The Robinson-Schensted-Knuth Correspondence

QFT notes, week 5

In this lecture, we give a description of the correspondences between pairs of tableau and certain structures, due to Robinson, Schensted and Knuth.

Recall that a semi-standard Young tableau (tableau, or SSYT) is a filling of a tableau $\lambda \vdash n$ with natural numbers which is

1. weakly increasing across each row, and
2. strictly increasing down each column.

A standard Young tableau (SYT) is a semi-standard Young tableau where the entries are 1 through to $n$.

We have the following three correspondences, the first two of which will be described in detail in this lecture.

1. The Robinson correspondence gives a bijection between the symmetric group $S_n$ and ordered pairs $(P, Q)$, where $P, Q$ are SYT.

2. The Robinson-Schensted correspondence gives a bijection between words and ordered pairs $(P, Q)$, where $P$ is a SSYT and $Q$ is a SYT.

3. The Robinson-Schensted-Knuth correspondence gives a bijection between biwords and ordered pairs $(P, Q)$, where $P, Q$ are SSYT.

Bumping

Given a tableau $T$ and a natural number $x$, we can construct a new tableau $T \leftarrow x$, with

- one more box than $T$, and
- one more entry labelled $x$.

The entries of $T$ may be moved around. We construct $T \leftarrow x$ with the following Bumping Algorithm.

If $x$ is at least as large as the entries of the first row of $T$, append $x$ to the end of the row. Otherwise, find the first entry $x_1$ strictly larger than $x$. Replace the entry in this box with $x$ (the 'bumping' step), and repeat for $x_1$ for the next row. Repeat this procedure until an entry is bumped out of the bottom row in which case it forms a new row with one entry, or an entry can be placed at the end of the row it is bumped into.
**Example 1.** Consider inserting 2 into the following tableau.

\[
\begin{array}{cccc}
1 & 2 & 2 & 3 \\
2 & 3 & 5 & 5 \\
4 & 4 & 6 & \\
5 & 6 & \\
\end{array}
\]

2 is less than 3 and will therefore bump the 3 in the first row (in red). We now wish to insert 3 into the second row of the following tableau.

\[
\begin{array}{cccc}
1 & 2 & 2 & 2 \\
2 & 3 & 5 & 5 \\
4 & 4 & 6 & \\
5 & 6 & \\
\end{array}
\]

We now repeat this process. 3 will bump the 5 in red to give the following tableau.

\[
\begin{array}{cccc}
1 & 2 & 2 & 2 \\
2 & 3 & 3 & 5 \\
4 & 4 & \textcolor{red}{6} & \\
5 & 6 & \\
\end{array}
\]

5 bumps 6.

\[
\begin{array}{cccc}
1 & 2 & 2 & 2 \\
2 & 3 & 3 & 5 \\
4 & 4 & 5 & \\
5 & 6 & \\
\end{array}
\]

Finally, we can add 6 to the end of the last row, and so the algorithm concludes with the following resultant tableau.

\[
\begin{array}{cccc}
1 & 2 & 2 & 2 \\
2 & 3 & 3 & 5 \\
4 & 4 & 6 & \\
5 & 6 & 6 & \\
\end{array}
\]

Note that the result of the bumping algorithm is always a SSYT. Clearly the rows are still weakly increasing, so it suffices to show that the columns are strictly increasing. If \(y\) bumps \(z\), the entry below \(z\) is strictly larger than \(z\), so after bumping, \(z\) will either stay in the same column, or move to the left. In either case, \(z\) is below \(y\) or a number less than \(y\), so the strictly increasing property is preserved.

This procedure is invertible provided we know which box has been added. Let \(y\) be the element in the added box. Then \(y\) will bump the rightmost entry strictly less that \(y\) in the row above, repeating until an entry in the first row. Note that the new box will always be created in the right side of the tableau.

**Example 2.** Consider example 1 but in reverse. Note in this case we are still bumping the element in red, but this time bumping 'upwards'.

2
Tableaux as Words

Given a tableau $T$, we define the word of $T$, denoted $w(T)$ to be the reading of the entries of $T$, from left to right and bottom to top.

**Example 3.** The word corresponding to the tableau

\[
\begin{array}{cccc}
1 & 4 & 4 & 6 \\
2 & 5 \\
4 \\
\end{array}
\]

is 4251446.

A tableau $T$ can be recovered from its word $w(T)$, by adding a new row whenever one entry is strictly greater than the next. Note that not all words are of the form $w(T)$.

