

Recursive Algorithm for Computing the Number of Partitions that are Less than a Given Partition with the Sum of the Partition Smaller than a Nonnegative Integer

This note presents a recursive algorithm for computing the number of partitions ν such that $\nu \leq \kappa$ and $0 \leq |\nu| \leq n$, where $\kappa = (k_1, \dots, k_r)$ is the given partition with length r and n is a nonnegative integer.

Let $f_r(\kappa, n)$ denotes the number of ν 's such that $0 \leq |\nu| \leq n$ and $\nu \leq \kappa$. We have the following recursive relation of $f_r(\kappa, n)$

$$f_r(\kappa, n) = \sum_{i=0}^{\min[n, k_r]} f_{r-1}(\kappa_{(r)}, n - i), \quad (1)$$

where $\kappa_{(r)} = (k_1, \dots, k_{r-1})$. The recursive relation holds because each term on the right hand side tells us that if $\nu_r = i$, then there are $f_{r-1}(\kappa_{(r)}, n - i)$ different $\nu_{(r)} = (\nu_1, \dots, \nu_{r-1})$ that satisfy the constraints $\nu_{(r)} \leq \kappa_{(r)}$ and $0 \leq |\nu_{(r)}| \leq n - i$. Note that when $k_r = 0$, we have $f_r(\kappa, n) = f_{r-1}(\kappa_{(r)}, n)$.

There are two cases that we can obtain explicit expression of $f_r(\kappa, n)$. The first case is when $\min_{1 \leq i \leq r} k_i \geq n$. In this case, the constraint $\nu \leq \kappa$ is not binding and we have

$$f_r(\kappa, n) = \binom{n+r}{r}. \quad (2)$$

The second case is when $|\kappa| \leq n$. In this case, the constraint $|\nu| \leq n$ is not binding and we have

$$f_r(\kappa, n) = \prod_{i=1}^r (k_i + 1). \quad (3)$$

From the recursive relation, we can get

$$\begin{aligned} f_r(\kappa, n) - f_r(\kappa, n-1) &= \sum_{i=0}^{\min[k_r, n]} f_{r-1}(\kappa_{(r)}, n - i) - \sum_{i=0}^{\min[k_r, n-1]} f_{r-1}(\kappa_{(r)}, n - 1 - i) \\ &= \sum_{i=0}^{\min[k_r, n]} f_{r-1}(\kappa_{(r)}, n - i) - \sum_{i=1}^{\min[k_r+1, n]} f_{r-1}(\kappa_{(r)}, n - i) \\ &= \begin{cases} f_{r-1}(\kappa_{(r)}, n) - f_{r-1}(\kappa_{(r)}, n - k_r - 1) & \text{if } k_r + 1 \leq n \\ f_{r-1}(\kappa_{(r)}, n) & \text{if } k_r + 1 > n \end{cases} \end{aligned} \quad (4)$$

Therefore, we have

$$f_r(\kappa, n) = \begin{cases} f_r(\kappa, n-1) + f_{r-1}(\kappa_{(r)}, n) - f_{r-1}(\kappa_{(r)}, n - k_r - 1) & \text{if } k_r + 1 \leq n \\ f_r(\kappa, n-1) + f_{r-1}(\kappa_{(r)}, n) & \text{if } k_r + 1 > n \end{cases} \quad (5)$$

This relation can be obtained using a combinatorial argument. Note that the difference between $f_r(\kappa, n)$ and $f_r(\kappa, n - 1)$ is the number of ν 's with $|\nu| = n$ and $\nu \leq \kappa$. For any $\nu_{(r)} \leq \kappa_{(r)}$, we can potentially choose $\nu_r = n - |\nu_{(r)}|$ to make $|\nu| = n$. However, ν_r must satisfy the constraint $0 \leq \nu_r \leq k_r$, which implies $n \geq |\nu_{(r)}| \geq n - k_r$. Therefore, the number of ν 's with $|\nu| = n$ and $\nu \leq \kappa$ is given by

$$f_{r-1}(\kappa_{(r)}, n) - f_{r-1}(\kappa_{(r)}, n - k_r - 1), \quad (6)$$

with the last term vanishes when $n < k_r + 1$.

The boundary conditions for the above recursive relation are

$$f_r(\kappa, 0) = 1, \quad (7)$$

and

$$f_1(k_1, n) = \min(k_1, n) + 1. \quad (8)$$