Investigating Young Tableaux in a Box

Emilie Hogan

July 13, 2008

For my Experimental Math project I investigated Young Tableaux that fit inside an a row by b column box. As a reminder, a Young Tableaux is an array of a certain shape, containing integers which are increasing along rows and columns. For example, the following is a Young Tableaux of shape $[4, 2]$ because there are 4 boxes in the first row and 2 boxes in the second row.

1	2	4	6
3	5		

I wrote some maple programs that would compute the number of Young Tableaux strictly inside an a by b box. When I say "strictly" I mean that they don't fit in any smaller box. So that means that the largest part of the shape is a and there are exactly b parts. the program for this is called `YTaxbStrict(a,b)`. I first tried `seq(YTaxbStrict(a,2),a=1..10)` and searched for this in Sloane and found A001453, the Catalan numbers minus 1. It appeared that $YTaxbStrict(a, 2) = Catalan(a + 1) - 1$. This intrigued me so I decided to find a bijection. This probably isn't new but I didn't look it up before I solved it.

1 The mapping

Let me first describe the problem in a bit more detail. For any $a \in \{1, 2, 3, \ldots\}$ I will produce a bijection from the set of all Young Tableaux of shapes $[a, 1], [a, 2], [a, 3], \ldots, [a, a]$ to the set of lattice paths from $(0, 0)$ to $(a+1, a+1)$ consisting only of up steps $\uparrow$ and right steps $\rightarrow$ with the exception of the path of all up steps followed by all right steps. For $a = 2$ the set of Young Tableaux is:

$$\begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 & 4 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & 4 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 & \\ \cline{1-1} \end{array} \quad \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & \\ \cline{1-1} \end{array}$$

Instead of using lattice paths I will use vectors of length $2(a+1)$ consisting of $a+1$ 1s and $a+1$ -1s in which no partial sum is less than 0, and the final sum is equal to 0. These have an easy bijection with lattice paths and are easier to illustrate (I don't know how to draw lattice paths in Latex). The easy bijection is 1 maps to $\uparrow$ and -1 maps to $\rightarrow$. The no partial sum < 0 criteria assures that the lattice path won't go below the $y = x$ line. The final sum equalling 0 assures that the path goes from $(0, 0)$ to $(a + 1, a + 1)$. The path of all up arrows followed by all right arrows corresponds to the vector $[1, \ldots, 1, -1, \ldots, -1]$ of $a + 1$ plus ones followed by $a + 1$ minus ones. The set for $a = 2$ is:

$$\begin{gathered}[1, 1, -1, 1, -1, -1] \\ [1, -1, 1, 1, -1, -1] \\ [1, 1, -1, -1, 1, -1] \\ [1, -1, 1, -1, 1, -1]\end{gathered}$$

So now I will describe the mapping, φ, and show that it is one-to-one and onto. Let YT be a 2-row, a-column Young Tableaux containing the numbers $1, 2, \ldots, k$ with $k \leq 2a$. Then applying φ does the following:

- If the number i is in the top row then put $+1$ in the i^{th} spot of the vector.
- If the number i is in the bottom row then put -1 in the i^{th} spot of the vector.
- The vector is now of length $k \leq 2a$. Now put $+1$ in the $k + 1$ spot followed by enough -1's to fill up the rest of the $2(a + 1)$ spots in the vector.

For example, the mapping for $a = 2$ is:

$$\begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 & 4 \\ \hline \end{array} \longrightarrow [1, 1, -1, -1, 1, -1]$$

$$\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & 4 \\ \hline \end{array} \longrightarrow [1, -1, 1, -1, 1, -1]$$

$$\begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 & \multicolumn{1}{c}{} \\ \cline{1-1}\end{array} \longrightarrow [1,1,-1,1,-1,-1]$$

$$\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & \multicolumn{1}{c}{} \\ \cline{1-1}\end{array} \longrightarrow [1,-1,1,1,-1,-1]$$

Now I will show that this mapping is one-to-one and onto.

