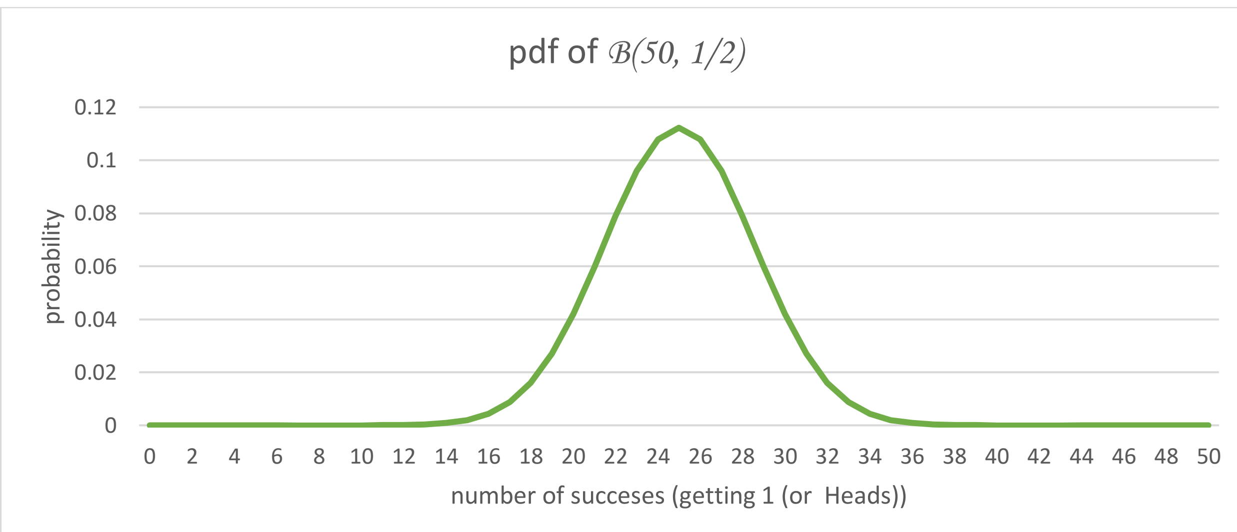
Since I started learning programming it was often said that algorithms built in programming languages can generate sequence of pseudo random numbers what imitates real random numbers. But how do we know that our pseudo random numbers “behave” themselves as real ones?

Let’s simplify this situation by taking not any numbers but only from $\{0, 1\}$. Before reading further generate a random sequence of 0 and 1 (for instance, with length 50) using any programming language. Then write down 50 numbers choosing from $\{0, 1\}$ by yourself. Create the third sequence flipping a fair coin and writing down 1 if it shows up Heads and 0 otherwise. Can you now distinguish where is your sequence and where are the real random ones (theoretically there is no way to find which of them is program generated sequence)? How many successes (1 or Heads) should be in the real random sequence – 25? Or maybe 24?

According to binomial distribution of this event,



We see, that probability to get (22 or 28) Heads is equal to 0.16, when $P(25) = 0.11$. Actually, there is only 0.52 chance to get value in range 23:27. So, we can’t use this information to analyze a small number of experiments.

RUNS

There is one more thing to characterize sequences – runs. How many runs do you think should have random sequence and what are their lengths? And what about runs in your sequence?

Let’s generalize our case taking n coin flips instead of 50 random zeroes and ones. Therefore, the fist flip will start a run. Then, we have $n - 1$ flips, which can start new run (in case, coin shows up other side but not the same as in previous flip) with probability $\frac{1}{2}$. Let's denote S_n for number of flips when coin switch its value to opposite from previous one. It has binomial distribution $\mathcal{B}(n-1, 1/2)$ and expected value $E[S_n] = \frac{n-1}{2}$. Number of values switching also means number of *new* runs, so total number of runs is equal to $R_n = S_n + 1$ with expected value $E[R_n] = E[S_n + 1] = \frac{n-1}{2} + 1 = \frac{n+1}{2}$.

