

4/1

5th May

The Robinson-Schensted Algorithm

Recall $f^\lambda = \# \text{SYTs of sh. } \lambda$. We will prove following identity:

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

$$\text{e.g. } n=2 \Rightarrow (f^{\square})^2 + (f^{\square\square})^2 = 1^2 + 1^2 = 2! \quad \checkmark$$

$$n=3 \Rightarrow (f^{\square\square})^2 + (f^{\square\square\square})^2 + (f^{\square\square\square})^2 = 1^2 + 2^2 + 1^2 = 6 = 3! \quad \checkmark$$

May seem like a strange formula, but has algebraic meaning.

We will explain a bijection pf. of this thm., using a very important procedure called Robinson-Schensted Algorithm.

Observe that

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

and of course

$$n! = \# \text{ permutations in } S_n$$

so the thm. would follow from a bijection

$$S_n \rightarrow \left\{ \begin{array}{l} \text{pairs } (P, Q) \text{ of SYTs w/ } \text{sh}(P) = \text{sh}(Q) + n \end{array} \right\}$$

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 its increase down rows/cols, but #s are not $1, 2, \dots, n$. That's ok.

call it i

How do we carry out the insertion? 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. $\begin{matrix} 1 & 3 & 6 & 9 \\ 2 & 8 & 10 \end{matrix} \xleftarrow{5 \text{ bumps}} \begin{matrix} 1 & 3 & 5 & 9 \\ 2 & 8 & 10 \end{matrix} \xleftarrow{6 \text{ moves}} \begin{matrix} 1 & 3 & 5 & 9 \\ 2 & 6 & 10 \\ 4 & 7 \end{matrix} \xleftarrow{8 \text{ moves}} \begin{matrix} 1 & 3 & 5 & 9 \\ 2 & 6 & 10 \\ 4 & 8 \\ 7 \end{matrix} \boxed{\begin{matrix} 1 & 3 & 5 & 9 \\ 2 & 6 & 10 \\ 4 & 8 \\ 7 \end{matrix}}$

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 $\sigma = (\sigma_1 \ \sigma_2 \ \sigma_3 \ \dots \ \sigma_n) \in S_n$.

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 an empty tableau)

e.g. $\sigma = (1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7)$

$\emptyset, \boxed{5}, \boxed{\frac{2}{5}}, \boxed{\frac{2}{3}}, 2 \ 3 \ 6, 2 \ 3 \ 4, 1 \ 3 \ 4, \boxed{\begin{matrix} 1 & 3 & 4 & 7 \\ 2 & 6 \\ 5 \end{matrix}} = P$

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

e.g. $\emptyset, \boxed{1}, \boxed{\frac{1}{2}}, \boxed{\frac{1}{3}}, 1 \ 3 \ 4, 1 \ 3 \ 4, 1 \ 3 \ 4, \boxed{\begin{matrix} 1 & 3 & 4 & 7 \\ 2 & 5 \\ 6 \end{matrix}} = Q$

4/4

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

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

Pf: As w/ Hillman-Grassl, goal is to show we can locally undo steps.

In other words, we can describe inverse $(P, Q) \xrightarrow{RS^{-1}} \tau$. Here is how that works. Suppose we are given (P, Q) :

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

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/inset" this entry out of P:

- if it is in row 2, simply remove it,
- otherwise, have it replace the ^{largest} # less than it in the row above, and bump that # out and repeat.

$$\text{e.g. } \begin{matrix} 1 & 3 & 4 & 5 \text{ bump 2} \\ 2 & 6 \end{matrix} \xrightarrow{\substack{\text{rev.} \\ \text{inset} \\ \text{twr}}} \begin{matrix} 1 & 3 & 4 & 2 \text{ bump 2} \\ 5 & 6 \end{matrix} \xrightarrow{\substack{\text{rev.} \\ \text{inset} \\ \text{twr}}} \begin{matrix} 2 & 3 & 4 \\ 5 & 6 \end{matrix} \xrightarrow{\substack{\text{rev.} \\ \text{inset} \\ \text{twr}}} \boxed{1} \xrightarrow{\substack{\text{rev.} \\ \text{inset} \\ \text{twr}}} \text{removed}$$

Then write down $\sigma_n := \# \text{ removed from rev. insertion}$.

And remove n from Q, and repeat same step but

now with box containing $n-1$ in Q. In this way,

we build up sequence $\sigma_n, \sigma_{n-1}, \dots, \sigma_1$ and

then $\tau = \sigma_1 \sigma_2 \dots \sigma_n$ 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... leave to you as Exercise). 

