Fibonacci Partitions and Variance

Owen Jones

September 23, 2023

Introduction

There are many ways that we can generate a given positive whole number by adding smaller, distinct numbers together. For instance, $7 = 4 + 3$ but also $7 = 6 + 1 = 4 + 2 + 1$ and $7 = 5 + 2$. Notice that in the last example, the two numbers added together are examples of Fibonacci numbers. In other words, they appear in the sequence $0, 1, 1, 2, 3, 5, 8, ...$, in which the next term is produced by adding together the two previous terms. In doing this 'generating' of 7, sometimes people will say that we have 'partitioned' 7, since we have separated 7 into different 'parts'. If each of the parts (whole numbers) is a Fibonacci number, then we call this a Fibonacci partition.

Counting partitions

Throughout the last 60 years, the question: "How many Fibonacci partitions are there for each given positive whole number?" has been investigated by mathematicians. Indeed, the idea of counting these Fibonacci partitions was first mentioned all the way back in 1963, in the very first edition of the Fibonacci Quarterly, a journal whose papers focus on Fibonacci numbers. Another way of stating this question is to ask: "For any given number, how many ways can I generate it by adding together different Fibonacci numbers?". This question is therefore easy to state, but is there actually an easy formula to work this out?

Unfortunately, the answer is no. Mathematicians have proven formulas which give us the number of Fibonacci partitions for each whole number, although these are all very complicated to write down. The graph on the next page shows just how randomly the number of these partitions changes as our whole number gets larger. This indicates the difficulty that arises when trying to find a simple 'rule' to help us count partitions.

Why so random?

It is important to recognise that the Fibonacci sequence grows exponentially. As we saw above, the initial terms do not seem that big, but, for instance,

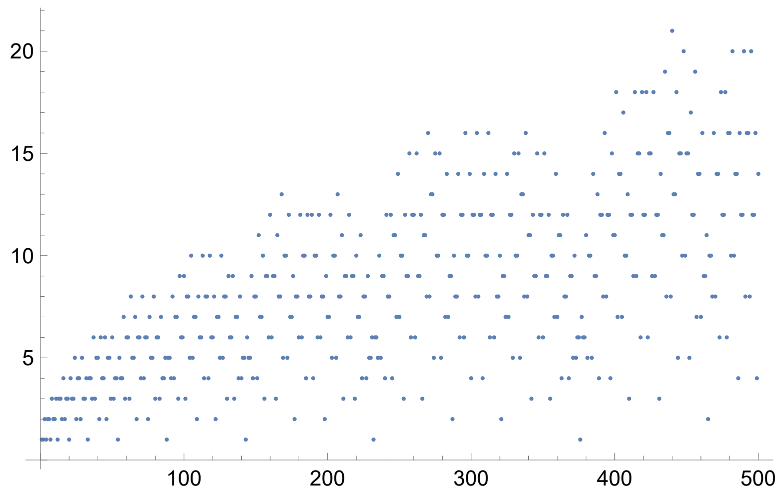

Figure 1: Number of Fibonacci partitions for n against n

only the $20th$ Fibonacci number is as large as 6765. Now, we need larger Fibonacci numbers to be part of our partition in order to generate greater numbers on the number line. This means that, although we may expect the number of Fibonacci partitions to be greater for a larger whole number (since there are 'more smaller parts' behind it), this is countered by the fact that larger Fibonacci numbers (which we need to use) are far more thinly spaced out along the number line.

Variance of the data

There has been some success in studying various quantities associated with the number of Fibonacci partitions per whole number. We will call our whole number n and its number of Fibonacci partitions $R(n)$. Let's also define the average number ('total divided by how many numbers there are') of partitions per whole number up to n to be $M(n)$. For example,

$$M(4) = \frac{R(1) + R(2) + R(3) + R(4)}{4}$$

It turns out that the behaviour of this average is far less random (although an exact formula is still hard to write out).

