

有关概率和期望问题的研究

河北唐山一中 高融

摘要

在各类信息学竞赛中（尤其是 ACM 竞赛中），经常出现一些与概率和期望有关的题目。这类题目需要较高的数学水平和一定的算法技巧，必须经过仔细分析，选择合适的数学模型和算法才能顺利地解决问题。本文就对这类题目的一些常见方法进行了研究。

数学基础

这里写的数学基础是有关概率和期望的一点简单的计算法则，虽然我们都很熟悉，但是有时也可能会忘记使用，所以在这里列出来，也作为以后内容的基础。

概率的运算

- 两个互斥事件，发生任一个的概率等于两个事件的概率和
- 对于不相关的事件或者分步进行的事件，可以使用乘法原则。
- 对于一般情况 $p(A+B)=p(A)+p(B)-p(AB)$

期望的运算

- $E(\Phi) = \sum \Phi_i P_i$ ，这是期望的定义，其中 Φ_i 是一个取值，而 P_i 是取这个值的概率
- 期望有“线性”，也就是说对于不相关的两个随机变量 Φ 和 ξ ， $E(\Phi \pm \xi) = E(\Phi) \pm E(\xi)$ ； $E(\Phi \xi) = E(\Phi)E(\xi)$ ； $E(\Phi / \xi) = E(\Phi) / E(\xi)$
- 在某些情况下，期望可以表示成一个无穷的等比数列，然后利用极限的思想来求。

当然，这些只是最基础的知识，要解决好概率和期望的问题，还需要掌握一些组合数学的知识。

常用方法

方法 1 直接计算

这种方法说起来很简单，就是推导出一个数学公式，然后通过程序来计算这个式子的值。这样的题目在与概率和期望有关的题目中比例不小，但是由于它们大都需要一定的组合数学基础，而一旦推导出公式，对算法的要求并不太高，而时间复杂度往往也比较低，所以这类问题不是本文的重点。有关内容可以在任何一本组合数学书中学到。

例一 百事世界杯之旅¹

“……在2003年6月之前购买的百事任何饮料的瓶盖上都会有一个百事球星的名字。只要凑齐所有百事球星的名字，就可以参加百事世界杯之旅的抽奖活动，获取球星背包、随身听，更可以赴日韩观看世界杯。还不赶快行动！……”

你关上电视，心想：假设有n个不同球星的名字，每个名字出现的概率相同，平均需要买几瓶饮料才能凑齐所有的名字呢？

输入输出要求

输入一个数字n， $2 \leq n \leq 33$ ，表示不同球星名字的个数。

输出凑齐所有的名字平均需要购买的饮料瓶数。如果是一个整数则直接输出。否则就用下面样例中的格式分别输出整数部分和小数部分。分数必须是不可约的。

样例输入和输出

输入	输出
2	3
5	5 11— 12
17	340463 58----- 720720

分析 这是一道比较简单的概率和期望问题。只要确定好计算方法，就可以很容易的得到公式。如果单独考虑每一名球星，那么就中了命题人的圈套。因为考虑单独的一个球星的时候所买的“没用”的饮料在考虑其他球星的时候可能会变成有用的。正确的思路是，假设现在已经有k个球星的名字，那么要使球星的名字

达到k+1个平均需要买多少瓶饮料？这是很容易计算的，是 $\frac{n}{n-k}$ 。所以我们从没

有球星的名字开始，直到把所有的球星名字都凑齐，平均需要的饮料数(E)就可以计算出来：

$$E = n \left(\frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} \right)$$

由于题目的数据规模并不大，所以可以直接使用PASCAL的Comp或Int64(C/C++的long long)进行计算。而题目要求得到即约分数，只要在计算的时候使用分数并注意约分就可以了。

方法2 动态规划

动态规划是一种应用范围很广的方法，由于概率和期望具有前面提到过的一些性质（特别是期望的定义以及期望的“线性”性质），使我们可以在概率和期望之间建立一定的递推关系，这样就可以通过动态规划来解决一些概率问题。

