4/1

The Robinson-Schensted Algorithm

Recall f^λ = # SYTs of sh. λ. We will prove following identity:

Thm $\sum_{\lambda \vdash n} (f^\lambda)^2 = n!$ for all $n \geq 1$.

e.g. $n=2 \Rightarrow (f^{\square\square})^2 + (f^{\begin{smallmatrix}\square\\\square\end{smallmatrix}})^2 = 1^2 + 1^2 = 2!$ ✓

$n=3 \Rightarrow (f^{\square\square\square})^2 + (f^{\begin{smallmatrix}\square\square\\\square\end{smallmatrix}})^2 + (f^{\begin{smallmatrix}\square\\\square\\\square\end{smallmatrix}})^2 = 1^2 + 2^2 + 1^2 = 6 = 3!$ ✓

May seem like a strange formula, but has algebraic meaning. We will explain a bijective pf. of this thm., using a very important procedure called Robinson-Schensted Algorithm.

Observe that

$$\sum_{\lambda \vdash n} (f^\lambda)^2 = \#\left\{ \text{pairs } (P,Q) \text{ of SYTs w/ } n \text{ boxes, s.t. } sh(P) = sh(Q) \right\}$$

and of course

$n!$ = # permutations in S_n

so the thm. would follow from a bijection

$$S_n \longrightarrow \{ \text{pairs } (P,Q) \text{ of SYTs w/ } sh(P) = sh(Q) \vdash n \}$$

The Robinson-Schensted Algorithm is such a bijection.

The main "loop" of the RS algorithm involves insertion: we have a tableau and we want to put a new # in it.

e.g.

1	3	6	9
2	8	10	
4			
7			

← 5 insert

Note! the "tableau" here is like an SYT in that it's increase down rows/cols, but #s are not $1, 2, \ldots, n$ That's ok.

How do we carry out the insertion? (call it i) Well we start by trying to put the # we're inserting in the top row:

- if i is bigger than all #'s in 1st row, put it at end, call it j
- otherwise, put i where smallest # bigger than i is, and bump this j by inserting it into the next row.

e.g. $1\ 3\ 6\ 9 \leftarrow 5$ (5 bumps 6) $\rightsquigarrow 1\ 3\ 5\ 9$ (6 bumps 8) $\leftarrow 6 \rightsquigarrow 1\ 3\ 5\ 9$ $2\ 8\ 10 \rightsquigarrow 2\ 8\ 10 \leftarrow 6 \rightsquigarrow 2\ 6\ 10$ $4\ 7 \rightsquigarrow 4\ 7 \leftarrow 8 \rightsquigarrow 4\ 8$ 7

As depicted above, we keep doing the same procedure of bumping until we reach a row where the # we're inserting is biggest. The result is the insertion of the # into the tableau, and Exercise: it produces a new tableau.

The RS algorithm is built out of these insertions. We start with permutation $\sigma = \begin{pmatrix} 1 & 2 & 3 & \cdots & n \\ \sigma_1 & \sigma_2 & \sigma_3 & \cdots & \sigma_n \end{pmatrix} \in S_n$. We want to produce two SYTs, P and Q, of the same shape. The tableau P is called the insertion tableau and is the result of inserting σ_1, then σ_2, then σ_3, ... (starting from $\emptyset$ empty tableau)

e.g. $\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 5 & 2 & 3 & 6 & 4 & 1 & 7 \end{pmatrix}$

$\emptyset$, $\boxed{5}$, $\begin{array}{|c|}\hline 2 \\ \hline 5 \\ \hline \end{array}$, $\begin{array}{|c|c|}\hline 2 & 3 \\ \hline 5 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|}\hline 2 & 3 \\ \hline 5 & 6 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|}\hline 2 & 3 \\ \hline 4 & 6 \\ \cline{1-1} 5 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 4 & 6 \\ \cline{1-1} 5 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|c|}\hline 1 & 3 & 4 \\ \hline 2 & 6 \\ \cline{1-2} 5 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|c|c|}\hline 1 & 3 & 4 & 7 \\ \hline 2 & 6 \\ \cline{1-2} 5 \\ \cline{1-1} \end{array} = P$

Meanwhile, Q is the recording tableau and keeps track of the order in which boxes were added in insertion process:

e.g. $\emptyset$, $\boxed{1}$, $\begin{array}{|c|}\hline 1 \\ \hline 2 \\ \hline \end{array}$, $\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & 4 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & 4 \\ \cline{1-1} 5 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & 4 \\ \cline{1-1} 5 \\ \cline{1-1} 6 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|c|}\hline 1 & 3 & 4 \\ \hline 2 & 5 \\ \cline{1-2} 6 \\ \cline{1-1} \end{array}$, $\begin{array}{|c|c|c|c|}\hline 1 & 3 & 4 & 7 \\ \hline 2 & 5 \\ \cline{1-2} 6 \\ \cline{1-1} \end{array} = Q$

4/4

By construction, P and Q have the same shape. So we get a map $S_n \xrightarrow{RS} \{(P,Q) : sh(P) = sh(Q) \vdash n\}$

Thm This map $\sigma \xmapsto{RS} (P,Q)$ is a bijection.