Besides averages, statisticians often look into the 'range' and 'spread' of data in a dataset. One quantity in particular that is frequently mentioned is the *variance*, which measures how far data lies from its mean value. To illustrate this, consider the numbers $2, 3, 3, 3, 4$. These are all quite close to the mean of 3, so we would say that the variance is low. Inspired by the relatively successful investigation of the average number of partitions up to n, we might try to think about how the variance behaves. This will be particularly interesting to look at because of the wide variety of values that $R(n)$ takes that are far from the mean value. The usual way to calculate variance is to compute "The mean of the squares minus the square of the mean". As an illustration, if, again, we were going up to 4, we would need to find

$$\frac{R(1)^2 + R(2)^2 + R(3)^2 + R(4)^2}{4} - \left(\frac{R(1) + R(2) + R(3) + R(4)}{4}\right)^2$$

Since we already understand the average, this problem reduces to understanding the behaviour of the sum of squares $V(n) = R(1)^2 + ... + R(n)^2$.

Pairs of solutions

On the face of it, figuring out how to start this problem might seem tricky. After all, we do not yet have a good understanding of what the quantity $R(n)^2$ really *means* in terms of counting partitions. To give us an interpretation of $R(n)^2$, it is useful to consider $R(n)$ as the *number of solutions* to a particular equation. The precise details of the complete equation for any n are a bit technical , but we can think of it as a more general version of something like $x + y + z = n$ where x, y and z are distinct Fibonacci numbers; in the general equation we have more than three possible variables (ie more symbols on the left added together than just x, y and z), since we could add more than just three Fibonacci numbers together. Now, imagine you treat each solution as one 'package' and bundle all the packages into a 'box'. Suppose we have two such boxes and you pair up each solution from the first box with a solution from the second box. How many ways are there to do this? Well, there are $R(n)$ possibilities for the package from the first box, and for each of these, there are precisely $R(n)$ packages that it could be paired with from the second box. This gives $R(n) \times R(n) = R(n)^2$ possible (ordered) pairs. For example, if there are just three solutions S_1, S_2, S_3 to our general equation then I can represent the box as $\{S_1, S_2, S_3\}$. The list of pairs would then be

$$(S_1, S_1), (S_1, S_2), (S_1, S_3), (S_2, S_1), (S_2, S_2), (S_2, S_3), (S_3, S_1), (S_3, S_2), (S_3, S_3)$$

so there would be $3 \times 3 = 9$ possibilities.

Case by case analysis

What, then, can be said of the quantity V defined already? Well, if we were to calculate $V(n)$, we would go through each whole number from 1 to n and deal with finding the number of pairs from the 'boxes' one by one, before adding all of these pairs together. Therefore, it is possible to interpret V as the number of solutions to an inequality which looks something like this:

$$0 < x + y + z = a + b + c \leq n$$

so the a, b, c form one solution and the x, y, z make up another, and it does not matter which whole number this pair of solutions is associated with as long as it is between 1 and n. To solve this, it is best to start with the case of n being a Fibonacci number. Suppose $n = F_m$, the mth Fibonacci number. If we can find a general formula for $V(F_m) - V(F_{m-1})$ then we could use this to find $V(F_m)$. This amounts to solving something like

$$F_{m-1} < x + y + z = a + b + c \leq F_m$$

Still, it is not obvious where to begin. What size can our Fibonacci numbers be so that they are within this range? Well, when summing Fibonacci numbers, it is possible to prove (using a method called induction) that, for any positive whole number m, the sum of all the distinct Fibonacci numbers from 1 to F_{m-2} is equal to $F_m - 2$ which is of course less than F_m. For example,

$$F_2 + F_3 + F_4 = F_6 - 2 < F_6$$

This might seem surprising but, as we noted before, the Fibonacci numbers grow very fast, so even bumping $m - 2$ up by 2 to get to m causes an enormous growth in the sequence - a growth so large that even if you added together *every single* Fibonacci number up to $m-2$ you *still* would not reach F_m. This inequality means that in order to be at least as large as F_{m-1}, the biggest Fibonacci number in the sum must be at least F_{m-2} (even if, say, instead of x, y and z we had 10 different variables) since summing all the Fibonacci numbers up to F_{m-3} would create a number that is below F_{m-1}. For example, in the case $m = 7$, we see that we can not have F_4 as our largest Fibonacci number. Note also that we cannot have any Fibonacci number in the sum exceeding F_m itself, since otherwise the sum would be too big. When $m = 7$, this means we can not have anything like F_8 as our highest Fibonacci number. As a result of this analysis, we can start breaking down the problem into different cases depending on what the largest Fibonacci number is in each sum, and counting the number of solutions in each case.

