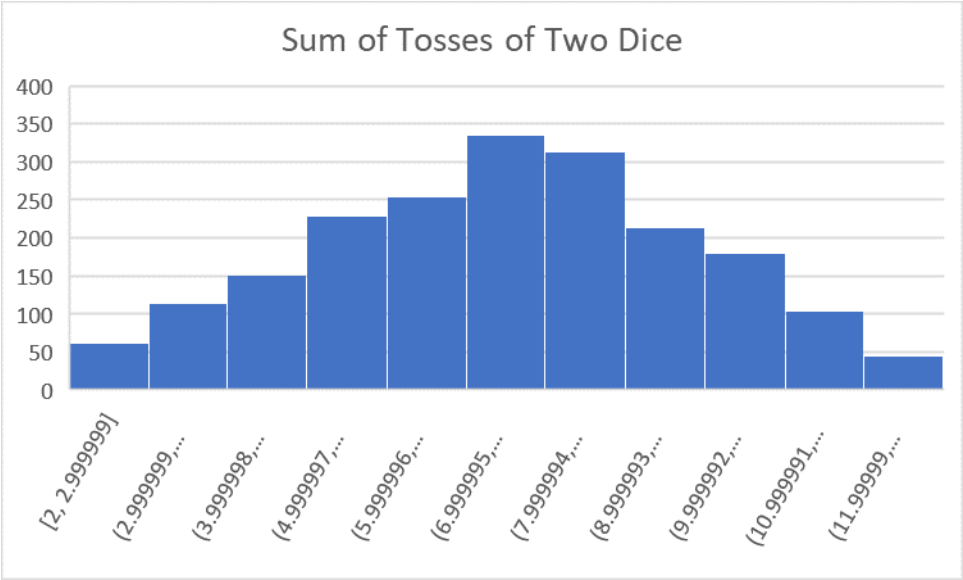
To simulate more than one toss, copy the formulas down the columns. Let's look at 10 simulated rolls of both dice.

	A	B	C	D
1	Toss 1	Toss 2	Sum	
2	2	4	6	
3	4	6	10	
4	1	5	6	
5	2	5	7	
6	3	2	5	
7	3	6	9	
8	4	3	7	
9	5	6	11	
10	4	2	6	
11	1	1	2	

The values you obtain will be different than mine because the outcomes are random. (A word of caution, every time a new value is calculated, all the random variables will recalculate. This is normal in Excel.)

Now, 10 simulations isn't that many, and we could do that many without too much trouble without technology as long as we had two dice and recorded the results of 10 tosses. But what if we wanted to look at 1000 or 10,000 tosses? Technology is a lot faster than people, and Excel (post 2015 or so) can handle about a million lines in a spreadsheet.

The histogram below is constructed from 2000 tosses.



We can also run binomial simulations to determine the number of successful outcomes given n trials and p probability of success.

Since I want to vary the probability value, I'll use the RAND() function here. As noted above, this function produces a random value between 0 and 1, the same values as probabilities can take on. If we set p, the probability of success on each trial, to a given value, we will evaluate the random number generated to see if it is less than the given value of p, or more. If less, we'll count that as success, and if more, we'll count that as failure.

	A	B	C
1	Attempts/Trials	1	2
2	1	=IF(RAND()<0.5,1,0)	
3	2		
4	3		

We use the IF function here to determine if the random number meets the conditions. The syntax requires a condition, the value or expression to be displayed if the condition is satisfied, and then the value to be displayed if the condition is not satisfied. The successes here will be listed as 1, and failures as 0. Here, we are treating the probability as $p = 0.5$, similar to a fair coin toss.

If we copy the formula down the first column, we can simulate 20 coin tosses to determine the number of heads of a coin we might obtain.