We can think of the bumping algorithm as acting on words. We have two elementary operations that allow us to do this.

\[
yzx \mapsto yxz \quad \text{if} \quad x < y \leq z
\]

\[
xzy \mapsto zxy \quad \text{if} \quad x \leq y < z
\]

An **elementary Knuth transformation** of a word $w$ is the application of (1), (2) or their inverses on $w$. Two words are **Knuth equivalent** (denoted by $\equiv$) if they can be changed into each other via a sequence Knuth transformation. This is an equivalence relation.

**Proposition 4.** Let $T$ be a tableau and $k$ be a natural number. Then $w(T \leftarrow k) \equiv w(T)k$.

Given a word $w = w_1, \ldots, w_k$, we can therefore speak of the tableau of the word $T(w)$ as

\[T(w) = ((\square \leftarrow w_1) \ldots \leftarrow w_k)\]

**Corollary 5.** Every word is Knuth equivalent to the word of a semi-standard Young tableau.

**Example 6.** We consider the tableau $T$ in Example 1. The word corresponding to this tableau is 56446235122. Adding brackets for convenience, we consider the word (56)(446)(2355)(122)2, i.e. $(T \leftarrow 2)$. We therefore form a series of Knuth equivalent words, corresponding to the tableaux present in Example 1:

\[
(56)(446)(2355)(122)2 \equiv (56)(446)(2355)3(122) \\
\equiv (56)(446)(2335)(122) \\
\equiv (56)(445)(2335)(122) \\
\equiv (56)(445)(2335)(122).
\]
The RSK Correspondence

Given any word $w$ of length $r$, the RSK Algorithm gives us a pair $(P, Q)$ of tableau, and we write $w \xrightarrow{\text{RSK}} (P, Q)$. $P$ is defined to be $T(w)$, and $Q$ is known as the recording (insertion) tableau with entries $1, \ldots, r$, where $k$ is placed in the box added at the $k^{\text{th}}$ step in the construction of $P$. The resulting pair at the $k^{\text{th}}$ step of the algorithm is denoted $(P_k, Q_k)$.

Note that $Q_k$ is a standard Young tableau, since we are adding the new box at the end of the row of $Q_{k-1}$.

**Example 7.** Consider the word $(5)(48)(23)$. Our pairs $(P_i, Q_i)$ are as follows:

\[
\begin{align*}
P_1 &= \begin{array}{cc}
5 \\
4 & 5
\end{array} & Q_1 &= \begin{array}{cc}
1 \\
\end{array} \\
P_2 &= \begin{array}{cc}
4 & 8 \\
5
\end{array} & Q_2 &= \begin{array}{cc}
1 & 2 \\
\end{array} \\
P_3 &= \begin{array}{cc}
2 & 8 \\
4 & 5
\end{array} & Q_3 &= \begin{array}{cc}
1 & 3 \\
2 & 4
\end{array} \\
P_4 &= \begin{array}{cc}
2 & 3 \\
4 & 8
\end{array} & Q_4 &= \begin{array}{cc}
1 & 3 \\
2 & 5 & 4
\end{array} \\
P_5 &= \begin{array}{cc}
5 \\
\end{array} & Q_5 &= \begin{array}{cc}
1 & 2 & 5 & 4
\end{array}
\end{align*}
\]

Given a pair $(P, Q)$, we can also recover our original word. To go from $(P_k, Q_k)$ to $(P_{k-1}, Q_{k-1})$, remove the largest numbered box in $Q_k$ to get to $Q_{k-1}$, and apply the reverse bumping procedure with that block on $P_k$ to obtain $P_{k-1}$. The element bumped out of $P_k$ is the $k^{\text{th}}$ element of $w$. Hence every pair $(P, Q)$ can arise in this way.

We therefore have a one-to-one correspondence of words of length $r$ using letters from $[n]$ and pairs $(P, Q)$ of tableaux of the same shape with $r$ boxes, where the entries of $P$ are from $[n]$ and $Q$ is standard. Indeed, if $w$ is a permutation of $[n]$, then we obtain our desired bijection from the symmetric group $S_n$ and pairs of standard Young tableaux of the same shape.

**Corollary 8.** Let $f^\lambda$ be the number of standard Young tableaux of shape $\lambda$. Then

$$\sum_{\lambda \vdash n} (f^\lambda)^2 = n!.$$ 

The following result is also nice to know.

**Proposition 9.** If $w$ is a permutation and $w \xrightarrow{\text{RSK}} (P, Q)$, then $w^{-1} \xrightarrow{\text{RSK}} (Q, P)$. 

4