A formula for $V(F_m)$

Due to the fact that $F_m = F_{m-1} + F_{m-2}$, it turns out that each case reduces us to another equation related to V or R. In our inequality above, imagine I replace both c and z with F_{m-1} (one of the 'cases'). We then see that the inequality can be simplified to get

$$0 < x + y = a + b \leq F_{m-2}$$

which is like $V(F_{m-2})$. Consequently, we can derive what is known as a *recursive* formula. That is, a formula for $V(F_m)$ which depends on the values taken by V when smaller numbers are used, so $V(F_m)$ might depend on $V(F_{m-4})$ for example.

Recursive formulas appear in many different contexts in maths, like population growth. The simplest form of a recurrence is something like $x_k = 2 \times x_{k-1}$ where, perhaps, x_k represents the number of animals in a habitat in year k. Thus the amount of animals there doubles every year. Therefore, if you start at year 0, the total number of animals by year k would be 2 multiplied by itself k times. In other words, $x_k = 2^k \times x_0$. Consequently, the population increases exponentially year on year. The recurrence for $V(F_m) - V(F_{m-1})$ is far more complicated than this, but it can be solved and also gives us a formula for $V(F_m)$ involving exponential terms. Therefore, as m increases, the quantity $V(F_m)$ increases exponentially. It turns out that $V(F_m) \approx u \times v^m$ – where $u \approx 0.074$ and $v \approx 2.48$ – for very large m.

How big is $V(n)$?

This therefore gives a pretty satisfying result for $V(n)$ when n is a Fibonacci number, but we still need some information about what happens in other cases. However, this does not create too much trouble, since we can always find some Fibonacci number F_m such that $F_{m-1} \leq n \leq F_m$ and, since $V(n)$ gets larger whenever n gets larger (it is the sum of terms *up to* n after all), we must have $V(F_{m-1}) \leq V(n) \leq V(F_m)$. Then, using the fact that $F_m \approx \sqrt{5} \times \phi^m$, we can show that $V(n)$ grows at the same rate as n^p where $p \approx 2$. More precisely, $0.3 \times n^p < V(n) < 0.4 \times n^p$.

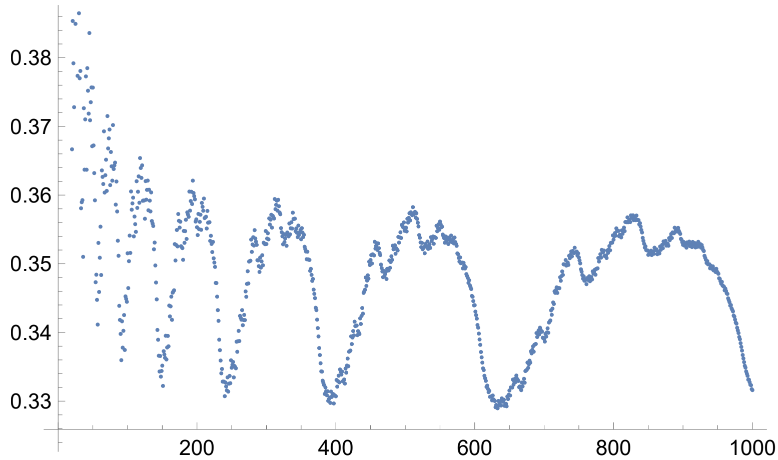

Figure 2: $\frac{V(n)}{n^p}$ against n