A23

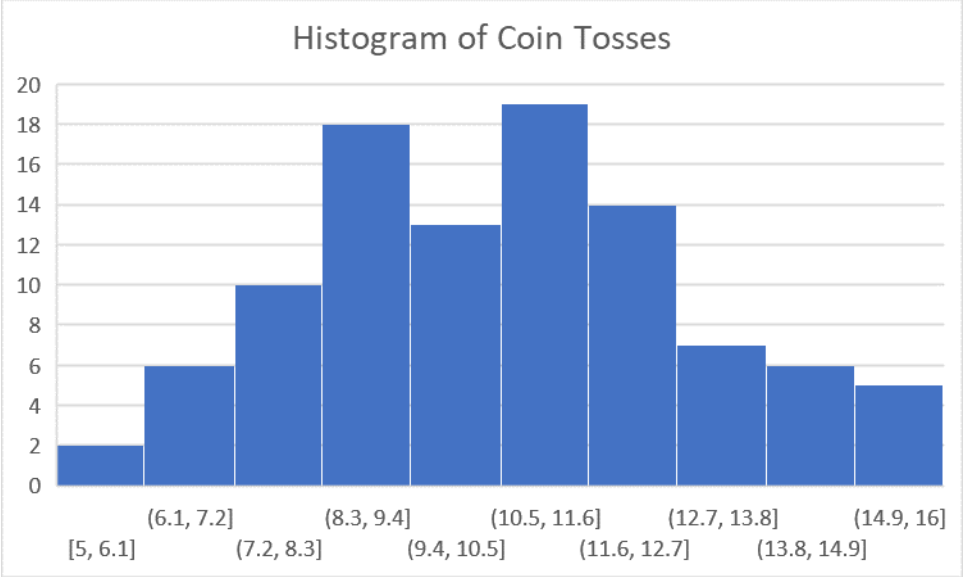
	A	B	C
1	Attempts/Trials	1	2
2	1	0	
3	2	1	
4	3	0	
5	4	0	
6	5	1	
7	6	0	
8	7	0	
9	8	1	
10	9	1	
11	10	1	
12	11	0	
13	12	0	
14	13	0	
15	14	1	
16	15	1	
17	16	1	
18	17	0	
19	18	0	
20	19	0	
21	20	0	
22	Sum	8	
23			

At the bottom of the column, I've added up the values above to count the number of successes (don't add the header that is just counting the number of 20 tosses).

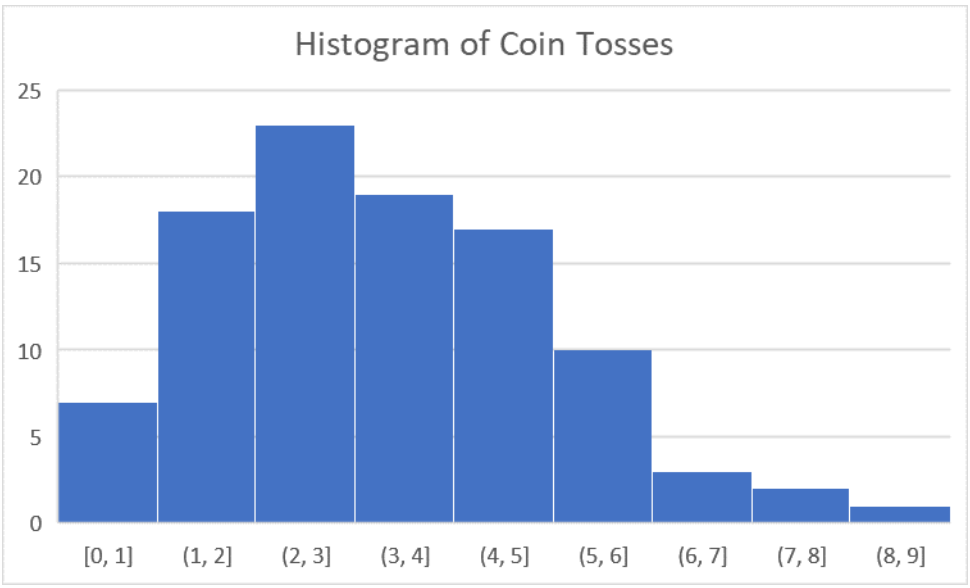
This is just one set of 20 tosses. What happens if we ran more trials? We can copy the formulas into other columns to see what happens when we make 20 tosses over and over again.