与其他方面的动态规划一样，合理的选择状态以及高效的状态转移方程是应用这种方法的关键，而状态的选择在这类问题中尤为重要。选择合适的状态不仅

¹ 题目来源 SHTSC 2002 Day 1 Prob 2

可以提高效率,而且可以保证动态规划所必须的无后效性。而动态规划的各种优化方法也可以应用。

概率和期望的最值问题也往往使用动态规划的方法来解决。

例二 多米诺骨牌²

你试图把一些多米诺骨牌排成直线,然后推倒它们。但是如果你在放骨牌的时候不小心把刚放的骨牌碰倒了,它就会把相临的一串骨牌全都碰倒,而你的工作也被部分的破坏了。

比如你已经把骨牌摆成了 DD_DxDDD_D 的形状,而想要在 x 这个位置再放一块骨牌。它可能会把左边的一块骨牌或右边的三块骨牌碰倒,而你将不得不重新摆放这些骨牌。

这种失误是无法避免的,但是你可以应用一种特殊的放骨牌方法来使骨牌更多的向一个方向倒下。

给出你要摆放的骨牌数目,以及放骨牌时它向左和向右倒的概率,计算你为完成任务摆放的骨牌数目的平均数。假设你使用了最佳的摆放策略。

输入将包含至多 100 个测试点,每个测试点占一行,包含需要摆放的骨牌数目 n ($1 \leq n \leq 1000$), 以及两个非负实数 Pl , Pr , 表示骨牌向左和向右倒的概率。保证 $1 < Pl + Pr \leq 0.5$ 。

最后一个测试点包含一个数 0。对于每个测试点输出题目要求的数目,保留两位小数。

分析

首先应该明确怎样找到最佳的摆放策略。我们可以考虑在位置 i 放最后一块骨牌。显然, i 前面的 $i-1$ 块骨牌和 i 后面的 $n-i$ 块骨牌是互不影响的。所以我们假设摆放 $i-1$ 块骨牌需要的次数平均是(或说期望是) E_1 , 摆放 $n-i$ 块骨牌需要的次数平均是 E_2 。那么我们摆放了这两段之后,就要把最后一块放上。这时如果把左边的碰倒了,就只好重新摆放。右边的也是同样的道理。所以需要摆放的平均值(E)是:

$$E = \frac{1-Pr}{1-Pl-Pr} E_1 + \frac{1-Pl}{1-Pl-Pr} E_2 + \frac{1}{1-Pl-Pr}$$

这个式子的推导并不困难,方法之一就是应用方程的思想(参见后面介绍的概率一期望系统)。

既然得到了这个式子,我们就可以通过动态规划来得到最优的摆放方案。设 E_i 是摆放 i 块骨牌所需要的最少期望次数,那么状态转移方程是:

$$E_i = \min \left\{ \frac{1-Pr}{1-Pl-Pr} E_k + \frac{1-Pl}{1-Pl-Pr} E_{i-1-k} + \frac{1}{1-Pl-Pr} \right\} (0 \leq k \leq i-1)$$

这样就得到了一个 $O(n^2)$ 的算法。根据题目中的数据规模,最大的运算量是 $100 \times 1000^2 = 10^8$, 虽然可以忍受,但是还是比较慢的,如果数据稍大一点就容易超时。

这就需要我们对动态规划进行优化。

这是一个 1D/1D 的动态规划,我们自然希望得到 $O(n)$ 的算法,而这种优化一般都是通过动态规划的方程性质得到的。

观察动态规划的方程,我们可以发现,当 k 从 0 变化到 $i-1$ 时第一项是不断增大的,第二项是不断减小的,第三项则是一个常数。因此整个函数一定是单峰的,这样就可以通过二分的方法进行优化,复杂度已经降到了 $O(n \log n)$ 。而事实