Pf: As w/ Hillman-Grassl, goal is to show we can locally undo steps. In other words, we can describe inverse $(P,Q) \xmapsto{RS^{-1}} \sigma$. Here is how that works. Suppose we are given (P,Q):

e.g. $P = \begin{array}{ccc} 1 & 3 & 4 \\ 2 & 6 & \\ 5 & & \end{array}$ $\quad Q = \begin{array}{ccc} 1 & 3 & 4 \\ 2 & 5 & \\ 6 & & \end{array}$

The location of the biggest #, n, in Q tells us the # in P that was the termination of the last bumping sequence. Then we can "reverse bump/insert" this entry out of P:

- if it is in row 1, simply remove it,
- otherwise, have it replace the ~~smallest~~ (largest) # less than it in the row above, and bump that # out and repeat.

e.g. $\begin{array}{ccc} 1 & 3 & 4 \\ 2 & 6 & \end{array}$ 5 bumps 2 $\begin{array}{ccc} 1 & 3 & 4 \\ \boxed{2} & 5 & 6 \end{array}$ 2 bumps 1 $\begin{array}{ccc} 2 & 3 & 4 \\ 5 & 6 & \end{array}$ $\boxed{1}$ → removed

rev. insert this → $\boxed{5}$

Then write down $\sigma_n :=$ # removed from rev. insertion. And remove n from Q, and repeat same steps but now with box containing n-1 in Q. In this way, we build up sequence $\sigma_n, \sigma_{n-1}, \ldots, \sigma_1$ and then $\sigma = \sigma_1 \sigma_2 \cdots \sigma_n$ is our desired permutation.

It is easy to see that this is the inverse, b.c. reverse insertion "locally" inverses insertion (again there are a few things to check... I leave to you as Exercise). ∎

The Robinson-Schensted-Knuth Algorithm

The RSK algorithm is an extension of RS alg. to semistandard (as opposed to standard) tableaux. Again we have a motivational formula:

Thm (Cauchy identity)

$$\sum_{\lambda} s_\lambda(x_1, x_2, \ldots)\, s_\lambda(y_1, y_2, \ldots) = \prod_{i,j \geq 1} \frac{1}{(1 - x_i y_j)}$$

Before we start the proof, a few remarks about this identity:

- the sum is over all partitions λ (of all sizes)!
- there are two infinite sets of variables $\vec{x} = \{x_1, x_2, \ldots\}$ and $\vec{y} = \{y_1, y_2, \ldots\}$
- it is an identity in $\mathbb{C}[[x_1, x_2, \ldots, y_1, y_2, \ldots]]$

The Cauchy identity is again very important result in sym. fn. theory.

By standard limit argument we've seen before, it suffices to prove a "finite" version for all $n \geq 1$:

$$\sum_{\lambda : \ell(\lambda) \leq n} s_\lambda(x_1, x_2, \ldots, x_n)\, s_\lambda(y_1, y_2, \ldots, y_n) = \prod_{i,j=1}^{n} \frac{1}{1 - x_i y_j}$$

here we are only finitely many variables.

We want to give a bijective pf. of Cauchy identity so let's interpret the coefficient of $\vec{x}^{\alpha}\vec{y}^{\beta}$ for $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, $\beta = (\beta_1, \beta_2, \ldots, \beta_n)$ on LHS + RHS.

On LHS,

$$\text{coeff. of } \vec{x}^{\alpha}\vec{y}^{\beta} = \#\{(P, Q) : P \text{ and } Q \text{ are SSYT w/ } sh(P) = sh(Q) \text{ and } con(P) = \alpha \text{ and } con(Q) = \beta\}$$

matrices w/ entries in $\mathbb{N} = \{0,1,2,\dots\}$

What about RHS? For this we will use $n \times n$ $\mathbb{N}$-matrices M:

- for $M=(m_{ij})$, let $\text{row}_i(M) = \sum_j m_{ij}$, be sum of i^{th} row, and let $\text{col}_j(M) = \sum_i m_{ij}$ be sum of j^{th} col.

