Counting Fibonacci Numbers: What the Density Leaves Behind

J. R. Landers

November 15, 2026

How many of the integers from 1 to k are Fibonacci numbers? We can generate them and count. Their exponential growth also suggests a smooth density, which replaces the loop with a logarithm. An exact formula then tells us how much the density misses. The resulting error has a story of its own: it records the nearly regular spacing of Fibonacci numbers on a logarithmic scale.

1. Count first, then smooth

Let F_n denote the Fibonacci number at index n, with indexing starting at zero. The sequence begins with $F_0 = 0$ and $F_1 = 1$; subsequent entries satisfy $F_{n+2} = F_{n+1} + F_n$ for integers $n \geq 0$. We count distinct positive values, so the relevant list is $F_2, F_3, \ldots = 1, 2, 3, 5, \ldots$; the repeated 1 is counted once. For an integer cutoff $k \geq 1$, define the count $N(k)$ and proportion $p(k)$ by

$$N(k) = \#\{n \geq 2 : F_n \leq k\}, \qquad p(k) = \frac{N(k)}{k}. \tag{1}$$

Here # counts the elements of a set. Equivalently, $p(k)$ is the probability that an integer chosen uniformly from $\{1, \ldots, k\}$ is Fibonacci. The direct algorithm stores consecutive Fibonacci values a, b and a count r, starting with $(a, b, r) = (1, 2, 0)$, and repeatedly makes the simultaneous update

$$(a, b, r) \longleftarrow (b, a + b, r + 1) \quad \text{while } a \leq k,$$

then returns r/k. It visits only Fibonacci numbers, rather than scanning all k integers.

Let $\varphi = (1 + \sqrt{5})/2$ be the golden ratio, and write $\log_\varphi x = \ln x / \ln \varphi$, where ln is the natural logarithm. Binet's classical formula, obtained from the two roots of the recurrence's characteristic equation, gives [1]

$$F_n = \frac{\varphi^n - (-\varphi)^{-n}}{\sqrt{5}} \sim \frac{\varphi^n}{\sqrt{5}}. \tag{2}$$

The symbol $\sim$ means that the ratio tends to one as $n \to \infty$. Thus $N(k) = \log_\varphi k + O(1)$, and the loop takes $\Theta(\ln k)$ arithmetic steps. Here O denotes an upper bound up to a constant factor, and Θ matching upper and lower bounds, as $k \to \infty$. For a real cutoff $x \geq 1$, inverting the leading growth law gives a smooth index estimate $n(x)$; differentiating suggests

$$n(x) \approx \log_\varphi(\sqrt{5}x), \qquad \rho(x) = \frac{1}{x \ln \varphi}.$$

Here ρ is a smooth *counting density*: an approximate number of Fibonacci values per unit interval. It is not a normalized probability density. Integrating it from 1 to k produces the simple estimates

$$\widetilde{N}(k) = \int_1^k \rho(x)\, dx = \log_\varphi k, \qquad \widetilde{p}(k) = \frac{\ln k}{k \ln \varphi}. \tag{3}$$

Integration leaves an additive constant undetermined; this choice retains just the leading term. Both proportions tend to zero, at rate $(\ln k)/k$. The logarithm has a simple interpretation: multiplying the cutoff by φ adds approximately one Fibonacci number. Doubling the cutoff therefore adds only a bounded number, even when the interval contains millions of new integers. The density captures this thinning of the sequence without locating any individual member.

The density estimate uses a fixed number of operations if logarithms are treated as primitive operations. This is a useful shortcut, not a claim of constant bit complexity. If $\ell = \lceil \log_2(k+1) \rceil$ is the binary input length ($\lceil y \rceil$ is the least integer at least y), the direct loop uses $\Theta(\ell^2)$ bit operations with ordinary addition and $\Theta(\ell)$ bits of working storage. Evaluating logarithms also has a precision cost.