The Robinson-Schensted-Knuth Algorithm

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

Thm (Cauchy identity)

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

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 $\vec{x} = \{x_1, x_2, \dots\}$ and $\vec{y} = \{y_1, y_2, \dots\}$

• it is an identity in $\mathbb{C}[[x_1, x_2, \dots, y_1, y_2, \dots]]$

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 $n \geq 1$:

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

here we use 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 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$, $\beta = (\beta_1, \beta_2, \dots, \beta_n)$ on LHS + RHS.

On LHS,

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

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 $\prod_{i,j \geq 1} \frac{1}{1 - x_i y_j} = \#\left\{ \begin{array}{l} n \times n \text{ if-matrices } M \\ \text{w/ } \text{row}_i(M), \text{row}_2(M), \dots \\ \text{col}_j(M), \text{col}_2(M), \dots \end{array} \right\}$

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

4/6 Hence the Cauchy identity will follow from the existence of a bij.
 $\{n \times n \mathbb{N}\text{-matrices } M\} \rightarrow \{(P, Q) : \text{SSYT}s \text{ w/ } \text{sh}(P) = \text{sh}(Q)\}$

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

$$\sigma = 2 \ 1 \ 3 \longrightarrow \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The $n \times n$ matrices w/ row/col sums all = 1
are exactly the permutation matrices, and
similarly, the SSYT's 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. $\begin{matrix} 1 & 2 & 2 \\ 3 & 3 \end{matrix} \xleftarrow{\text{bump } 2} \begin{matrix} 1 & 1 & 2 \\ 3 & 3 \end{matrix} \xleftarrow{\text{bump } 3} \begin{matrix} 1 & 1 & 2 \\ 2 & 3 \\ 3 \end{matrix}$

Next, we need to explain what sequence of #'s we are inserting.

Given matrix M , form biarray $(\begin{smallmatrix} a_1 & a_2 & \dots & a_k \\ b_1 & b_2 & \dots & b_k \end{smallmatrix})$ that has $m_{i,j}$ copies of j and is s.t. $a_1 \leq \dots \leq a_k$

$$\bullet b_i \leq b_j \text{ if } i \leq j \text{ and } a_i = a_j$$

e.g. $M = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 1 & 1 & 0 \end{pmatrix} \rightarrow \text{biarray } (\begin{matrix} 1 & 1 & 1 & 2 & 2 & 3 & 3 \\ 1 & 3 & 3 & 2 & 2 & 1 & 2 \end{matrix})$

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

$$\emptyset \leftarrow 1, \boxed{1} \leftarrow 3, \boxed{1 \boxed{3}} \leftarrow 3, 133 \leftarrow 2, \underset{3}{123} \leftarrow 2, \underset{33}{122} \leftarrow 1, \underset{23}{112} \leftarrow 2$$

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, \dots, a_k :

$$\emptyset, \boxed{1}, \boxed{1 \boxed{1}}, \boxed{1 \boxed{1} \boxed{1}}, \underset{2}{1 \boxed{1} \boxed{1}}, \underset{22}{1 \boxed{1} \boxed{1}}, \underset{22}{1 \boxed{1} \boxed{1}}, \underset{3}{1 \boxed{1} \boxed{1}} = Q$$

The map $M \xrightarrow{\text{RSK}} (P, Q)$ 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

samueljhopkins.com/docs/rsk.pdf

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

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

$$\begin{matrix} g_{1,1} & g_{1,2} & g_{1,3} & \cdots & g_{1,n} \\ & g_{2,2} & \cdots & g_{2,n} & \\ & & g_{3,3} & \cdots & g_{3,n} \\ & & & \ddots & \\ & & & & g_{n,n} \end{matrix}$$

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

such that $g_{i,j} \geq g_{i+1,j+1} \geq g_{i,j+1} \neq g_{i,j}$.

There is a bijection

$$\{\text{SSYTs with entries } \begin{matrix} g_{i,j} \\ \text{in } \{1, 2, \dots, n\} \end{matrix}\} \longrightarrow \{\text{GT patterns of size } n\}$$

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

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

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

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

Exercise: prove this really is a bijection.

Recall RSK is a bijection

$$M, \text{ } n \times n \text{ matrix} \xrightarrow{\text{RSK}} (P, Q) \text{ pair of SSYT w/ } sh(P) = sh(Q) \text{ and entries } \subseteq [n]$$

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

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

Notice that since $\text{sh}(P) = \text{sh}(Q)$, 1st rows of $\text{GT}(P), \text{GT}(Q)$ are same.
So we can glue $\text{GT}(P)$ and $\text{GT}(Q)$ into a matrix:

e.g. $\begin{array}{c|cc} & \text{GT}(Q) & \\ \hline & \text{GT}(P) & \end{array} = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 2 & 4 & 4 \end{pmatrix}$ $\xrightarrow{\text{UHST}}$ observe: inequalities
on GT-pattern become: weakly increasing along rows + cols!