Prop. Coeff. of $\vec{x}^{\alpha}\vec{y}^{\beta}$ in $\prod_{i,j \geq 1} \frac{1}{1-x_i y_j} = \#\left\{ n\times n \ \mathbb{N}\text{-matrices } M \text{ w/ } \beta = (\text{row}_1(M), \text{row}_2(M), \dots), \ \alpha = (\text{col}_1(M), \text{col}_2(M), \dots) \right\}$

Pf: Associate $M=(m_{ij})$ to choice of $(1 + x_i y_j + (x_i y_j)^2 + \cdots + (x_i y_j)^{m_{ij}} + \cdots$ that term when expanding the product. ☒

4/6 Hence the Cauchy identity will follow from the existence of a bij.

$$\left\{ n\times n \ \mathbb{N}\text{-matrices } M \right\} \to \left\{ (P,Q) : \text{SSYTs w/ } sh(P) = sh(Q) = \text{[illegible]} \right\}$$

s.t. $\text{con}(Q) = (\text{row}_1(M), \text{row}_2(M), \dots)$, $\text{con}(P) = (\text{col}_1(M), \text{col}_2(M), \dots)$ when $M \mapsto (P,Q)$. The Robinson-Schensted-Knuth algorithm is this bijection, → RSK for short

Let's first explain how this is a generalization of RS. The idea is that we encode a permutation $\sigma \in S_n$ by its permutation matrix:

$$\sigma = 213 \longrightarrow \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The $n\times n$ $\mathbb{N}$-matrices w/ row/col sums all $= 1$ are exactly the permutation matrices, and similarly, the SSYTs w/ content $= (1,1,\dots,1)$ are exactly the standard tableaux.

RSK applied to a perm. matrix will be RS.

In fact, RSK is only a very slight extension of RS, once we have the correct set-up. . . .

The first thing we have to clarify is how to insert into a SSYT. Now we use the rules: to insert $T \leftarrow i$

- if i is not less than any # in 1st row, put it there,
- otherwise, find leftmost entry it is less than, bump that entry j into the next row, and repeat.

e.g. $\begin{matrix}1&2&2\\3&3\end{matrix} \leftarrow 1 \overset{\text{bump } 2}{\rightsquigarrow} \begin{matrix}1&1&2\\3&3\end{matrix} \leftarrow 2 \overset{\text{bump } 3}{\rightsquigarrow} \begin{matrix}1&1&2\\2&3\\3\end{matrix}$

Next, we need to explain what sequence of #'s we are inserting. Given matrix M, form biarray $\begin{pmatrix} a_1 & a_2 & \cdots & a_k \\ b_1 & b_2 & & b_k \end{pmatrix}$ that has $m_{i,j}$ copies of $\begin{smallmatrix} i \\ j \end{smallmatrix}$ and is s.t.

- $a_1 \le \cdots \le a_k$
- $b_i \le b_j$ if $i \le j$ and $a_i = a_j$

e.g. $M = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 1 & 1 & 0 \end{pmatrix} \rightsquigarrow$ biarray $\begin{pmatrix} 1&1&1&2&2&3&3 \\ 1&3&3&2&2&1&2 \end{pmatrix}$, $m_{1,3}=2$

Then we insert the sequence $b_1, b_2, \ldots, b_k$ to form P:

$\emptyset \leftarrow 1, \boxed{1} \leftarrow 3, \boxed{1}\boxed{3} \leftarrow 3, \begin{matrix}1&3&3\\3\end{matrix} \leftarrow 2, \begin{matrix}1&2&3\\3\end{matrix} \leftarrow 2, \begin{matrix}1&2&2\\3&3\end{matrix} \leftarrow 1, \begin{matrix}1&1&2\\2&3\\3\end{matrix} \leftarrow 2$

$P = \begin{matrix}1&1&2&2\\2&3\\3\end{matrix}$

And what about Q? again it records order new boxes were added to P, but now the entries we add to Q are $a_1, a_2, \ldots, a_k$:

$\emptyset, \boxed{1}, \boxed{1}\boxed{1}, \begin{matrix}1&1&1\\2\end{matrix}, \begin{matrix}1&1&1\\2&2\end{matrix}, \begin{matrix}1&1&1\\2&2\\3\end{matrix}, \begin{matrix}1&1&3\\2&2\\3\end{matrix} = Q$

The map $M \xrightarrow{RSK} (P, Q)$ is the RSK algorithm, and it is a straightforward ext. of our arguments about RS to show that it has the desired properties (e.g., is a bijection). ∎

4/7

Another construction of RSK via toggles

We will now give a very different description of RSK, which will reveal some hidden symmetries of the algorithm. This does not appear in Sagan. Instead you can read

samuelfhopkins.com/docs/rsk.pdf

To start, we want to encode SSYTs in a different way.