	A	B	C	D	E	F	G	H	I	J	K
1	Attempts/Trials	1	2	3	4	5	6	7	8	9	10
2	1	1	0	1	1	1	1	1	1	1	0
3	2	1	1	1	1	1	0	0	1	1	1
4	3	0	1	1	0	1	1	1	1	1	1
5	4	1	1	0	1	0	0	0	0	1	0
6	5	0	0	0	0	0	1	0	0	0	0
7	6	1	1	1	1	0	1	1	1	1	1
8	7	1	1	1	1	1	0	1	1	1	1
9	8	0	1	1	0	1	0	1	1	1	1
10	9	1	1	1	1	1	0	1	1	0	0
11	10	1	1	1	0	1	0	1	0	1	0
12	11	0	0	1	1	1	1	0	1	0	0
13	12	0	1	1	0	1	1	1	0	0	0
14	13	1	0	0	0	0	0	1	0	1	0
15	14	0	1	1	1	0	0	0	1	0	0
16	15	0	0	0	0	1	1	1	1	0	1
17	16	0	0	0	1	0	0	1	1	1	0
18	17	0	0	0	1	1	1	1	0	0	1
19	18	1	1	0	1	0	0	0	1	0	0
20	19	0	1	1	0	1	1	1	1	0	0
21	20	0	1	1	1	1	0	0	1	0	0
22	Sum	9	13	13	12	13	9	13	14	10	7

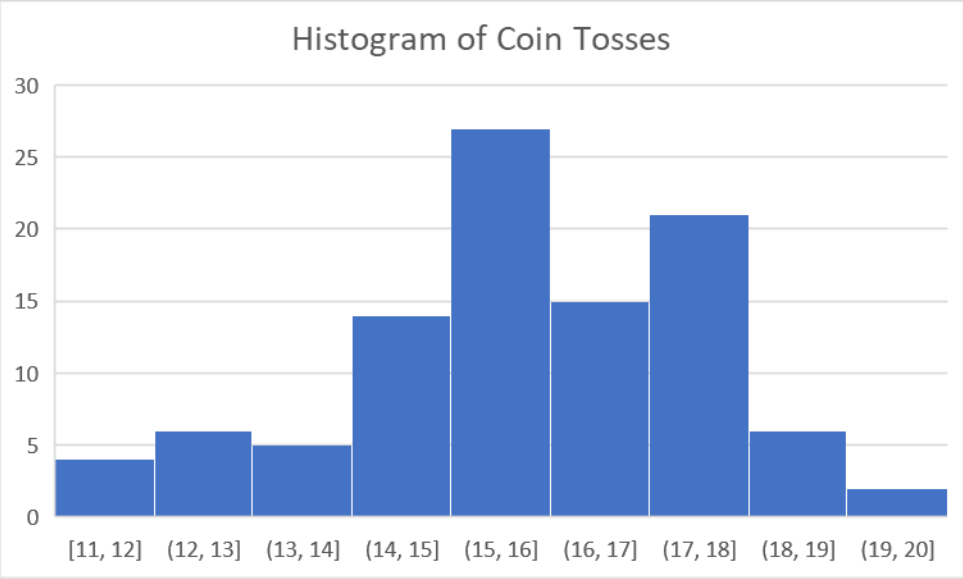
We can see a snapshot of the spreadsheet showing the successes and the sum along the bottom. The histogram below displays the results of 200 experiments of 20 tosses by their number of successes. The histogram in this case is roughly symmetric. But we might wonder what happens if we change the probability of success on each toss? What if the coin is not fair?



If we set the probability of success to 20%, the following histogram is obtained. As you can see, it is more right-skewed than symmetric.