² 题目来源 UVA 10529

上, E 这个数列不但是单增的, 而且是凹的 (如果 $PIPr=0$ 就不凹也不凸, 但是这不影响这里的讨论), 通过这个性质我们还可以证明决策使用的 k 一定是不减的 (证明很简单, 略去)。这样通过记录上一次决策使用的 k , 就得到了一个 (均摊的) $O(n)$ 的算法。

方法 3 迭代

动态规划要求问题无后效性, 而如果问题有不可避免的后效性, 动态规划就无能为力了。这时我们可以采用迭代的方法来进行计算。

当然, 迭代也不是万能的, 它要求问题有收敛性而且收敛的速度要足够快, 而且要求的结果精度不要太高。对于同一规模的不同的输入, 迭代法的效率可能会有很大的改变, 所以这种方法有可能因为遇到比较坏的数据而失效。

此外, 迭代法也未必是要解决有后效性的问题, 只要问题有收敛性, 迭代都可以起到一定的作用, 下面这道例题就没有后效性, 但是由于动态规划的时间复杂度过高, 所以采用了一种动态规划和迭代混合的方法来解决。

例三 巧克力³

2100 年, ACM 牌巧克力将风靡全球。

“绿的, 橘红的, 棕色的, 红的...”, 彩色的糖衣可能是 ACM 巧克力最吸引人的地方。你一共见过多少种颜色? 现在, 据说 ACM 公司从一个 24 种颜色的调色板中选择颜色来装饰他们的美味巧克力。

有一天, Sandy 用一大包有五种颜色的巧克力玩了一个游戏。每次他从包里拿出一粒巧克力并把它放在桌上。如果有桌上有两粒相同颜色的巧克力, 他就把他们吃掉。他惊奇的发现大多数时候桌上都有 2 到 3 粒巧克力。

如果 ACM 巧克力有 $C(1 \leq C \leq 100)$ 种颜色, 在拿出了 $N(1 \leq N \leq 1000000)$ 粒巧克力之后在桌上恰有 $M(1 \leq M \leq 1000000)$ 粒的概率是多少?

分析

如果 N 不是那么大的话, 我们是很容易用动态规划来解决此题, 状态转移方程就是

$$P_{i+1,k} = P_{i,k-1} * (C-k+1)/C + P_{i,k+1} * (k+1)/C$$

其中 $P_{i,k}$ 表示拿出了 i 粒巧克力后桌上剩余 M 粒的概率 (当然还要考虑一些边界情况)。但是现在 N 可以达到 1000000, 如果直接动态规划肯定是要超时的。

这个题的标准算法是使用生成函数。也就是说把“桌上有 m 块巧克力”转化成“有 m 种巧克力取了奇数块, 其余的都取偶数块的取法”。所以就可以列出

生成函数是 $\frac{(e^x - e^{-x})^m (e^x + e^{-x})^{c-m}}{2^c}$, 所以总的取法数就是 x^n 的系数乘以 $n!$ 和 $C(c, m)$,

而概率就是总的取法数除以 c^n , 然后通过进一步的代数分析来化简解决。这种方法当然是很优秀的, 复杂度是 $O(c^2)$ 。但是由于这道题的精度要求很低, 迭代的方法也是可以达到目的的, 而且复杂度也接近 $O(c^2)$ 。

这道题里不会出现极大或极小的概率, 一般来说这种情况下的收敛是比较快的。我们可以不断的计算 P 的值, 当它的变化不足以影响结果时就停止计算。当然这道题里的收敛是分奇偶的 (显然桌上剩余的巧克力数和拿出的巧克力数是同奇偶的), 所以不能比较 P_i 和 P_{i-1} , 而要比 P_i 和 P_{i-2} , 只要看到 P_i 和 P_{i-2} 差距小

³ 题目来源 ZJU Online Judge 1363

于一个定值（比如 $1e-5$ ），而且 i 和 N 同奇偶，就可以停止动态规划，因为此时继续规划下去所产生的不同已经不可能影响到要输出的结果。经过实验发现，最大的数据也只在几百次迭代中就稳定了，这样就将效率大大提高，满足了题目的要求。