2. How much does the density miss?

The estimated count is always within one of the answer:

$$\big|N(k) - \log_\varphi k\big| \le 1, \qquad \big|p(k) - \widetilde{p}(k)\big| \le \frac{1}{k}. \tag{4}$$

To see this, induction gives $\varphi^{n-2} \le F_n \le \varphi^{n-1}$ for $n \ge 2$. Let m be the largest index with $F_m \le k$, so $k < F_{m+1}$, $N(k) = m - 1$, and $m - 2 \le \log_\varphi k < m$, proving the claim. The relative error $|\widetilde{p}(k) - p(k)|/p(k)$ is at most $1/N(k)$, hence $O(1/\ln k)$: small absolute error need not mean equally small relative error when Fibonacci numbers are scarce.

k	Exact $p(k)$	Density $\widetilde{p}(k)$	Relative error
10	50%	47.8497%	4.30%
100	10%	9.5699%	4.30%
1,000	1.5%	1.4355%	4.30%
1,000,000	0.0029%	0.002871%	1.00%

The table suggests a good approximation, but it does not identify which side of a counting step a given cutoff occupies. We can recover that information exactly without generating the sequence. For integers $k \ge 1$, define the auxiliary logarithmic index $u(k)$ below; $\lfloor y \rfloor$ denotes the greatest integer at most y:

$$u(k) = \log_\varphi\Big(\sqrt{5}\,\big(k + \tfrac{1}{2}\big)\Big). \qquad \boxed{p(k) = \frac{\lfloor u(k) \rfloor - 1}{k}.} \tag{5}$$

Indeed, (2) implies $|F_n - \varphi^n/\sqrt{5}| = \varphi^{-n}/\sqrt{5} < 1/2$, so F_n is the nearest integer to $\varphi^n/\sqrt{5}$. The half-unit shift in $u(k)$ comes from this rounding: $k + 1/2$ is the boundary between real numbers rounding to k and those rounding to $k+1$. Thus a Fibonacci value is at most k precisely when its exponential approximation lies below that boundary. For $n \ge 2$,

$$F_n \le k \quad\Longleftrightarrow\quad \frac{\varphi^n}{\sqrt{5}} < k + \tfrac{1}{2} \quad\Longleftrightarrow\quad n < u(k).$$

The middle expression cannot be an equality: its left side lies strictly within one half of an integer. Consequently $u(k)$ is not an integer, the largest admissible index is $\lfloor u(k) \rfloor$. The indices we count run from 2 through this index, giving $\lfloor u(k) \rfloor - 1$ distinct values and hence (5). For example, $u(100) \approx 11.253$, so the largest index is 11 and the count is $11 - 1 = 10$: the Fibonacci values through $F_{11} = 89$. The next value, $F_{12} = 144$, exceeds the cutoff.

The formula is exact mathematically; a numerical implementation must control rounding before applying the floor. All three approaches exploit the same exponential growth. The loop follows individual values, the density retains their smooth growth rate, and the floor formula restores the precise cutoff.

3. The error is a pattern

Having the exact answer does not make the approximation uninteresting. Define the count error $E(k) = N(k) - \log_\varphi k$. On extending the cutoff to a real variable $x \geq 1$, the count $N(x) = \#\{n \geq 2 : F_n \leq x\}$ is constant between successive Fibonacci numbers. There,

$$\frac{d}{dx}\big(N(x) - \log_\varphi x\big) = -\frac{1}{x \ln \varphi}.$$

At the next Fibonacci number the error jumps upward by one. On the horizontal axis $t = \log_\varphi x$, the falling pieces have slope -1: the error is a sawtooth. Moreover, (2) gives the exact jump locations