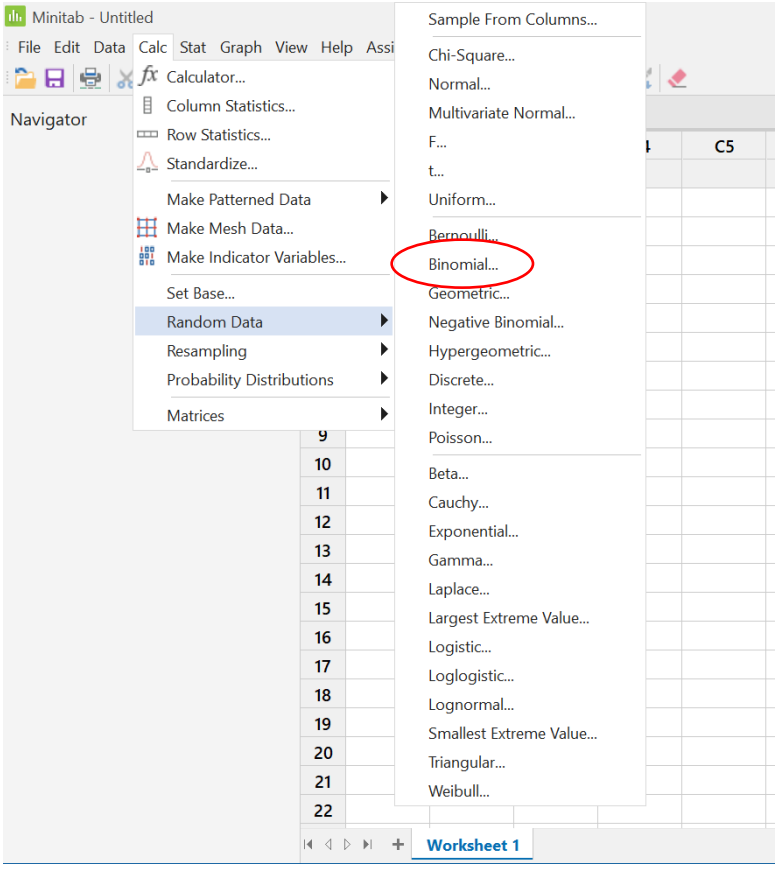


If we set the probability of success to 80%, the following histogram is obtained. As you can see, it is more left-skewed than symmetric.

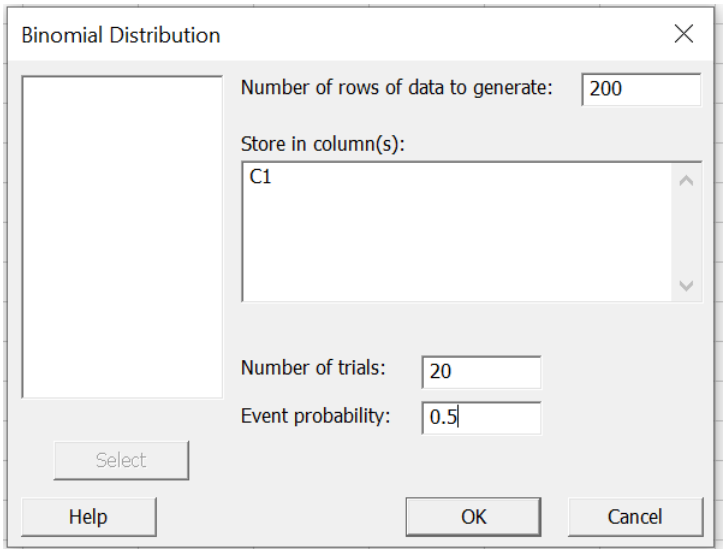


As you can see from the spreadsheet, this is based on a lot of calculations that have to be repeated over and over. Minitab makes this much easier to generate data that follows the binomial distribution so that we can see what happens if we run multiple simulations, each one just as complex as the ones above.

