Young tableaux and Gelfand-Tsetlin patterns

2017-05-16

young_tableaux gelfand_tsetlin_patterns

One can convert a (semi-standard) Young tableau T to a Gelfand-Tsetlin pattern $(\lambda_j^k)_{1 \le j \le k}$ in the following way:

$$\lambda_j^k = \text{number of entries } \le k \text{ in the } j\text{th row of } T.$$

This map is invertible: given GT pattern (λ_j^k), one can reconstruct the corresponding tableau by putting $\lambda_j^k - \lambda_j^{k-1}$ of ks and λ_j^j of js in the jth row.

Example.

$$\begin{array}{ccccc} 1 & 2 & 2 & 3 & 3 \\ 2 & 3 & 4 & & \\ 4 & & & & \end{array}$$

corresponds to

$$(\lambda_1^1, \lambda_1^2, \lambda_2^2, \lambda_1^3, \lambda_2^3, \lambda_3^3, \lambda_1^4, \lambda_2^4, \lambda_3^4, \lambda_4^4) = (1, 3, 1, 5, 2, 0, 5, 3, 1, 0)$$

□

The shape of the Young tableau is thus the bottom row of the Gelfand Tsetlin pattern. In the above example it is $(5, 3, 1)$.

For example here are sample Python codes of the two conversions:

```
def Tableau2GT(T):
    if T == []:
        return []
    l = max([max(i) for i in T])
    Lambda = [[0] * (k + 1) for k in range(l)]
    for j, row in enumerate(T):
        for i in range(j, l):
            Lambda[i][j] = sum([1 if e <= i + 1 else 0 for e in row])
```

```
    return Lambda

def GT2Tableau(Lambda):
    T = []
    l = len(Lambda)
    for j in range(l):
        row = [j + 1] * Lambda[j][j]
        for k in range(j + 1, l):
            row += [k + 1] * (Lambda[k][j] - Lambda[k - 1][j])
        T += [row]
    T += [[]]
    return T[:T.index([])]
```