One-to-one: Let γ_1 and γ_2 be two different Young tableaux. Then I need to show that $\varphi(\gamma_1) \neq \varphi(\gamma_2)$.

Case 1 - The two Young tableaux are of the same shape. Then their images must be different because there is at least one i for which i is in the first row in γ_1 and in the second row for γ_2. Then $\varphi(\gamma_1)[i] = 1$ and $\varphi(\gamma_2)[i] = -1$. So we have the desired $\varphi(\gamma_1) \neq \varphi(\gamma_2)$.

Case 2 - The two Young tableaux are of different shapes. Then their images must be different because of the placement of the $a+1^{st}$ plus one. In any $\varphi(\gamma)$ the $a+1^{st}$ plus one comes in the $a + nops(\gamma[2]) + 1$ spot. So since γ_1 and γ_2 are different shapes then the $a+1^{st}$ plus ones must be in different spots in their images.

Onto: Notice that the vector of $a+1$ plus ones followed by $a+1$ minus ones is not in the image of φ. This is because everything before the $a+1^{st}$ plus one is in the Young tableaux, and there must be at least one number in the second row. So the $a+1^{st}$ plus one must come after the first minus one, which is not the case in the aforementioned vector. So now, given a vector v, I will produce a γ such that $\varphi(\gamma) = v$. If I can do that then I will have shown φ to be onto. Let w be the vector contained in v consisting of everything before the $a+1^{st}$ plus one. The number of entries in the top row of γ is the number of $+1$ in w which must equal a. The number of entries in the bottom row of γ is the number of -1 in w, this must be $\leq a$ since the sum of w is ≥ 0 and there are a plus ones. The first entry in w must be $+1$ because the partial sums must not be less than 0. This corresponds to placing a 1 in the first box in the top row. Say we've placed l numbers and it satisfies the Young Tableaux criteria (increasing by row and column and the number of entries in the top row is greater than the number of entries in the bottom row). In the $l+1$ spot there is either a $+1$

or a -1. If at the l step the number of elements in the top row equals the number in the bottom row then the $l+1$ entry must be $+1$, so we place $l+1$ at the end of the top row of γ. This preserves the Young Tableaux criteria because $l+1$ is the largest thing we've placed so far. If, on the other hand, the number of elements in the bottom row is less than in the top row we can either have a $+1$ or a -1 in spot $l+1$ of w. In either case, $l+1$ is the largest entry so far so wherever we put it won't disrupt the Young Tableaux criteria of increasing along rows and columns. When we've exhausted the length of w we have a Young Tableaux with a elements in the top row and $\leq a$ elements in the bottom row. Clearly this is unique because any change in w changes the location of an integer in γ. So for any vector, v, of length $2(a+1)$ of $+1$ and -1 there is a Young Tableaux, γ, with a elements in the top row and $\leq a$ elements in the second row such that $\varphi(\gamma) = v$. Thus φ is onto.

So I have shown that φ is a one-to-one and onto function between the two domains of interest. Therefore we have a bijection and the desired result

$$YTaxbStrict(a, 2) = Cat(a+1) - 1$$

2 Possible Higher Dimensional Generalizations

I thought that this could generalize to higher dimensional lattice paths with some criteria but have yet to formalize it. My first thought was to go to three dimensions since it's still possible to visualize. So I computed the sequence

$$\texttt{seq(YTaxbStrict(a,3),a = 1..7) = 1,13,148,1809,24529,365184,5862451}$$

But this sequence is not yet in Sloane. This would be the number of Young Tableaux with largest part exactly a and exactly 3 parts. My thought was that this could count the number of 3-dimensional lattice paths from $(0,0,0)$ to $(a+1, a+1, a+1)$ staying in the region $\{(x,y,z) : x \geq y \geq z\}$. The mapping from the set of these Young Tableaux to a vector of length $3(a+1)$ consisting of x's, y's, z's in which the $\#x \geq \#y \geq \#z$ could be:

- If i is in the first row then place an x in the i^{th} spot.
- If i is in the second row then place a y in the i^{th} spot.

