# 概率和期望

张 捷钧
zhangjiejun1992@gmail.com

# 1. 概率期望基础

# 概率

- 丢硬币，正面朝上和反面朝上的概率为1/2
- 丢一个色子，每一面出现的概率相等，1/6
- 当两件事独立时，同时发生的概率，为两件事发生概率的乘积。独立性通常都是很显然的，比如说，同时丢两个色子得到两个1的概率为1/6*1/6 = 1/36
- $P(AB) = P(A)P(B)$

# 概率计算：组合计数

- 组合计数即将符合条件的事件全部列举出来（或者通过排列组合算出个数），除以事件的总个数
- A抛一枚硬币一次，B抛一枚硬币两次，B比A得到更多正面的概率是多少？

# 概率计算：组合计数

| A | B1 | B2 |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| 1 | 1 | 1 |

1代表正面朝上，红色行满足“B拥有更多正面朝上”，故概率为4/8，即1/2

# 概率计算：枚举法

- 计算两个色子，点数和为8的概率
  - 当色子1点数为1时，不可能，0 * 1/6
  - 当色子1点数为2时，色子2必须为6，1/6 * 1/6
  - …
  - 当色子1点数为6时，色子2必须为2，1/6 * 1/6
- 上面共有1个0，5个1/6 * 1/6，相加就得到答案：5/36
- 这个方法也叫做“全概率公式”

# SRM 536, DIV 1, Level 2 RollingDiceDivOne

- 有n个奇怪的色子，每个色子有d[i]面。点数从1到d[i]，每一面等概率出现。
- 输入 n, d[i]
- 输出 最有可能出现的点数和，如有多个，输出最小的
- 范围 n <= 50, d[i] <= 10^9

# RollingDiceDivOne 参考解答

- d[] = {5, 3, 2}
- 手算一下：

| 点数和 | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
|---|---|---|---|---|---|---|---|---|---|---|
| {5} | 1/5 | 1/5 | 1/5 | 1/5 | 1/5 | 0 | 0 | 0 | 0 | 0 |
| {5,3} | 0 | 1/15 | 2/15 | 3/15 | 3/15 | 3/15 | 2/15 | 1/15 | 0 | 0 |
| {5,3,2} | 0 | 0 | 1/30 | 3/30 | 5/30 | 6/30 | 6/30 | 5/30 | 3/30 | 1/30 |

# RollingDiceDivOne 参考解答

- 可观察到如下规律
  - 有一段概率最大的区间
  - 这一段区间左边递增，右边递减
- 从计算的过程来看，根据以下信息
  - 上一行的最大概率区间的开始位置、长度
  - 本行的色子点数
- 就可以计算出下一行的区间开始位置、长度
- AC

# 期望

- 简单的来说，期望就是加权平均
- 例如，一个6面的色子，期望得到的点数为

$$1\quad\frac{1}{6}+2\quad\frac{1}{6}+\ldots+6\quad\frac{1}{6}=3.5$$

# 随机变量

- 随机变量，就是一个变量
- 例如，定义随机变量X为丢色子后朝上的点数，X的期望记为E[X]，则E[X]=3.5
- 随机变量X取值k的概率记为P(X=k)，所以P(X=1)=P(X=2)=…= P(X=6)=1/6
- 形式化的说，随机变量X的期望 $P(X==i) \quad i$
- 虽然说X是个变量，但是E[X]就是个常量了

# SRM 518, DIV 2, Level 3 CoinReversing

- 有N个硬币在桌上排成一排，正面朝上。给出K个指令，第i个指令的内容是“在桌上随机选择a[i]个硬币，翻面”。计算最终正面朝上硬币的期望个数。
- 范围 N <= 1000, K <= 50, a[i] <= N

# CoinReversing 参考解答

- 期望有很多神奇的性质，例如本题的解答
- 第一步之后，正面朝上的个数是确定的N-a[0]。
- 执行指令a[1]的时候，有a[0]个正面朝下，N-a[0]个正面朝上。单个硬币被挑中的概率

$$p = \binom{n-1}{a[1]-1} / \binom{n}{a[1]} = \frac{a[1]}{n}$$

- 那么，执行完a[1]的时候，正面朝上的期望就是 p*a[0] + (1-p) * (N - a[0])

# CoinReversing 参考解答

- 那么神奇的性质是什么呢
- 把算出来的执行完指令a[1]之后，正面朝上的硬币期望个数当作硬币朝上的实际个数，用同样的方法算出执行完a[2]的期望个数
- 5行代码，AC

# SGU 508. Black-white Balls

- 在一个背包里面有白球和黑球一共n个。黑球的数量等概率为0, 1, 2, …, n个。我们现在随机取出L个球，其中有L1个黑球，L2个白球（L1+L2 == L）。计算有k个黑球的概率，k=0, 1, 2, …, n
- 范围 n <= 50, 0 <= L1 <= n, 0 <= L1 + L2 <= n