DEF'N A Gelfand-Tsetlin pattern of size n is a triangular array

$$\begin{array}{ccccccccc} g_{1,1} & & g_{1,2} & & g_{1,3} & \cdots & & & g_{1,n} \\ & \searrow & & \searrow & & & & \searrow & \\ & & g_{2,2} & & g_{2,3} & & \cdots & & g_{2,n} \\ & & & \searrow & & \searrow & & & \\ & & & & g_{3,3} & & \cdots & & g_{3,n} \\ & & & & & \searrow & & & \\ & & & & & & g_{n,n} & & \end{array}$$

of nonnegative integers $g_{i,j} \in \mathbb{N}$ for $1 \le i \le j \le n$

such that $g_{i,j} \ge g_{i+1,j+1} \ge g_{i,j+1}$ $\forall\, i, j$.

There is a bijection

$$\{\text{SSYTs with entries in } \{1,2,\dots n\}\} \longrightarrow \{\text{GT patterns of size } n\}$$

$$T \longmapsto GT(T) = (g_{i,j})$$

where $(g_{i,i}, g_{i,i+1}, \dots g_{i,n}) = sh(T \text{ restricted to entries } \{1, 2, \dots, n+1-i\})$.

e.g. $P = \begin{matrix} 1 & 1 & 2 & 2 \\ 2 & 3 \\ 3 \end{matrix}$ SSYT w/ entries $\le [3]$ $\longrightarrow GT(P) = \begin{matrix} & & 4 & & 2 & 1 \\ & 4 & & 1 \\ & & 2 \end{matrix}$

since $sh\left(\begin{matrix} 1 & 1 & 2 & 2 \\ 2 & 3 \\ 3 \end{matrix}\right) = (4,2,1)$, $sh\left(\begin{matrix} 1 & 1 & 2 & 2 \\ 2 \end{matrix}\right) = (4,1)$, $sh\left(\begin{matrix} 1 \\ 1 \end{matrix}\right) = (2)$

Exercise: Prove this really is a bijection.

Recall RSK is a bijection

$$M, \text{ } n \times n \text{ } \mathbb{N}\text{-matrix} \xrightarrow{RSK} (P, Q)$$

pair of SSYT w/ $sh(P) = sh(Q)$ and entries $\le [n]$

e.g. $M = \begin{pmatrix} 1&0&2 \\ 0&2&0 \\ 1&1&0 \end{pmatrix} \overset{RSK}{\mapsto} (P,Q) = \left(\begin{matrix} 1&1&2&2 \\ 2&3 \\ 3 \end{matrix}, \begin{matrix} 1&1&1&3 \\ 2&2 \\ 3 \end{matrix} \right)$ which have

$GT(P) = \begin{matrix} 4&2&1 \\ 2&1 \\ 2 \end{matrix}$ $GT(Q) = \begin{matrix} 4&2&1 \\ 3&2 \\ 3 \end{matrix}$

Notice that since $ch(P) = ch(Q)$, 1st rows of $GT(P)$, $GT(Q)$ are same.
So we can glue $GT(P)$ and $GT(Q)$ into a matrix;

e.g. GT(Q) / GT(P) $= \begin{pmatrix} 1&2&3 \\ 1&2&3 \\ 2&4&4 \end{pmatrix} \in VI$ ← observe: inequalities on GT-pattern become: weakly increasing along rows + cols!
$\geq$

In other words, we can view RSK as a bijection:

$\{ n \times n \ \mathbb{N}\text{-matrices } M \} \xrightarrow{RSK} \{ \text{reverse plane partitions } \pi \text{ of shape } n \times n \}$.

But what properties does this bij. satisfy?
Recall that $con(P) = col(M)$, and notice that

Sum of i^{th} row (from bottom) of $GT(P)$ = # entries in $\{1,2,\dots,i\}$ in P
" i^{th} "lower diagonal" of π " sum of 1st i columns of M.

e.g. $\pi = \begin{pmatrix} 1&2&3 \\ 1&2&3 \\ 2&4&4 \end{pmatrix}$ $M = \begin{pmatrix} 1&0&2 \\ 0&2&0 \\ 1&1&0 \end{pmatrix}$

2nd diag. → $4+1 = 1+2+1+1$ ← two columns in M

Similarly, the i^{th} upper diagonal sum of π
= sum of 1st i rows of M.

e.g. $\pi = \begin{pmatrix} 1&2&3 \\ 1&2&3 \\ 2&4&4 \end{pmatrix}$ $M = \begin{pmatrix} 1&0&2 \\ 0&2&0 \\ 1&1&0 \end{pmatrix}$

1st diag → $3 = 1+2$ ← one row in M.