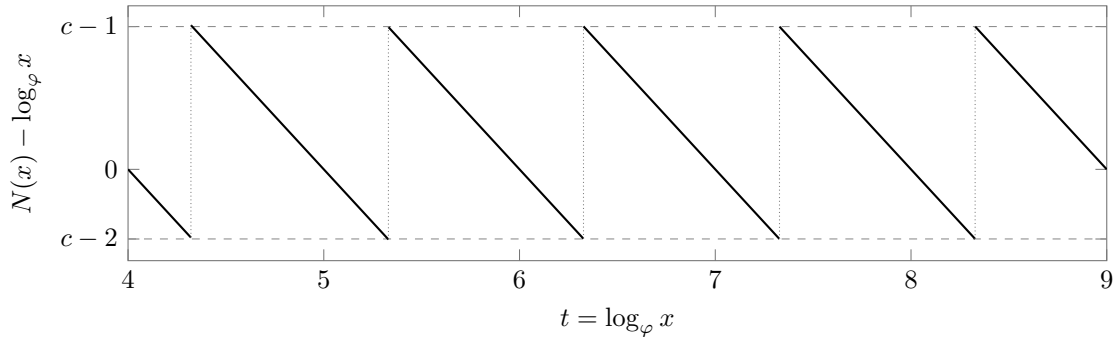
$$\log_\varphi F_n = n - c + \log_\varphi\big(1 - (-1)^n \varphi^{-2n}\big), \qquad c = \log_\varphi \sqrt{5}. \tag{6}$$

These locations approach a lattice of spacing one. Multiplication of the cutoff by φ therefore approximately repeats the error pattern. The small correction alternates in sign with n and decays as φ^{-2n}: the second root $-1/\varphi$ of the recurrence leaves a faint odd–even signature.

The oscillation persists. At $k = F_n$ the error tends to $c - 1$; at $k = F_{n+1} - 1$ it tends to $c - 2$. Since it decreases between these endpoints, its lower and upper limiting values (lim inf and lim sup) are

$$\liminf_{k\to\infty} E(k) = c - 2 \approx -0.328, \qquad \limsup_{k\to\infty} E(k) = c - 1 \approx 0.672. \tag{7}$$

Thus the count error does not vanish, although the proportion error $E(k)/k$ does. Looking only at proportions would conceal this persistent structure.



The count error for a real cutoff. Dotted segments mark unit jumps; dashed lines show the limiting upper and lower levels.

4. Centering the sawtooth

There is even a sharper smooth estimate hiding in the exact formula. Write $\{y\} = y - \lfloor y \rfloor$ for the fractional part of a real number y. Substituting $\lfloor u(k) \rfloor = u(k) - \{u(k)\}$ into the exact count gives

$$N(k) = u(k) - 1 - \{u(k)\}.$$

The fractional part lies between zero and one. Replacing it by the midpoint $1/2$ therefore defines the centered smooth count estimate

$$N_*(k) = u(k) - \tfrac{3}{2}.$$

This choice balances the two sides of the error: as the counting step is traversed, the residual falls from near $1/2$ to near $-1/2$. In particular,

$$N(k) - N_*(k) = \tfrac{1}{2} - \{u(k)\}, \qquad \left| p(k) - \frac{N_*(k)}{k} \right| \leq \frac{1}{2k}. \tag{8}$$

The bound holds for every integer $k \geq 1$, without assuming any distribution for the cutoff. For $k = 100$, the centered estimate gives $N_*(100) \approx 9.753$, compared with the exact count 10 and the original density estimate 9.570.

The improved estimate also explains the constant that was lost on integrating the density. With $c = \log_\varphi \sqrt{5}$ as before, it can be written

$$N_*(k) = \log_\varphi k + \left(c - \tfrac{3}{2}\right) + \log_\varphi \left(1 + \frac{1}{2k}\right).$$

The leading correction is $c - 3/2 \approx 0.1723$; the final logarithm tends to zero. Retaining both terms gives the uniform bound in (8). Thus examining the residual improves the approximation using the same fixed number of logarithm evaluations.

Centering the sawtooth halves the original uniform error bound. Counting, smoothing, and inversion have led to the same object from three directions: the density describes the growth, while its residual preserves the discrete spacing. These conclusions are elementary consequences of the classical Fibonacci formula; their interest here is what becomes visible when the approximation itself is examined.

References

[1] D. Grinberg, *Math 221, Winter 2023, Lecture 4: Induction*, version April 28, 2026, Theorem 1.8.6. `https://www.cip.ifi.lmu.de/~grinberg/t/23wd/lec04.pdf`.