# MONTE CARLO METHODS

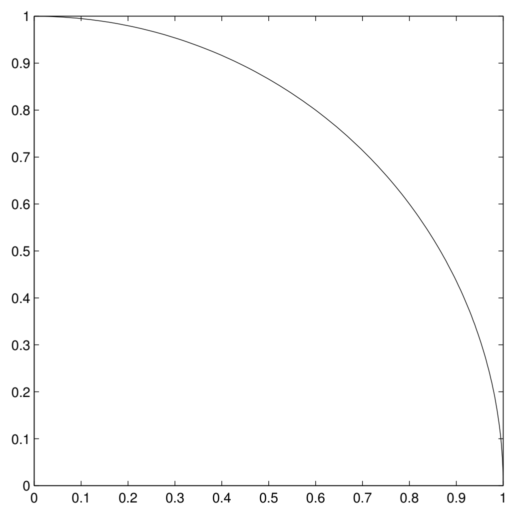
A *Monte Carlo method* is a means of using random numbers to compute something that is not random.

This method is employed in physics, engineering, biology, computer graphics and gaming, business and finance, and other areas.

An example of sampling a population at random would be to estimate the number of the 20,000 students at UH that own a car. This number of students is fixed—it's just 12,000 out of 20,000 or something. This value is not random, but it is unknown. We can try to estimate it by asking 100 random students if they own a car. If 57 out of 100 say they own a car, 57%, we predict that 0.57∗20,000 = 11,400 of all students own a car. We can improve this estimate by asking 1,000 students. Maybe by the time you've asked 10,000 students you feel like your estimate is good enough.

This is the motivation behind Monte Carlo methods, which in general look at a suitable set of random numbers and checking which fraction have a certain property.

We will look specifically at just one Monte Carlo method: a way to use random numbers to find the value of $\pi$. We will discuss this informally, and so a lot of technicalities will be brushed aside in order to give a clear illustration of how the method works.




Recall that $\pi$ is the ratio of a circle's circumference to its diameter, and it is an irrational number, i.e., its digits are infinite and have no repeating pattern. The first few digits of $\pi$ are

$$\pi \simeq 3.14159265358979323385.$$

The formula to find the area of a circle is $\pi r^2$, where $r$ is the radius of the circle. So if a circle has radius $r = 1$, its area is $\pi$. The formula to find the area of a square is $s^2$, where $s$ is the side length of the square.

Let's use a Monte Carlo method to compute $\pi$. We will look at a quarter-circle inscribed in a square with side length 1.

The area of the square is 1, and the area of the quarter-circle is $\frac{\pi}{4}$ (it's one quarter of a circle with radius 1). So if you dropped a grain of sand in the square, it has a $\frac{\pi}{4}$ probability of landing inside the quarter-circle, since

$$P(\text{landing in quarter-circle}) = \frac{\text{outcomes inside quarter-circle}}{\text{total outcomes}} = \frac{\text{area of quarter-circle}}{\text{area of square}} = \frac{\pi}{4}.$$

Our plan will be to randomly drop grains of sand into the square. The grains of sand landing within the quarter-circle represent the area of the quarter-circle. Every grain will land inside the square, so the total number of grains dropped will represent the area of the square.

Suppose that the first grain of sand randomly dropped lands within the quarter-circle. Then our estimate is $\frac{\pi}{4} = \frac{1}{1}$, or that $\pi = 4 \cdot \frac{1}{1} = 4$.

Then we drop another grain, and it randomly lands outside the quarter-circle. Then we have one grain inside, and two total. Our estimate is $\pi = 4 \cdot \frac{1}{2} = 2$.

The next two randomly land inside the quarter-circle. Now we have three inside, and four total. Our estimate is $\pi = 4 \cdot \frac{3}{4} = 3$.

In short, our method for computing $\pi$ by this Monte Carlo method is

$$\boxed{\pi \simeq 4 \cdot \frac{\text{number of grains inside the quarter-circle}}{\text{total number of grains dropped}}}$$

This process is repeated many, many times to get further estimates of $\pi$. If you were to physically conduct this experiment, you would have to ensure that the grains of sand are falling in a evenly distributed way: if more land toward the middle of the square, the outcome will be biased toward that. This is a challenging technical aspect of setting up Monte Carlo simulations.

In class, we saw this method implemented on a computer, and after 100 random points are chosen, 78 of them are inside the quarter-circle, so the estimate for $\pi$ after 100 trials is $4 \cdot \frac{78}{100} = 3.1200$, which is within 0.69% of the true value of $\pi$. The outcome is as pictured below.

One thing to notice about this method is that the estimate of $\pi$ does not always improve with more trials. Between about 40 and 60 grains of sand dropped, the estimate gets worse in this particular example. In this Monte Carlo method, we are completely at the mercy of the random number generator.

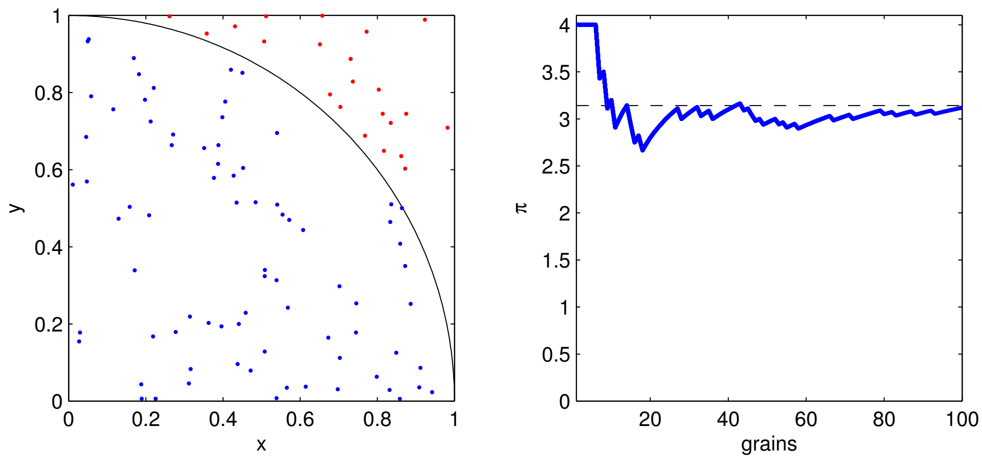



FIGURE 1. On the left, the 100 randomly-dropped grains of sand are shown. On the right, the estimated value of $\pi$ (updated each time a new grain is dropped) is shown. The dashed line shows the true value of $\pi$.