MAT344 Week 3

2019/May/20

1 This week

This week, we are talking about

1. Recursion
2. Induction

2 Recap

Last week we talked about

1. Binomial Coefficients
2. Stars and bars
3. Lattice paths
4. The binomial theorem

3 Recursion (Chapter 3.4 in [KT17])

Let's find a formula for the number of ways of triangulating a convex polygon. We start by computing some small examples. We see that In addition to the 1 way of triangulating the triangle, there are 2 ways to triangulate a

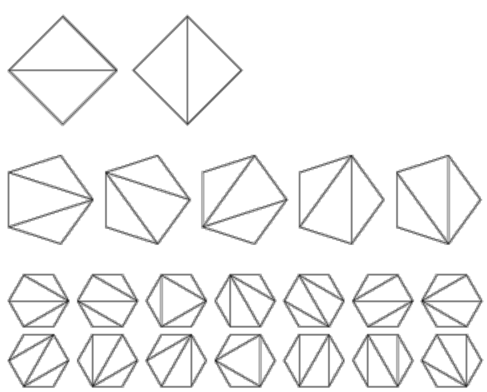

Figure 1: Triangulations of convex polygons

square, 5 to triangulate a pentagon and 14 to triangulate the hexagon. These agree with the number of Dyck paths for $n = 1, 2, 3, 4$. We would like to prove this, but there does not seem to be an obvious combinatorial proof.

Let us try a different method. We will find a way to express C_n in terms of C_ks with $k \leq n$. This is known as a **recurrence relation**. Then if any other counting problem satisfies the same recurrence relation and agrees with our values in small examples, the answer must be C_n.

Let us clarify what we mean here. Recall that we defined factorials as

$$n! = n(n-1)(n-2)\dots 2\cdot 1.$$

Technically, this is not quite a flawless definition, since multiplication should be a *binary operation*, i.e. you are only supposed to multiply two numbers at a time. We could instead say

$$n! = n\cdot(n-1)!.$$

It seems like that we are just pushing the problem one step further but if we also define $0! = 1$, then we see that (after n steps), we can find the value of $n!$. The formula $n! = n\cdot(n-1)!$ that lets us compute the value of a function (the one sending a number n to $n!$) in terms of other values of the function is called a **recursive formula**. We call this process **recursion**.

Recursion can involve more than one variable, for example, we proved the identity

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1} \tag{1}$$

with binomial coefficients. If we define $\binom{n}{n} = \binom{n}{0} = 1$ for all n, then we can use (1) to compute binomial coefficients.

Exercise 3.1. *What does this recursion look like in Pascal's triangle?*

Let us get back to Catalan numbers that we know count Dyck paths. Notice that after the first up step, any Dyck path will touch the diagonal at some point. Let (k,k) be the first time this happens. Then the $2k$-th step must have been a up step. Since this is the first time we are touching the diagonal, our path from $(1,0)$ to $(k,k-1)$ never crosses the line $y = x-1$, so we may remove the first and last step from this initial segment and end up with a Dyck path from $(0,0)$ to $(k-1,k-1)$, in addition to the tail of the path (from (k,k) to (n,n)), which itself can be considered as a Dyck path from $(0,0)$ to $(n-k,n-k)$. Counting Dyck paths this way we obtain the recursive formula

$$C_n = \sum_{k=1}^{n} C_{k-1}C_{n-k}.$$

It is customary to reindex this as

$$C_{n+1} = \sum_{k=0}^{n} C_k C_{n-k}. \tag{2}$$

Let's check that this agrees with what we know about Catalan numbers. We define $C_0 = 1$ and we compute

$$\begin{aligned}
C_0 &= 1\\
C_1 &= C_0C_0 = 1\\
C_2 &= C_0C_1 + C_1C_0 = 2\\
C_3 &= C_0C_2 + C_1C_1 + C_2C_0 = 5\\
C_4 &= C_0C_3 + C_1C_2 + C_2C_1 + C_3C_0 = 14
\end{aligned}$$