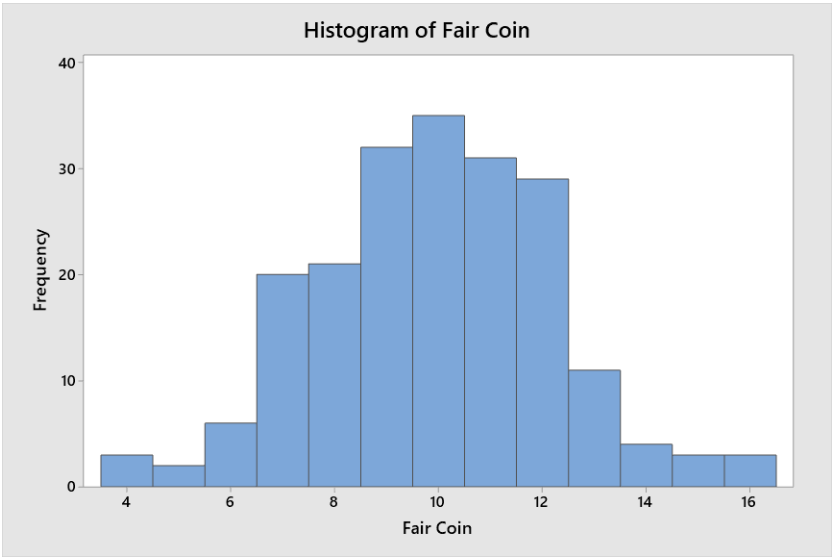
In Minitab, go to the Calc menu, then Random Data, and then select Binomial.



The following box pops up.

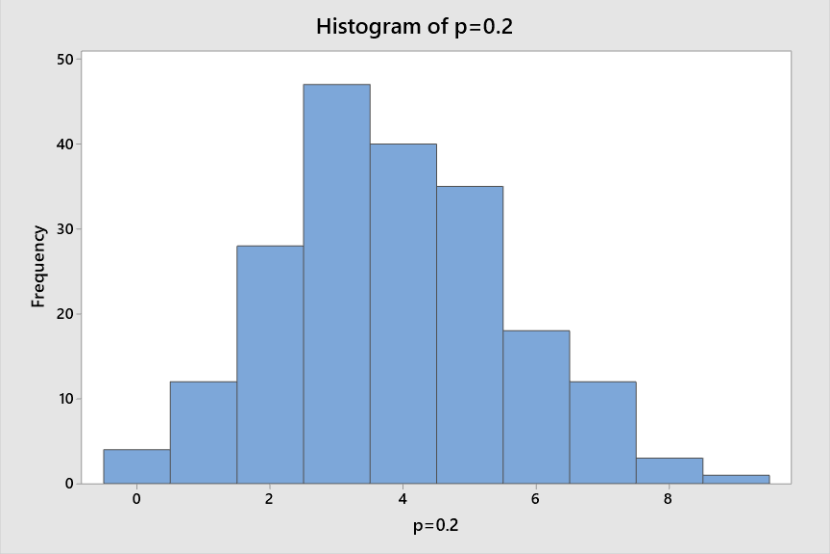


The number of experiments to run here is the same number of columns from our spreadsheet, 200. Specify the column where you want to store the data, C1. The number of trials is n, and p is the event probability. Here, we will simulate a fair coin.

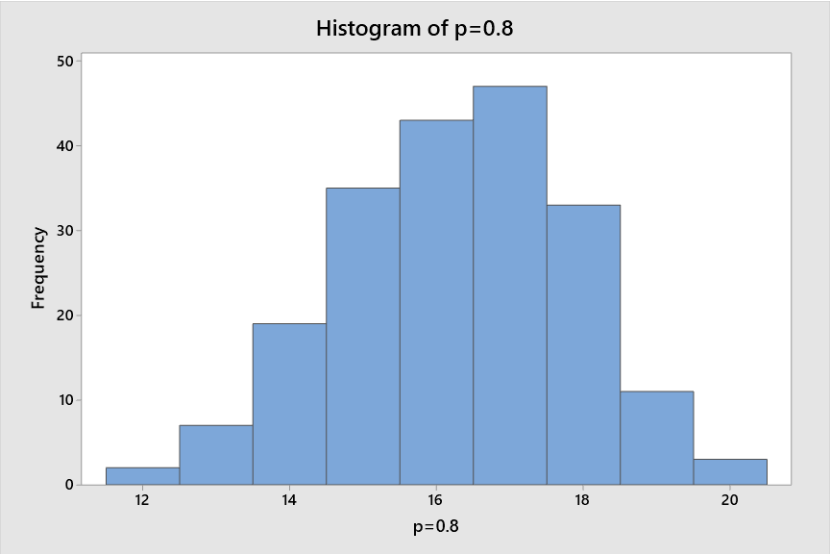


The data is pasted into the indicated column, which just shows the number of successes. And if we make a histogram, we see that the resulting graph is roughly symmetric.

