FIND-MAX 中的算法分析

问题求解 Open Topic II

黄文睿　学号: 221180115

南京大学　2022 级计算机拔尖班

2023 年 4 月 21 日

参考资料

主要参考了 Donald Knuth 教授于 2015 年在斯坦福的讲座 The Analysis of Algorithms.

其视频可在 YouTube 上观看.

目录

Find-Max 算法

尝试分析以下算法中 $(*)$ 句的执行次数, 元素各不相同

```
function FIND-MAX(a[1..n], n)
    m ← a[n]
    k ← n − 1
    while k ≠ 0 do
        if a[k] > m then
            m ← a[k]        // (*)
        end if
        k ← k − 1
    end while
    return m
end function
```

分析内容

一般会分析 $(*)$ 执行次数的最小值、最大值、均值、方差.

分析内容

一般会分析 $(*)$ 执行次数的最小值、最大值、均值、方差.

- 最小值: 0 次, 当 $a[n]$ 是整个序列的最大值.
- 最大值: $n-1$ 次, 当序列是降序的.
- 均值: $H_n - 1$ 次, H_n 为调和数. (见 TC §5.3 Indicator random variables)

分析内容

一般会分析 $(*)$ 执行次数的最小值、最大值、均值、方差.

- 最小值: 0 次, 当 $a[n]$ 是整个序列的最大值.
- 最大值: $n-1$ 次, 当序列是降序的.
- 均值: $H_n - 1$ 次, H_n 为调和数. (见 TC §5.3 Indicator random variables)

那么方差呢? 需要新的方法.

斯特林数的定义

斯特林轮换数

将 n 个两两不同的元素划分为 k 个互不区分的非空轮换 (圆排列) 的方案数, 记为 $\begin{bmatrix} n \\ k \end{bmatrix}$.

斯特林数的定义

斯特林轮换数

将 n 个两两不同的元素划分为 k 个互不区分的非空轮换 (圆排列) 的方案数, 记为 $\begin{bmatrix} n \\ k \end{bmatrix}$.

有递推公式为

$$\begin{bmatrix} n \\ k \end{bmatrix} = (n-1)\begin{bmatrix} n-1 \\ k \end{bmatrix} + \begin{bmatrix} n-1 \\ k-1 \end{bmatrix}, (n \geq 1, k \geq 1).$$

初始条件为 $\begin{bmatrix} n \\ 0 \end{bmatrix} = [n = 0]$.

递推公式的证明

$$\begin{bmatrix} n \\ k \end{bmatrix} = (n-1)\begin{bmatrix} n-1 \\ k \end{bmatrix} + \begin{bmatrix} n-1 \\ k-1 \end{bmatrix}, (n \geq 1, k \geq 1).$$

考虑第 n 个元素的位置:

1. 若它接在了原来的某个元素后面, 那么没有新增轮换, 方案数为 $(n-1)\begin{bmatrix} n-1 \\ k \end{bmatrix}$.
2. 若它独占一个轮换, 那么前 $n-1$ 个元素需要占 $k-1$ 个轮换, 方案数为 $\begin{bmatrix} n-1 \\ k-1 \end{bmatrix}$.

斯特林数和原问题的关系

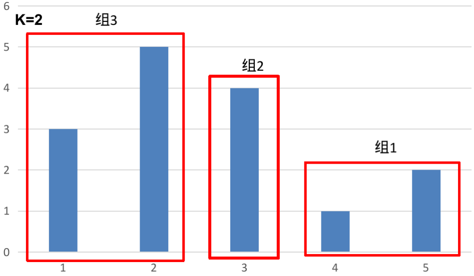
设在原问题中, $(*)$ 执行 k 次的概率为 $p_{n,k}$, 那么断言

$$p_{n,k} = \frac{1}{n!} \begin{bmatrix} n \\ k+1 \end{bmatrix}, (0 \leq k \leq n-1).$$

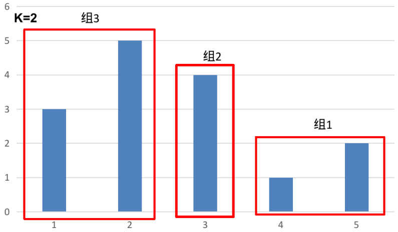
斯特林数和原问题的关系

设在原问题中，$(*)$ 执行 k 次的概率为 $p_{n,k}$，那么断言

$$p_{n,k} = \frac{1}{n!}\begin{bmatrix} n \\ k+1 \end{bmatrix}, (0 \le k \le n-1).$$

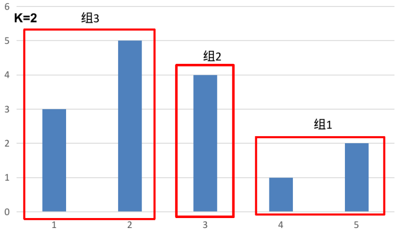


斯特林数和原问题的关系——证明



- 组内最大值为最右——圆排列;
- 组间满足递减——组间无序.