# 条件概率

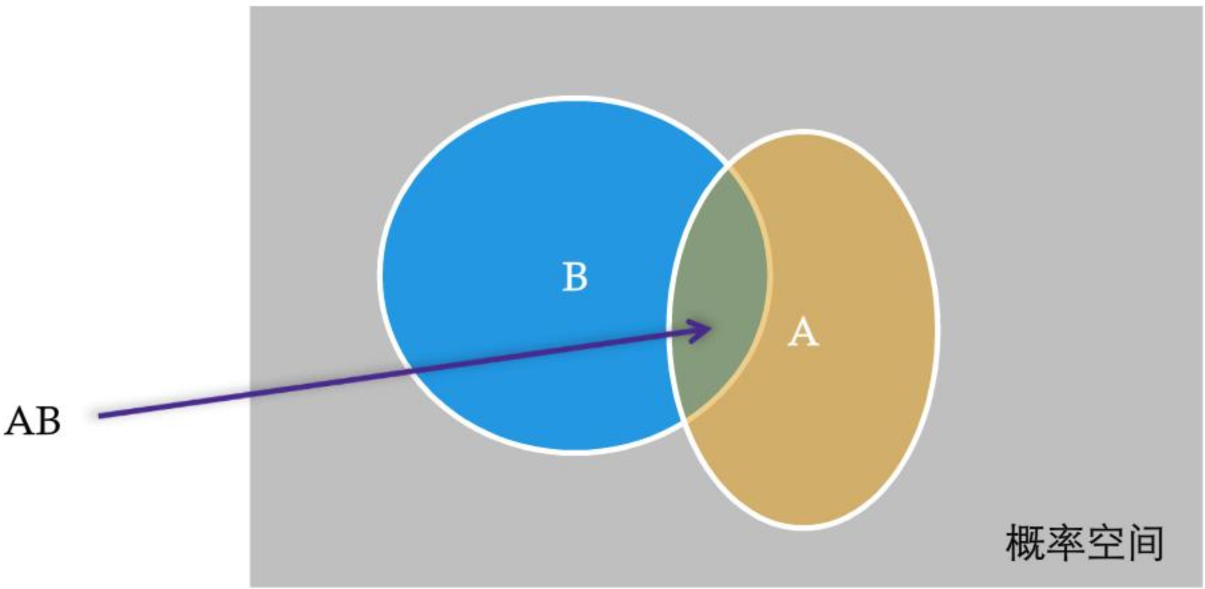
- 已知事件A发生，事件B发生的概率记作$P(B|A)$，此为条件概率，下面假设$P(A)$和$P(B)$均不等于0
- 使用枚举法计算概率的时候，就是条件概率的简单应用
- $P(AB) = P(A|B)P(B) = P(B|A)P(A)$
- $P(A|B)=P(A)$等价于A、B独立

# 条件概率

- 丢两个色子
  - A为色子1上的点数
  - B为色子2上的点数
- 若已知A+B ≤ 5，计算A=2的概率
- 右表中A+B≤5共10种情况，A=2占3种，P(A=2 | A+B≤5)=3/10
  - P(A+B≤5且A=2)=3/36
  - P(A+B≤5)=10/36

| + | A1 | A2 | A3 | A4 | A5 | A6 |
|---|---|---|---|---|---|---|
| B1 | 2 | 3 | 4 | 5 | 6 | 7 |
| B2 | 3 | 4 | 5 | 6 | 7 | 8 |
| B3 | 4 | 5 | 6 | 7 | 8 | 9 |
| B4 | 5 | 6 | 7 | 8 | 9 | 10 |
| B5 | 6 | 7 | 8 | 9 | 10 | 11 |
| B6 | 7 | 8 | 9 | 10 | 11 | 12 |

# 条件概率



# Bayes公式

- 以下即为Bayes公式，将P(A|B)转化为P(B|A)的计算。展开到哪一步最为适用还需看情况而定。

$$
\begin{aligned}
P(A|B) &= \frac{P(AB)}{P(B)} \\
&= \frac{P(B|A)P(A)}{P(B)} \\
&= \frac{P(B|A)P(A)}{P(B|A)P(A) + P(B|\bar{A})P(\bar{A})}
\end{aligned}
$$

# SGU 508 参考解答

- 定义事件
  - A: 有k个黑球
  - B: 在随机抽取的L个球中有L1个黑球
- 题目要计算的是P(A|B)，应用Bayes公式到第2步
  - P(A|B)=P(B|A)P(A) / P(B)
  - P(B) = 1 / (L + 1)，P(A) = 1 / (N + 1)
  - P(B|A) = C(K,L1)*C(N-K,L2) / C(N, L)
- ac