We want to show that triangulations of polygons satisfy the same recurrence. The relation suggests that from a triangulation of an $n+2$-gon, we should produce two other polygons (with triangulations), a k-gon, and an $n-k+1$-gon, since we expect $j+2$-gons to be counted by C_j. Number the vertices from 1 to $n+2$ and focus on the single external edge between the two vertices 1 and $n+2$. This is part of a triangle in the triangulation, and it connects to, say, vertex k. The rest of the polygon is now split into two polygons (with triangulations). Note that if $k = 2$ or $k = n+1$ one of these polygons is empty, and we have just removed a triangle from our $n+2$-gon to get a triangulation of an $n+1$-gon. If $3 \le k \le n$, then we get a k-gon on one side and an $n-k+1$-gon on the other side (both with triangulations).To check that this is a bijection, note that we can put the triangulations of the k-gon and $n-k+1$-gon together with the triangle connecting vertices $1, k$, and $n+2$ to recover the triangulation of the $n+2$-gon. This shows that the number of triangulations of a convex polygon satisfy the same recurrence, and therefore we conclude that the number of triangulations of a convex $n+2$-gon is C_n.

4 Fibonacci numbers

The famous Fibonacci sequence starts like this:

$$1, 1, 2, 3, 5, 8, 13, \ldots$$

The rule defining the sequence is $F_1 = 1, F_2 = 1$, and for $n \geq 3$,

$$F_n = F_{n-1} + F_{n-2}.$$

This is a recursive formula. As you might expect, if certain kinds of numbers have a name, they answer many counting problems.

Exercise 4.1 (Example 3.2 in [KT17]). *Show that a $2 \times n$ checkerboard can be tiled with 2×1 dominoes in F_{n+1} many ways.*

Solution: Denote the number of tilings of a $2 \times n$ rectangle by T_n. We check that $T_1 = 1$ and $T_2 = 2$. We want to prove that they satisfy the recurrence relation

$$T_n = T_{n-1} + T_{n-2}.$$

Consider the domino occupying the rightmost spot in the top row of the tiling. It is either a vertical domino, in which case the rest of the tiling can be interpreted as a tiling of a $2 \times (n-1)$ rectangle, or it is a horizontal domino, in which case there must be another horizontal domino under it, and the rest of the tiling can be interpreted as a tiling of a $2 \times (n-2)$ rectangle. Therefore

$$T_n = T_{n-1} + T_{n-2}.$$

Since the number of tilings satisfies the same recurrence relation as the Fibonacci numbers, and $T_1 = F_2 = 1$ and $T_2 = F_3 = 2$, we may conclude that $T_n = F_{n+1}$.

Exercise 4.2. *Use figure 2 to explain how the number of ancestors on the X chromosome inheritance line is related to Fibonacci numbers.*