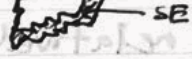
4/11

We will give another construction of this map $M \xrightarrow{RSK} \pi$ which converts row/col sums to diagonal sums. Actually, we will define an even more general bijection.

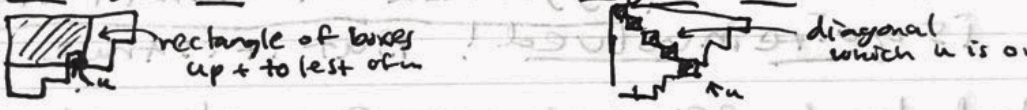
Thm for any partition shape λ $\exists$ bijection

$\{$ N-fillings of λ $M\} \xrightarrow{RSK} \{$ rev. plane partitions π ^{sh} $^{wt}= \lambda\}$

s.t. $\forall$ boxes u on SE ribbon boundary:

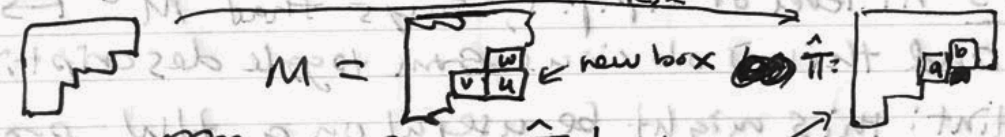


- rectangle sum of M at u = diagonal sum of π at u:



e.g. $\lambda = (2,1)$ $\begin{array}{|c|c|}\hline a & b \\ \hline c \\ \cline{1-1}\end{array} \xmapsto{RSK} \begin{array}{|c|c|}\hline a & a+b \\ \hline a+c \\ \cline{1-1}\end{array}$

We define RSK recursively. Suppose $\hat{M} \xmapsto{RSK} \hat{\pi}$ for $\hat{\lambda}$, shape obtained from λ by removing single box:



Then define π from $\hat{\pi}$ by:

- toggling all boxes in diagonal of the new box
- filling the new box w/ $\max(a,b) + u$

Here toggling an entry of an r.p.p. does:

$$u \begin{array}{|c|}\hline x \\ \hline z \\ \cline{1-1}\end{array}^{w} y \mapsto u \begin{array}{|c|}\hline x' \\ \hline z \\ \cline{1-1}\end{array}^{w} y \quad \text{where} \quad \boxed{x' = \max(u,w) + \min(y,z) - x}$$

Exercise: toggling maintains order for r.p.p.

Then we define $M \xmapsto{RSK} \pi$.

e.g.

$\begin{array}{|c|c|}\hline a & b \\ \hline c \\ \cline{1-1}\end{array} \xmapsto{RSK} \begin{array}{|c|c|}\hline a & a+b \\ \hline a+c \\ \cline{1-1}\end{array}$ so $\begin{array}{|c|c|}\hline a & b \\ \hline c & d \\ \cline{1-1}\end{array} \xmapsto{RSK}$

$\min(b,c)$	$a+b$
$a+c$	$d+\max(b,c)+a$

Exercise: Check $M = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 1 & 1 & 0 \end{pmatrix} \xrightarrow{RSK} \pi = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 2 & 4 & 4 \end{pmatrix}$ using toggles!

To show that the toggle definition of RSK:
- doesn't depend on order we remove boxes,
- is a bijection,
- converts rectangle sums to diagonal sums

is relatively easy via induction. See my write-up.
To show that toggle RSK = insertion RSK is quite involved! But it is true...

And toggle RSK makes one symmetry clear:

Thm If $M \xrightarrow{RSK} (P, Q)$ then $M^t \xrightarrow{RSK} (Q, P)$.
(transpose)

Pf: At level of r.p.p.'s, says that $M^t \xrightarrow{RSK} \pi^t$, and this is obvious from toggle description! ☺

Hint: this might be useful on a HW problem...

One final observation is that if $M = (m_{i,j}) \xrightarrow{RSK} \pi$ then $\sum_{(i,j) = u \in \lambda} h(u) \cdot m_{i,j} = |\pi| = \sum \pi_{i,j}$.

Exercise: Prove from the properties about rectangle and diagonal sums.

So... this "toggle RSK" gives another pf. of:

$$\sum_{\pi \in RPP(\lambda)} q^{|\pi|} = \prod_{u \in \lambda} \frac{1}{1 - q^{h(u)}}$$

But it is not the same bijection as Hillman-Grassl! //