斯特林数和原问题的关系——证明



- 组内最大值为最右——圆排列;
- 组间满足递减——组间无序.

原问题等价于将 n 个元素划分为 $k+1$ 个互不区分的非空轮换的方案数, 再除以 $n!$ 即为概率.

斯特林轮换数的生成函数

考虑对固定的 n, 一行斯特林轮换数的生成函数 $S_n(z)$.

斯特林轮换数的生成函数

考虑对固定的 n，一行斯特林轮换数的生成函数 $S_n(z)$.

于是 $S_0(z) = 1, S_1(z) = z$，当 $n \geq 2$ 有

$$
\begin{aligned}
S_n(z) &= \sum_{k\geq 0} \begin{bmatrix} n \\ k \end{bmatrix} z^k \\
&= \sum_{k\geq 1} \left((n-1) \begin{bmatrix} n-1 \\ k \end{bmatrix} + \begin{bmatrix} n-1 \\ k-1 \end{bmatrix} \right) z^k \\
&= (n-1) \sum_{k\geq 1} \begin{bmatrix} n-1 \\ k \end{bmatrix} z^k + z \sum_{t\geq 0} \begin{bmatrix} n-1 \\ t \end{bmatrix} z^t \\
&= (n-1+z) S_{n-1}(z).
\end{aligned}
$$

斯特林轮换数的生成函数

解得通项为

$$S_n(z) = \sum_{k\geq 0} \begin{bmatrix} n \\ k \end{bmatrix} z^k = z(z+1)\cdots(z+n-1) = z^{\overline{n}}.$$

斯特林轮换数的生成函数

解得通项为

$$S_n(z)=\sum_{k\geq 0}\begin{bmatrix}n\\k\end{bmatrix}z^k=z(z+1)\cdots(z+n-1)=z^{\overline{n}}.$$

很漂亮的结论，我们先放一边.

概率生成函数

概率生成函数的定义

对一个取值为非负整数的随机变量 X，设 $p_k = Pr\{X = k\}$，其中 $k \geq 0$，定义其概率生成函数为

$$G(z) = \sum_{k \geq 0} p_k z^k.$$

概率生成函数

概率生成函数的定义

对一个取值为非负整数的随机变量 X, 设 $p_k = Pr\{X = k\}$, 其中 $k \geq 0$, 定义其概率生成函数为

$$G(z) = \sum_{k \geq 0} p_k z^k.$$

它具有以下两个约束:

1. $G(z)$ 各项非负;
2. $G(1) = \sum_{k \geq 0} p_k = 1$.

利用生成函数求均值和方差

对 $G(z)$ 求导得

$$G'(z) = \sum_{k\geq 1} kp_k z^{k-1}, G'(1) = \sum_{k\geq 1} kp_k.$$

$$G''(z) = \sum_{k\geq 2} k(k-1)p_k z^{k-2}, G''(1) = \sum_{k\geq 2} k(k-1)p_k.$$

从而可以得到均值和方差为

$$E(X) = \sum_{k\geq 1} kp_k = G'(1);$$

$$D(X) = \sum_{k\geq 1} k^2 p_k - E^2(X) = G''(1) + G'(1) - [G'(1)]^2.$$

概率生成函数的卷积

设 X 的生成函数为 $G(z)$, Y 的生成函数为 $H(z)$, 卷积为 $F = G * H$.

概率生成函数的卷积

设 X 的生成函数为 $G(z)$, Y 的生成函数为 $H(z)$, 卷积为 $F = G * H$.

卷积的意义

1. 若 X 和 Y 独立, 那么 $F(z) = (G * H)(z)$ 是 $X + Y$ 的生成函数;
2. 若不一定独立, 则 $F(z)$ 不一定是 $X + Y$ 的生成函数, 但这不影响后文 F 的分析, 只是 F 和 $X + Y$ 不一定相联系.

概率生成函数的卷积

设 X 的生成函数为 $G(z)$, Y 的生成函数为 $H(z)$, 卷积为 $F = G * H$.

卷积的意义

1. 若 X 和 Y 独立, 那么 $F(z) = (G * H)(z)$ 是 $X + Y$ 的生成函数;
2. 若不一定独立, 则 $F(z)$ 不一定是 $X + Y$ 的生成函数, 但这不影响后文 F 的分析, 只是 F 和 $X + Y$ 不一定相联系.

验证 F 是概率生成函数:

1. F 各项非负, 因为 G 和 H 各项非负;
2. $F(1) = G(1)H(1) = 1$.

卷积的均值

设 X 的生成函数为 $G(z)$, Y 的生成函数为 $H(z)$, 卷积为 $F = G * H$, 对应的随机变量为 Z.