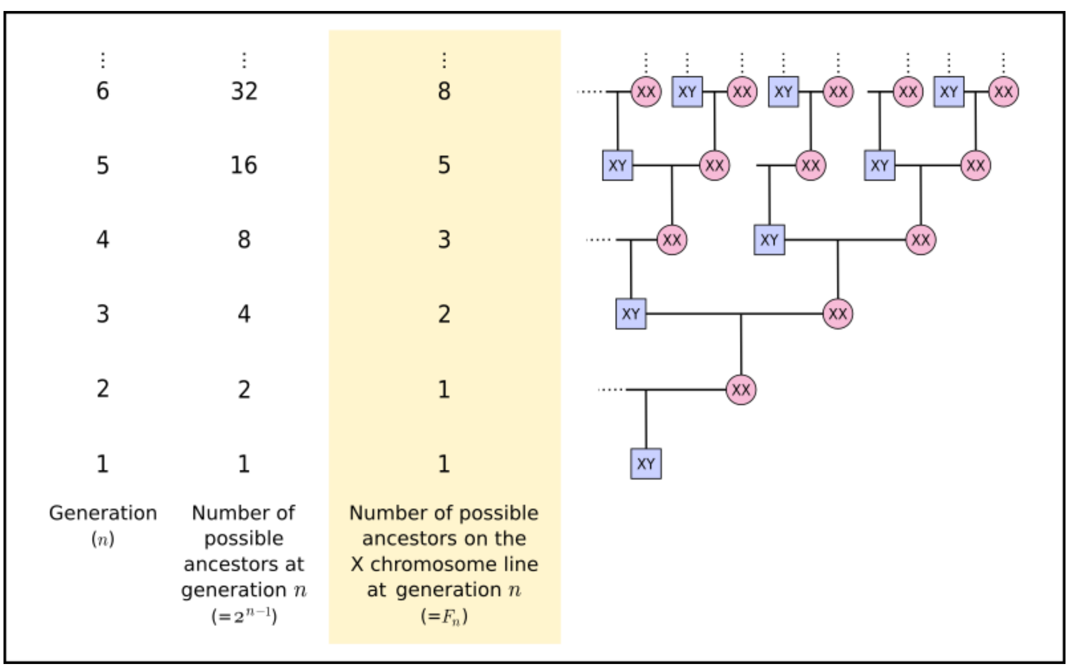

Figure 2: X chromosomes

Exercise 4.3 (from section 1.4 in [Gui18])**.** *A **partition** of a set S is a collection of non-empty subsets $A_i \subseteq S$, $1 \le i \le k$ (the **parts** of the partition), such that $\bigcup_{i=1}^k A_i = S$ and for every $i \neq j$, $A_i \cap A_j = \emptyset$.*

*The number of partitions of an n-element set is denoted B_n and is called the n-th **Bell number**. Show that the Bell numbers satisfy the recurrence*

$$B_{n+1} = \sum_{k=0}^{n} \binom{n}{k} B_k.$$

5 Induction (Chapter 3.6 in [KT17]

Many of you have probably seen mathematical induction before, but we review it here. We already proved combinatorially that

$$\sum_{i=1}^{n} i = \binom{n+1}{2}, \tag{3}$$

but let us forget that for a moment. Let us refer to (3) as **statement** S_n. That is, for example, S_1 is the following statement:

$$\sum_{i=1}^{1} i = \binom{2}{2}$$

which is true. The idea of mathematical induction is to infer the truth of S_n from the truth of the statements S_k for $k \le n$. How do we do this? We relate a statement S_n to earlier statements. For example, if we know that S_n is true, we know that

$$1 + 2 + \ldots + (n-1) + n = \frac{n(n+1)}{2} \tag{4}$$

is true, and we want to prove that

$$1 + 2 + \ldots + n + (n+1) = \frac{(n+1)(n+2)}{2} \tag{5}$$

is true. Notice how similar the two statements look. With a bit of algebra, starting from (4), we get

$$\begin{aligned} 1 + 2 + \ldots + (n-1) + n &= \frac{n(n+1)}{2} \\ 1 + 2 + \ldots + (n-1) + n + (n+1) &= \frac{n(n+1)}{2} + (n+1) \\ &= \frac{n(n+1) + 2(n+1)}{2} \\ &= \frac{(n+1)(n+2)}{2} \end{aligned}$$

and this is exactly S_{n+1}. So *assuming* S_n is true, we may conclude that S_{n+1} is true. We already checked that S_1 was true, which implies that S_2 is true, which implies that S_3 is true and so on. . . .

The **principle of mathematical induction** says that if there is such a sequence of statements S_n, and we can demonstrate that

- S_1 is true
- For each positive integer k, assuming that S_j is true for all $0 \le j \le k$ implies that S_k is true

then we may conclude that S_n is true for every positive integer. This is something that requires proof, and it relies on the **Well ordered property of positive integers**, that says that every non-empty set of positive integers has a minimal element. This is not very important for us, but if you are interested see Appendix A of [KT17]

Exercise 5.1. *Does every non-empty set of positive integers have a maximal element?*

How does induction work? We have a **base step** where we check a small case, or maybe several small cases, of the statement (S_0) we are interested in. This is followed by assuming that the statement is true for all k such that $k \leq n$ (or sometimes just $k = n$), this is called the **inductive hypothesis** and is commonly shortened to "IH" in proofs. Proving that S_k for $k \leq n$ being true implies that S_{n+1} is true is the **inductive step**, and this completes a proof by induction.

Our textbook distinguishes between two types of induction:

- Ordinary induction, where to show that S_{n+1} is true, we only need to know that S_n is true.
- **Strong inducion**, where in order to show that S_{n+1} is true we need to know that S_k is true for possibly all k such that $k \leq n$.

The underlying principle between both kinds of induction is the same. However, it is important to recognize what language we should be using when writing proofs by induction.

Example 5.2 (Example 6.13 in [Mor17])**.** *Let us define a sequence by the rule $a_1 = 2$ and for every integer $n \geq 2$, let*

$$a_n = \sum_{i=1}^{n-1} a_i.$$

Prove by induction that for every $n \geq 2$, we have $a_n = 2^{n-1}$.