方法 4 概率—期望系统

这个是我自创的，或许是由于不是很难吧，这个东西我在资料中没有见到过。其实这就是方程的思想在概率和期望问题中的一个应用。

概率—期望系统的定义

概率—期望系统是一个带权的有向图。这个图中的点代表一个事件，而如果点 A 与点 B 之间有一条权为 p 的边，就表示 A 发生后， B 紧接着发生的概率是 p 。初始的时候，有一个点（叫做初始点）代表的事件发生了，其他事件根据概率依次发生，每次只发生一个。求其他各个事件发生次数的期望。记时间 A 的发生次数期望为 E_A ， A 到 B 的边权为 P_{AB}

一些限制

➤ 对任意的 AB ， $P_{AB} \leq 1$

➤ 对于任意点 A ， $\sum_{(A,B) \in E} P_{AB} \leq 1$ ，且对于系统中的所有点，至少有一个点使等

号不成立。如果等号都成立的话这些事件将无穷无尽的发生下去，而概率—期望系统则变得没有意义（此时期望或者是无穷大，或者是 0）。

➤ 不能有指向初始点的边，这是因为求解时我们把初始顶点的概率设为 1。但是如果真的有这样的边，可以添加一个假点作为初始点，这个假点到真正的初始点有一条概率为 1 的边。

概率—期望系统的求解

我们首先来看一种最简单的概率—期望系统：有向无环图的概率—期望系统。这种系统是很简单的，因为它没有后效性，所以可以通过动态规划的方法在 $O(E)$ 的时间内解决。许多使用动态规划解决的概率—期望问题都是基于这类系统的。比如 ZJU1022 Parallel Expectation 就是这样的。

但是在有些问题中（比如下面的例 4），我们需要解决更一般的概率—期望系统。这时图中含有圈，因而造成了后效性。我们当然可以用迭代法，而如果设第 i 次迭代时 A 的期望用 $E_{i,A}$ 来表示，迭代的方程就是

$$E_{i+1,A} = \sum_{(B,A) \in E} P_{BA} E_{i,B}$$

通过若干次的迭代，就可以达到我们需要的精度。但是这个方法的效率是不稳定的，考虑这样一个例子： A 是起始点， $P_{AB}=1$ ， $P_{BC}=1$ ， $P_{CB}=0.99$ ，这个例子的解是 $E_A=1$ ， $E_B=100$ ， $E_C=100$ ，但是如果用迭代的话需要很多次才能达到这个结果。而如果 $P_{CB}=0.9999$ ，那么迭代法的速度就更加缓慢。

如何避免这种情况的出现呢？我们当然不能限制数据，而应该寻找更稳定的方法。考虑刚才迭代的过程，我们发现，在迭代的终点，近似的有

$E_A = \sum_{(B,A) \in E} P_{BA} E_B$ ，这是一个方程，而这个方程的解其实就是我们要求的解。

我们只要把这个方程解出来，就可以得到精确的结果，而刚才说的迭代过程，实际上就是这个方程的求解过程（线性方程组的一种解法——Jacobi 法就是这样做的）。而求线性方程组的解，我们更常用的稳定算法是高斯消元法，完全可以在这里使用。这样就得到了一种稳定而精确的解法：

首先根据概率一期望系统建立方程组，然后用高斯消元法去解，得到的结果就是我们要求的期望。这种算法的时间复杂度是 $O(n^3)$ 。

当然这个复杂度是不很理想的，但是对于解方程组，我们没有更简单而高效的方法，所以这个算法还是比较可取的。而迭代法也不能忽视，在概率不会出现极大或极小，而且边又不多时，迭代法的效率往往会高于高斯消元。