# Tech 1: 动态规划

对于概率 / 期望类的动态规划，设计状态的时候通常会设**f[S]=处于状态S时完成的概率**，或者**f[S]=处于状态S时到达终点的期望次数**，而不是f[S]=到达状态S的概率

# HDU 4405. Aeroplane Chess

- 飞行棋，一共有N+1格，从0到N标号。一开始处于0，每一步随机走1~6步。一些格子可以“飞”，踩到x[i]上立刻飞到y[i]，N ≥ y[i] > x[i]，飞不算作一步。到达N号格子即算到达终点，不论是否还有剩余步数。
- 期望多少步能走到终点

# Aeroplane Chess 参考解答

- 设计状态
  - NO f[i] – 走到第i格的期望步数
  - YES f[i] – 现在处于第i格，走到终点的
- 状态转移：
  - 若i可以飞到j，则f[i] = f[j]
  - 否则，f[i] = (1+f[i+1])/6 + (1+f[i+2])/6+ … + (1+f[i + 6])/6
  - Nice & Easy, AC
- 考虑一下，若不幸选择了第一种状态，如何转移？

# HDU 4336. Card Collector

- 很久以前小浣熊干脆面中有水浒108将的卡片，集齐这些卡片能够兑换大奖。
- 现在有一种零食中包含N种卡片，包含第i种卡片的概率为p[i]。零食中可能没有卡片。计算为了集齐所有的卡片，期望要买多少包零食?
- 范围 $N <= 20$

# Card Collector 参考解答

- 状态压缩，设S表示当前已经收集到的卡片集合，f[S]表示
  - NO! 收集到这些卡片的期望步数
  - YES 在这种情况下，集齐卡片期望还要买多少包零食
- 状态转移，f[S]
  - 设Q为还没收集到的卡片它们出现的概率和
  - $f[S] = (1 + f[S]) * (1 - Q) + (1 + f[S|c1]) * p[c1] + (1 + f[S|c2]) * p[c2] + \ldots$
- 转移的本质即枚举下一包零食收集到哪一张卡片

# Codeforces 148D. Bag of mice

- 一个包里面有b只黑老鼠和w只白老鼠，公主和龙轮流从这个包里随机抓一只出来，公主先来。龙抓老鼠的动作比较粗鲁，所以他把一只老鼠抓出来之后，包里面剩下的老鼠会恐慌并有随机的一只逃跑。公主的动作温柔所以不会出现这种情况。
- 谁先抓出来一只白老鼠，谁就获胜。若谁也没有抓到白老鼠那么算公主输。逃跑的老鼠不算
- 范围 0<= w, b <= 1000

# Bag of mice 参考解答

- f[w][b] – 有w白老鼠，b黑老鼠，轮到公主行动，公主胜利的概率
- g[w][b] – 有w白老鼠，b黑老鼠，轮到龙行动，公主胜利的概率
- f[w][b]=
  - w/(w+b) （抓到白老鼠）
  - b/(w+b)*g[w][b-1]（抓到黑老鼠）
- g[w][b]=
  - b/(w+b) * w/(w+b-1) * f[w-1][b-1] （抓到黑老鼠，跑掉一只白老鼠）
  - b/(w+b) * (b-1)/(w+b-1) * f[w][b-2] （抓到黑老鼠，跑掉一只黑老鼠）
  - 抓到白老鼠的话就是0

# Tech 2: 高斯消元/迭代

状态之间有环的时候使用

# HDU 2262. Where is the canteen

- 在一个n*m的矩阵中，随机上下左右移动。地图中有三种格子
  - 空地
  - 障碍
  - 出口，可能有不止一个出口
- 给定起点，到达出口期望要走多少步？
- 范围 n, m <= 15

# HDU 2262 参考解答

- 还是f[x][y]，表示当前处于(x, y)坐标，走到终点的期望步数
  - 若(x, y)是出口则f[x][y] = 0
  - 否则f[x][y] = 1+ (f[x-1][y]+f[x+1][y]+f[x][y-1]+f[x][y+1])/4
- 状态之间存在环，无法按特定的顺序计算每一个f
  - 此时将每个f[x][y]视作未知数，构成方程组，高斯消元。
  - 迭代，无视状态之间的环直接递推，重复多次直到f稳定下来。

# 练习题

- SGU 422,465
- ZOJ 3302
- ZOJ 3200
- UVA 12013 （难）
- Codeforces 148D. Bag of mice
- HDU 2262. Where is the canteen
- SRM 536, DIV 1, Level 2 RollingDiceDivOne
- SRM 518, DIV 2, Level 3 CoinReversing
- SGU 508. Black-white Balls
- HDU 4405. Aeroplane Chess
- HDU 4336. Card Collector