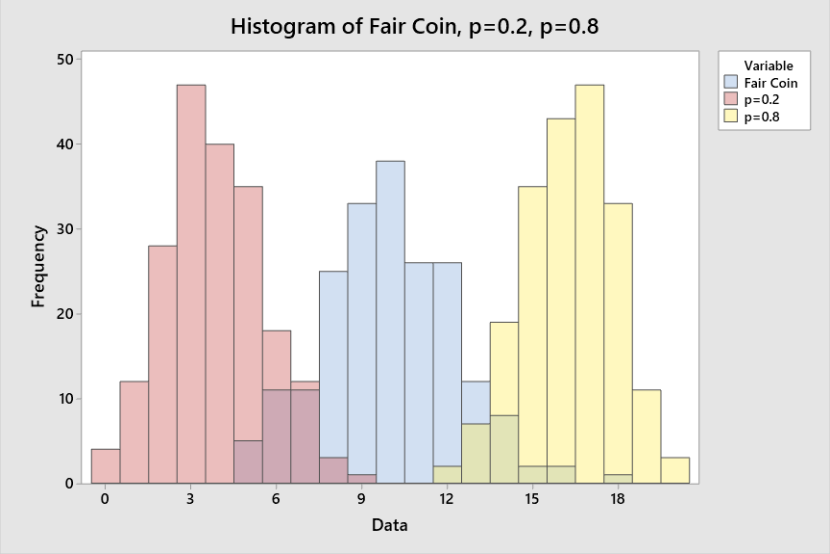
Let's test the other two probabilities. Put $p = 0.2$ in Column C2, and $p = 0.8$ in Column C3.



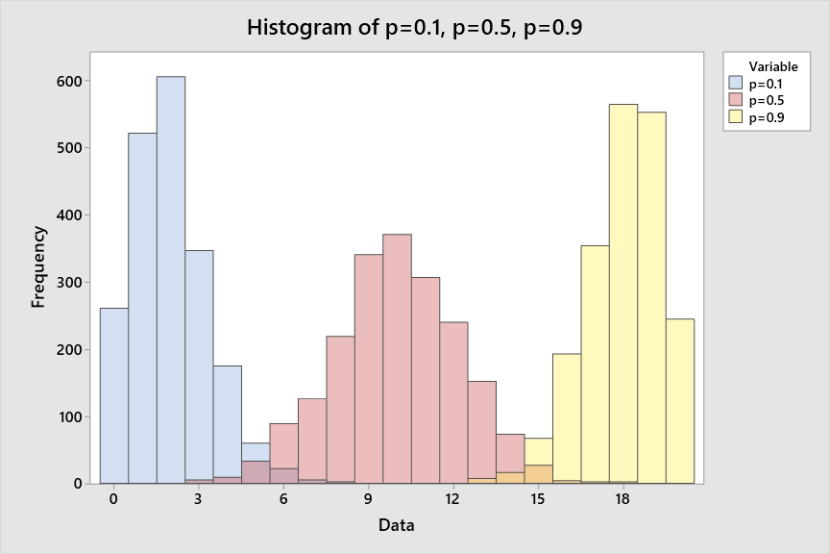
You can see there is a right-skew in the $p = 0.2$ graph, and the values have shifted far to the left, while in the $p = 0.8$ graph, the data is left-skewed and the values have shifted far to the right.



We can look at all three on the same graph in Minitab (something we can't do in Excel).



As we increase the number of experiments, and push the probabilities closer to 0 or 1, we can enhance the appearance of the skew in the distributions.



Experiment yourself with increasing the number of trials n to see how that impacts the shape of the resulting distribution. And you can use these simulations to verify that the mean of the binomial distribution is $\mu = np$ and the standard deviation is $\sigma = \sqrt{npq}$ by calculating these stats on the raw data generated.

We can do more simulations with other kinds of distributions, but we'll leave that for another handout.