- If i is in the third row then place a z in the i^{th} spot.
- After the Young Tableaux is exhausted place an x then $(a+1) - nops(YT[2])$ y's, then $(a+1) - nops(YT[3])$ z's in the vector.

Then to turn the vector into a lattice path start at $(0,0,0)$ and move one step in the positive x direction each time there is an x in the vector, move one step in the positive y direction each time there is a y in the vector, and move one step in the positive z direction each time there is a z in the vector.

3 Maple Package

I wrote a bunch of Maple programs and took some from the work we did in class. Here is a list of all the programs in the accompanying package with descriptions:

Mamas(L): Inputs a partition shape L and outputs all shapes obtained from L by legally taking away one box from each row. Legally means that you can't remove a box if there is a box underneath it. For example $Mamas([5,4,4,3]) = \{[4,4,4,3],[5,4,3,3],[5,4,4,2]\}$. Notice $[5,3,4,3]$ is not included because that is not a legal partition.

YT(L): Inputs a partition shape L and outputs the number of Young Tableaux of that shape using the 'going down formula'

$$YT(L) = \sum_{M \in Mamas(L)} YT(M)$$

YTixi(i): Inputs an integer i and outputs the number of Young Tableaux fitting inside an $i \times i$ box. In other words, the number of Young Tableaux whose shape has at most i parts and largest part is less than or equal to i.

YTaxb(a,b): Inputs two integers a and b and outputs the number of Young Tableaux whose shape has at most b parts and whose largest part is less than or equal to a.

YTaxbStrict(a,b): Inputs two integers a and b and outputs the number of Young Tableaux whose shape fits strictly inside an a column, b row box (it fits in no smaller box). In other words, the shape of the Young Tableaux has largest part exactly a and number of parts exactly b.

Cat(n): The n^{th} Catalan number.

Phi1(YT): Inputs a Young Tableaux with 2 rows and outputs the vector of ± 1 as described in Section 1. It's called φ in that section.

Phi2(vec): Inputs a vector of ± 1 and outputs the corresponding lattice path.

Phi(YT): The composition Phi2(Phi1(YT)).

Phi1Inv(vec): The inverse of Phi1. Inputs a vector of ± 1 and outputs the corresponding Young Tableaux.

Par(n): Inputs an integer n and outputs the set of partitions of n.

Par1(n,k): Inputs a pair of integers n and k and outputs the set of partitions of n with largest part equal to k.

ParBox(n,i,j): Inputs three integers n, i, j and outputs the set of partitions of n whose largest part is $\leq i$ and has $\leq j$ parts.

ParBox1(n,k,l): Inputs three integers n, k, l and outputs the set of partitions of n whose largest part is k and has exactly l parts.

F(L): Inputs a partition shape L and outputs the set of Young Tableaux of shape L.

Fset(shapes): Inputs a set of partitions *shapes* and outputs the set of Young Tableaux of all partitions in *shapes*.

AllixiParts(i): Inputs an integer i and outputs the set of all partitions fitting inside an $i \times i$ box as described above.

AllaxbParts(a,b): Inputs two integers a, b and outputs the set of all partitions that fit inside an a (columns) by b (rows) box as described above.

AllaxbPartsStrict(a,b): Inputs two integers a, b and outputs the set of all partitions that fit strictly inside an a (columns) by b (rows) box and don't fit in any smaller box.

GuessRat(L,x): Inputs a list L and a variable x and outputs a rational function $f(x)$ such that $f(n) = L[n]$.

GuessRat1(L,x,d): Inputs a list L, a variable x, and an integer d and outputs a rational function $f(x)$ where the degree of the numerator and degree of the denominator are each less than d such that $f(n) = L[n]$.

GuessPol(L,n): Inputs a list L and a variable n and outputs a polynomial $f(n)$ such that $f(i) = L[i]$.

GuessPol(L,d,n): Inputs a list L, a variable n, and an integer d and outputs a polynomial $f(n)$ with degree $\leq d$ such that $f(i) = L[i]$.