Classic Riddler

13 November 2020

Riddle:

Football season is in full swing, and with it have been some incredible blown leads. The Atlanta Falcons know a few things about this, not to mention a certain Super Bowl from a few years back. Inspired by these improbabilities, Angela wondered just how likely one blown lead truly is.

The Georgia Birds and the Michigan Felines play a game where they flip a fair coin 101 times. In the end, if heads comes up at least 51 times, the Birds win; but if tails comes up at least 51 times, the Felines win.

What's the probability that the Birds have at least a 99 percent chance of winning at *some point* during the game—meaning their probability of victory is 99 percent or greater given the flips that remain—and then proceed to lose?

Extra credit: Instead of 101 total flips, suppose there are many, many more (i.e., consider the limit as the number of flips goes to infinity). Again, the Birds win if heads comes up at least half the time. *Now* what's the probability that the Birds have a win probability of at least 99 percent at some point and then proceed to lose?

Solution:

Let k be the number of flips remaining in the game, and h be the number of heads still needed to win. The number of ways to win with exactly h out of k flips $w(k, h)$ is given by

$$w(k, h) = \binom{k}{h} = \frac{k!}{h!(k-h)!}$$

Of course, any larger number of heads will still win the game, so the number of ways to win in any fashion $W(k, h)$ is

$$W(k, h) = \sum_{i=h}^{k} w(k, i) = \sum_{i=h}^{k} \binom{k}{i}$$

But this must be turned into a probability. Each binomial coefficient is a number on Pascal's triangle, and the sum across each row of the triangle is a power of two. Specifically,

$$\sum_{i=0}^{N} \binom{N}{i} = 2^N$$

From here I can define the normalized probability of winning $P(k, h)$ (with h or more heads out of k flips). Noting the specific cases when $h = 0$ (the game is already won) and $h > k$ (winning is out of reach, and the opponent has won), the probability is

$$P(k, h) = \begin{cases} 1 & h = 0 \\ \dfrac{1}{2^k} \displaystyle\sum_{i=h}^{k} \binom{k}{h} & 0 < h \le k \\ 0 & h > k \end{cases}$$

The process used to solve this riddle in the code is relatively straightforward. During each turn (actually only some of the turns), the probability of winning is calculated, then the coin is flipped. The code keeps track of whether the 99% threshold has been met at some point in each game. After the game ends, it checks whether the game is

indeed lost and whether the threshold was met. If so, it increments the blown lead counter. This is done for many individual games.

While the process is straightforward, getting the program to work is actually very complicated. The hard step is the calculation of $P(k, h)$. For small numbers ($k \lesssim 10$), this can be calculated by hand. For larger numbers ($k \lesssim 30$), they can be calculated relatively easy by any computer program. For even larger numbers (say, $k = 101$), calculating this probability becomes extremely difficult.

First, because the binomial coefficients are defined in terms of factorials, most integer formats cannot handle these large values. Because there is no easily available math function to calculated either the coefficients or the normalized coefficients (that I know of), I have defined this calculation recursively, with the property that

$$P(k, h) = \frac{1}{2}\big(P(k-1, h) + P(k-1, h-1)\big)$$

Ultimately, this procedure is not enough, because the number of recursion steps is equal to the coefficient itself, so the number of steps is much larger than my computer's computing power. Instead, for large $k > 29$, I use an alternative calculation using the error function. This is because the binomial distribution can be approximated by the normal distribution. This approximation is valid when the mean is large, and the probability of success is not close to 0 or 1. In this case, the probability of success (heads) is $1/2$, and the mean is $k/2$. The error function is defined as

$$\mathrm{erf}(x) = \frac{2}{\pi} \int_0^x e^{-t^2}\, dt$$

with

$$x = \frac{h - 1/2 - k/2}{\sqrt{k/2}}$$

based on the way my variables are defined. This also includes a continuity correction ($h - 1/2$). Importantly, based on the way the probability is set up, with large probability for $h \ll k$ and small probability for $h \sim k$, what is actually calculated is

$$P(k, h) = \frac{1}{2}\left[1 - \mathrm{erf}\left(\frac{h - 1/2 - k/2}{\sqrt{k/2}}\right)\right]$$

There are a few other steps to save time, specifically avoiding the calculation altogether if a win is impossible, or if the 99% threshold is mathematically impossible. The best-case scenario occurs after 21 initial heads, when $k = 80$ and $h = 30$. I made this determination from Wolfram Alpha, which calculates large binomial coefficients quite easily. For larger h (or k), there's no point in calculating the probability anyway. It also doesn't bother to check again if the 99% threshold has already been met; it only needs to be met once per game to count.

Once I got the code working and producing output in useful time, I was able to get 10,000 games simulated. Of those games, 32 had a big lead that was blown (for heads, not for tails). So my very rough answer to the riddle is $\boxed{\mathbf{0.32\%}}$.