则卷积的均值为

$$
\begin{aligned}
E(Z) &= F'(1) \\
&= [G'(z)H(z) + G(z)H'(z)]\Big|_{z=1} \\
&= G'(z) + H'(z) \\
&= E(X) + E(Y).
\end{aligned}
$$

卷积的方差

设 X 的生成函数为 $G(z)$, Y 的生成函数为 $H(z)$, 卷积为 $F = G * H$, 对应的随机变量为 Z.

则卷积的方差为

$$
\begin{aligned}
D(Z) &= F''(1) + F'(1) - [F'(1)]^2 \\
&= \{G''(z)H(z) + 2G'(z)H'(z) + G(z)H''(z) + \\
&\qquad + G'(z) + H'(z) - [G'(z) + H'(z)]^2\}\Big|_{z=1} \\
&= G''(1) + G'(1) - [G'(1)]^2 + H''(1) + H'(1) - [H'(1)]^2 \\
&= D(X) + D(Y).
\end{aligned}
$$

概率生成函数的性质 (小结)

对两个概率生成函数 $G(z)$ 和 $H(z)$, 有

$$E(G) = G'(1),$$
$$D(G) = G''(1) + G'(1) - [G'(1)]^2.$$

以及卷积的性质

$$E(G * H) = E(G) + E(H),$$
$$D(G * H) = D(G) + D(H).$$

解决原题

设 X_n 表示对于长度为 $n(n \geq 1)$ 的数组, $(*)$ 的执行次数. 则它的概率生成函数

$$\begin{aligned} P_n(z) &= \sum_{k\geq 0} p_{n,k} z^k \\ &= \sum_{k\geq 0} \frac{1}{n!} \begin{bmatrix} n \\ k+1 \end{bmatrix} z^k \\ &= \frac{1}{n! \cdot z} \sum_{t\geq 1} \begin{bmatrix} n \\ t \end{bmatrix} z^t \\ &= \frac{1}{n!}(z+1)(z+2)\cdots(z+n-1). \end{aligned}$$

解决原题

$$P_n(z) = \frac{1}{n!}(z+1)(z+2)\cdots(z+n-1).$$

解决原题

$$P_n(z) = \frac{1}{n!}(z+1)(z+2)\cdots(z+n-1).$$

把它分成 $n-1$ 个概率生成函数的卷积, 即

$$P_n = A_2 * A_3 * \cdots * A_n,$$

其中

$$A_i(z) = \frac{z+i-1}{i}, (2 \le i \le n).$$

解决原题

$$P_n(z) = \frac{1}{n!}(z+1)(z+2)\cdots(z+n-1).$$

把它分成 $n-1$ 个概率生成函数的卷积, 即

$$P_n = A_2 * A_3 * \cdots * A_n,$$

其中

$$A_i(z) = \frac{z+i-1}{i}, (2 \le i \le n).$$

容易验证 $A_i(1) = 1$, 它是概率生成函数.

解决原题

那么要求 P_n 的均值和方差, 只要求出 A_i 的均值和方差. 由于 $A_i'(1) = 1/i, A_i''(z) = 0$, 有

$$E(A_i) = A_i'(1) = \frac{1}{i},$$
$$D(A_i) = A_i''(1) + A_i'(1) - [A_i'(1)]^2 = \frac{1}{i} - \frac{1}{i^2}.$$

解决原题

那么要求 P_n 的均值和方差, 只要求出 A_i 的均值和方差. 由于 $A_i'(1) = 1/i, A_i''(z) = 0$, 有

$$
\begin{aligned}
E(A_i) &= A_i'(1) = \frac{1}{i}, \\
D(A_i) &= A_i''(1) + A_i'(1) - [A_i'(1)]^2 = \frac{1}{i} - \frac{1}{i^2}.
\end{aligned}
$$

从而把它们的均值和方差求和, 得到

$$
\begin{aligned}
E(P_n) &= \sum_{i\ge 2} E(A_i) = \sum_{i\ge 2} \frac{1}{i} = H_n - 1, \\
D(P_n) &= \sum_{i\ge 2} D(A_i) = \sum_{i\ge 2} (\frac{1}{i} - \frac{1}{i^2}) = H_n - H_n^{(2)}.
\end{aligned}
$$

分析结果

根据分析, (*) 的执行次数的数据特征为

- 最小值: 0;
- 最大值: $n-1$;
- 均值: $H_n - 1$,
- 方差: $H_n - H_n^{(2)}$.

分析结果

根据分析, (*) 的执行次数的数据特征为

- 最小值: 0;
- 最大值: $n-1$;
- 均值: $H_n - 1$,
- 方差: $H_n - H_n^{(2)}$.

发现 “修改最大值” 的操作平均执行 $H_n - 1 = \Theta(\lg n)$ 次. 该支路容许运行复杂度稍高的算法, 但是不会影响总体的时间复杂度. 使用随机打乱可以消除针对性输入.

内容总结

Knuth 主要介绍了算法定量分析、计数技巧 (斯特林数等)、生成函数等工具在算法分析中的应用.

谢谢大家!