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

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

But what properties does this bij. satisfy?

Recall that $\text{con}(P) = \text{col}(M)$, and notice that

Sum of i^{th} row of $\text{GT}(P)$ = # entries in $\{1, 2, \dots, i\}$ in P

Sum of i^{th} diagonal of π = sum of 1st i columns of M .

e.g. $\pi = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 2 & 4 & 4 \end{pmatrix}$ $M = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 1 & 1 & 0 \end{pmatrix}$
 $\xrightarrow{\text{2nd diag.}} 4+1 = 1+2+1+1 \geq$ two columns in M

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

e.g. $\pi = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 2 & 4 & 4 \end{pmatrix}$ $M = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 1 & 1 & 0 \end{pmatrix}$
 $\xrightarrow{\text{1st diag.}} 3 = 1+2 \quad \text{one row in } M,$
 $(2)_{\text{row}} = (1)_{\text{row}}$
 $1+3 = 2 \text{ boxes here}$

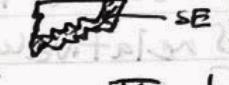
4/11

We will give another construction of this map $M \xrightarrow{\text{RSK}} \pi$ which converts row/col sums to diagonal sums.

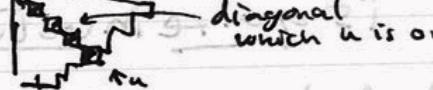
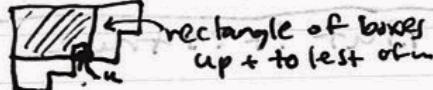
Actually, we will define an even more general bijection.

Then for any partition shape $\lambda \models k$ bijection

$\{\text{N-fillings of } \lambda \text{ M}\} \xrightarrow{\text{RSK}} \{\text{rev. plane partitions } \pi \text{ s.t. } \pi_{\text{sh}} = \lambda\}$

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

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



e.g. $\lambda = (2, 1)$

$$\begin{array}{|c|c|} \hline a & b \\ \hline c & \\ \hline \end{array} \xrightarrow{\text{RSK}} \begin{array}{|c|c|} \hline a & a+b \\ \hline a+c & \\ \hline \end{array}$$

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

$$\hat{M} = \boxed{}$$

$$M = \boxed{} \in \text{new box } u \text{ int band}$$

Then define ~~RSK~~ π from $\hat{\pi}$ by:

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

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

$$u \begin{smallmatrix} w \\ x \\ z \end{smallmatrix} y \mapsto u \begin{smallmatrix} w \\ x' \\ z \end{smallmatrix} y \text{ where } \begin{cases} x' = \max(u, w) \\ \quad + \min(y, z) \\ \quad - x \end{cases}$$

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

Then we define $M \xrightarrow{\text{RSK}} \pi$,

e.g.

$$\begin{array}{|c|c|} \hline a & b \\ \hline c & \\ \hline \end{array} \xrightarrow{\text{RSK}} \begin{array}{|c|c|} \hline a & a+b \\ \hline a+c & \\ \hline \end{array}$$

so

$$\begin{array}{|c|c|} \hline a & b \\ \hline c & d \\ \hline \end{array} \xrightarrow{\text{RSK}} \begin{array}{|c|c|} \hline \min(b, c) & a+b \\ \hline a+c & d+\max(b, c) \\ \hline \end{array}$$

$$\begin{array}{|c|c|} \hline \min(b, c) & a+b \\ \hline a+c & d+\max(b, c) \\ \hline \end{array}$$

Exercise: Check $M = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 1 & 1 & 0 \end{pmatrix} \xrightarrow{\text{RSK}} \pi = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 2 & 4 & 4 \end{pmatrix}$ 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 $\text{Toggle RSK} = \text{insertion RSK}$

is quite involved! But it is true...

And toggle RSK makes one symmetry clear:

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

Pf: At level of r.p.p.s, says that $M^t \xrightarrow{\text{RSK}} \pi^t$, transpose
and this is obvious from toggle description!

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

One final observation is that if $M = (m_{i,j}) \xrightarrow{\text{RSK}} \pi$

then $\sum_{(i,j) \in \lambda} h(u) \cdot m_{i,j} = |\pi| = \sum_{i,j} \pi_{i,j}$

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

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

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

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