Next, let's count the average length of run. As total length of all runs is n, average one is counting the following way $L_n = \frac{n}{R_n}$, and expected value: $E[L_n] = \frac{n}{\frac{n+1}{2}} = \frac{2n}{n+1} = \frac{2n+2-2}{n+1} = 2 - \frac{2}{n+1}$.

Looks good, doesn't it? It doesn't.

$E[\frac{1}{R_n}]$ isn't equal to $\frac{1}{E[R_n]}$. As we remember, S_n has a binomial distribution and we should use it to calculate expected value of average length of runs:

$$E[L_n] = \sum_{k=1}^{n} \frac{n}{k}(R_n = k) = \sum_{k=1}^{n} \frac{n}{k}(S_n = k-1) = \sum_{k=1}^{n} \frac{n}{k}\binom{n-1}{k-1}\left(\frac{1}{2}\right)^{n-1}$$
$$= \frac{1}{2^{n-1}}\sum_{k=1}^{n}\binom{n}{k}1^k = \frac{1}{2^{n-1}}((1+1)^n - 1) = 2 - \frac{1}{2^{n-1}}.$$

So, $E[L_n]$ is in range from $[1, 2)$, but it converges to 2 as $n \to \infty$.

MAXIMAL RUN LENGTH

Can you suggest how long will be the longest run?

Let's denote M_n for maximum run length. As for one flip for every $1 \leq k$ we cannot have run with length less than 1 or larger than k, $P(M_1 \leq 0) = 0, P(M_1 \leq k) = 1$. To count these probabilities – $P(M_n \leq k)$, where $k \leq n$ – we can use recursion. To do this, we can use complement event to previous one: take $k \geq n$, then $P(M_n \leq k) = 1$.

Let us have sequence of n flips and $M_n \leq k$. For one new flip, we'll have still $M_{n+1} \leq k$ but not $M_{n+1} \leq k + 1$, unless the last k flips from sequence form a run and our new flip $n + 1$ matches the previous last flip n. For this, to be correct, the first $n - k$ flips should have all runs less than k: $M_{n-k} \leq k$, when $k < n$. After this initial subsequence, the next k flips can make a run with length $k+1$ if $(n-k+1)^{th}$ flip differs from $(n-k)^{th}$ one and all the remaining flips matches to previous one. This happens with probability $\frac{1}{2^{k+1}}$, as there are k flips remaining + one additional flip. So, we can come from $M_n \leq k$ to $M_{n+1} \leq k+1$ with probability

$$\frac{P(M_{n-k} \leq k)}{2^{k+1}}$$

Now, our recursive formula, when $k < n$, looks the following way

$$P(M_{n+1} \leq k) = P(M_n \leq k) - \frac{P(M_{n-k} \leq k)}{2^{k+1}}.$$

However, we cannot use this formula for $n = k$, as M_0 isn't defined:

$$P(M_{n+1} \leq k) = P(M_n \leq k) - \frac{P(M_{n-k} \leq k)}{2^{k+1}} = 1 - \frac{P(M_0 \leq k)}{2^{k+1}}$$

So, let's observe this case (when $n = k$). The fact is, that for sequence with length $n+1$, there are 2 possible outcomes when $M_{n+1} = n+1$: all heads or all tails. As total number of outcomes is 2^{n+1}, $P(M_{n+1} = n+1) = \frac{2}{2^{n+1}}$. And, it's obvious, $P(M_{n+1} \leq n+1) = 1$. Therefore,

$$P(M_{n+1} \leq n) = P(M_{n+1} \leq n+1) - P(M_{n+1} = n+1) = 1 - \frac{2}{2^{n+1}}.$$

Despite the rules of probability theory, let's declare $P(M_0 \leq k) = 2$. It won't influence on most of our results, but it will help to hold the case $k = n$.