此外，即使概率一期望系统中有环，也并不一定就不能使用动态规划来解决，当环之间满足一个偏序的时候，我们仍然可以使用等比数列求和的方法来进行处理，从而达到更好的复杂度。

例四 简单游戏⁴

所谓简单游戏，相信大家小时候都玩过，就是那种掷出骰子，然后按掷的步数走的游戏。现在有一个 $n(1 \leq n \leq 100)$ 个格子的游戏，一些格子上有指令。指令分成若干种，如下：

- 0——空指令
- -1——陷阱，到了这里后要掷出六才能继续向前，注意不是向前六步，而是要再掷一次决定步数。
- -2——停一次
- 其它数字——转移指令，走到数字所代表的格子

走到陷阱中是很难出来的，因此大家都不希望走到陷阱里。玩一次游戏走到陷阱里的平均次数到底是多少呢？这个问题将由你来解决。

输入：第一行 n ，表示游戏的格数。第二行有 n 个数，表示每个格子的指令（第一个和最后一个都没有指令）。注意如果某次走到的位置达到或超过最后一个格子，都表示游戏结束。

输出：走到陷阱里的平均次数。保留 3 位小数，请尽量使用 `extended` 类型。

样例：

输入	输出
4 0 -1 2 0	0.400

解释：第一次走到陷阱里，概率为 $1/3$ ，第二次走到陷阱里，概率为 $(1/3) \cdot (1/6)$ ，第三次 $(1/3) \cdot (1/6)^2$ ，……

由等比数列的求和公式中令 $n = \infty$ ，可得概率之和为 $(1/3) / (1 - (1/6)) = 2/5 = 0.4$

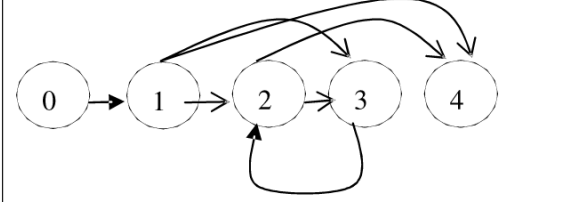
分析

如果按照题目中的那个“解释”的思路走，就不免中了圈套。题目中的转向指令可以造成非常复杂的圈套圈的情况，用求极限的思想很难解决。

有了概率一期望系统，这个问题就并不困难了。我们只要构造一个图就可以。把到达每个格子作为一个事件，每个事件如果没有转向指令就向后面的六个格子

⁴ 题目来源 原创

各连出一条概率为 $1/6$ 的边，如果有转向指令就向转向的那个格连一条概率为 1 的边。的由于题目中没有限制第一个格子不会有转移指令指向，所以我们不能简单的把第一个格子作为起始点，而应该做一个假起点，叫做 0 点，然后让 0 到 1 有一条概率为 1 的边就可以。例如样例中的游戏对应的图和方程就是：

 <p>其中三角形箭头表示概率是 1，普通箭头表示概率是 $1/6$。</p>	<p>方程组：</p> $E_1 = 1$ $E_2 = \frac{1}{6} E_1 + E_3$ $E_3 = \frac{1}{6} E_1 + \frac{1}{6} E_2$ <p>解得 $E_2 = 0.4$，这就是我们需要的结果。</p>
--	--

对于任何一个游戏，我们都可以按这种方法建立一个概率—期望系统，剩下的问题就是应用高斯消元法来解决就可以了。

当然我们不能忘记迭代法。但是容易看出，迭代法在这道题中可能遇到特殊设计的数据（一个典型的数据就是 5 个转向指令连成一排，这种结构只要重复几次就会造成答案非常大，可以达到 10^5 以上），对于这种数据，迭代是无法出解的。因此，本题选择高斯消元是比较好的方法。

总结

本文对有关概率和期望的问题进行了一些研究，介绍了直接计算，动态规划，迭代以及概率—期望系统四种常用方法。在实际解题时，只有具备扎实的数学基础，灵活运用这些方法，才能顺利的解决各种各样的概率问题。