The base case $a_2 = a_1 = 2$ is clear. If we assume that $a_n = 2^{n-1}$, and express

$$a_{n+1} = \sum_{i=1}^{n} a_i$$

since we assumed that $a_n = 2^{n-1}$, and this leads to

$$a_{n+1} = \sum_{i=1}^{n-1} a_i + 2^{n-1}.$$

But here we are stuck, as we do not know what to do with the other a_is. What we should do instead is to assume that $a_k = 2^{k-1}$ for all $k \leq n$. Then when we get to the induction step we can replace all the a_is with 2^{i-1} to get

$$a_{n+1} = 2 + \sum_{i=2}^{n} 2^{i-1}$$

and now this is a sum of a geometric sequence, in particular

$$\sum_{i=2}^{n} 2^{i-1} = \left(\sum_{i=0}^{n-1} 2^i \right) - 1 = (2^n - 1) - 1$$

so altogether we have

$$a_{n+1} = 2^n,$$

and we are done by induction.

This does not seem like a big difference, but when you write a proof, it is important to always check that you made the right assumptions (and that you *can* make those assumptions).

Exercise 5.3. *Find the mistake in the following famous proof that all horses are the same color:*

We will prove that all horses are the same color by induction. Let S_n be the statement:

Any set of n horses have the same color.

The base case S_1 is clearly true, as any horse is the same color as itself. Assume for induction that S_{n-1} is true. Consider a set of n horses and number them. By IH, the horses numbered $\{1, 2, \ldots, n-1\}$ are all the same color. Similarly, the horses numbered $\{2, 3, \ldots, n-1, n\}$ are also all the same color. But since

$$\{1, 2, \ldots, n-1\} \cap \{2, 3, \ldots, n\} = \{2, 3, \ldots, n-1\}$$

the two sets intersect, so all horses numbered $\{1, 2, \ldots, n\}$ are all the same color, hence S_n is true and by the principle of mathematical induction, all horses are the same color.

Proofs by induction are generally less satisfying and less enlightening than combinatorial proofs. On the plus side, doing things by induction is a relatively straightforward recipe that results in pefectly valid proof. Once you know something is true, you can still continue to think about combinatorial proofs. There are many statements that are combinatorial but have no known combinatorial proofs!

Induction can also help with some problems that may be inaccessible combinatorially

Exercise 5.4 (Example 6.16 in [Mor17]). *Prove by induction that the nth term of the Fibonacci sequence F_n is at least $\left(\frac{3}{2}\right)^{n-1}$ for every $n \geq 0$.*

Solution: We check the two base cases. When $n = 0$, we have

$$F_0 = 1 \geq \frac{2}{3} = \left(\frac{3}{2}\right)^{-1},$$

and when $n = 1$, we have

$$F_1 = 1 \geq 1 = \left(\frac{3}{2}\right)^{0}$$

so the base cases hold.

Let $n \geq 1$ and assume that for every integer k such that $1 \leq k < n$, $F_k \geq \left(\frac{3}{2}\right)^{k-1}$ (note that we are using strong induction here). We have

$$\begin{aligned}
F_n &= F_{n-1} + F_{n-2} && \text{by definition} \\
&\geq \left(\frac{3}{2}\right)^{n-2} + \left(\frac{3}{2}\right)^{n-3} && \text{by IH} \\
&= \left(\frac{3}{2}\right)^{n-3}\left(\frac{3}{2} + 1\right) \\
&= \frac{5}{2}\left(\frac{3}{2}\right)^{n-2} \\
&= \frac{5}{3}\frac{3}{2}\left(\frac{3}{2}\right)^{n-2} \\
&> \left(\frac{3}{2}\right)^{n-1}.
\end{aligned}$$

This completes the proof of the inductive step, and we are done by the principle of mathematical induction.

Exercise 5.5 (Exercise 6.17. 1) in [Mor17]). *Prove by induction that for every $n \geq 0$, the nth term of the Fibonacci sequence is no greater than 2^n.*

References

[Gui18] David Guichard. *Combinatorics and Graph Theory*. Open access, 2018. Available at https://www.whitman.edu/mathematics/cgt_online/book/. 4

[KT17] Mitchel T. Keller and William T. Trotter. *Applied Combinatorics*. Open access, 2017. Available at http://www.rellek.net/appcomb/. 1, 3, 4

[Mor17] Joy Morris. *Combinatorics*. Open access, 2017. Available at http://www.cs.uleth.ca/~morris/Combinatorics/Combinatorics.html. 5, 6