概率&期望DP

PKU Hzyoi

01 概率&期望

概率的起源

- 17世纪中叶盛行掷骰子，当时人们认为这两个游戏是等价的：
  1. 扔4次骰子，若出现6，则庄家赢；
  2. 扔24**对**骰子，若出现两个6，则庄家赢。
- 显然，出现两个6的概率是出现一个6的$\frac{1}{6}$，所以将次数扩大6倍（$4 \to 24$），两个游戏等价。

- 然而现在的小学生也知道这两个游戏的胜率：
  1. $1-(\frac{5}{6})^4 \approx 0.52$，庄家占优；
  2. $1-(\frac{35}{36})^{24} \approx 0.49$，玩家占优。
- 后来人们去请教数学家帕斯卡，这个问题的解决直接推动了概率论的发展。

概率

- 定义样本空间S是基本事件的集合
- 每个基本时间就是一个实验可能的结果
- 举个例子：对于抛两枚不同的硬币，正面为H，反面为T
- 那么$S=\{HH,HT,TH,TT\}$
- 事件是样本空间S的一个子集，如果两个事件$A\cup B=\emptyset$，则它们是互斥的
- 定义基本事件$s\in S$，可知所有基本事件都是互斥的，既它们不是互相独立的

概率论公理

- 对于任何事件A，$Pr\{A\} \geq 0$
- $Pr\{S\} = 1$
- 对于两个互质事件A与B，有$Pr\{A \cup B\} = Pr\{A\} + Pr\{B\}$，对于任意的两两互斥的事件序列$A_1, A_2, A_3, \ldots$,有$\mathrm{Pr}\{\bigcup_i A_i\} = \sum_i Pr\{A_i\}$
- $\mathrm{Pr}\{\emptyset\} = 0$
- 用$\overline{A}$表示事件$S - A$，有$Pr\{\overline{A}\} = 1 - Pr\{A\}$
- 对于两个任意事件A和B，$\mathrm{Pr}\{A \cup B\} = Pr\{A\} + Pr\{B\} - Pr\{A \cap B\} \leq Pr\{A\} + Pr\{B\}$

离散概率分布

- 如果一个概率分布在有限或无限可数的样本空间上，则该概率分布是离散的
- S是样本空间，A是任意事件，则有$Pr\{A\} = \sum_{s \in A} Pr\{s\}$
- 如果S是有限的，对于每个基本事件$s \in S$，有$Pr\{s\} = 1/|S|$
- 以以上方法得到的概率分布是S上的均匀概率分布，这样的情况才能用在“S中随机选择一个元素描述”
- 考虑抛掷一枚均匀硬币n次，事件$A = \{k$枚硬币正面朝上，$n - k$枚反面朝上$\}$
- A的大小$|A| = \binom{n}{k}$，事件A的概率是$Pr\{A\} = \binom{n}{k}/2^n$

条件概率与独立

- 在B条件下A发生的概率写作$Pr\{A|B\}$
- $Pr\{A|B\} = \frac{Pr\{A \cap B\}}{Pr\{B\}}$
- 当A和B是独立的
- 所以B不会影响A的情况，$Pr\{A|B\} = Pr\{A\}$
- 所以$Pr\{A \cap B\} = Pr\{A\}Pr\{B\}$
- 那么对于所有两两独立的A_i
- $\Pr\{A_1 \cap A_2 \dots \cap A_n\} = \prod_{k=1}^{n} Pr\{A_k\}$

贝叶斯定理

- 对于两个概率不为0的事件A和B
- $Pr\{A \cap B\} = Pr\{B\}Pr\{A|B\} = Pr\{A\}Pr\{B|A\}$
- 那么$Pr\{A|B\} = \frac{Pr\{A\}Pr\{B|A\}}{Pr\{B\}}$

生日悖论

- 如果一个房间有23个人，则至少两个人生日在同一天的概率大于50%。
- $1-\frac{365}{365}\times\frac{364}{365}\times\frac{363}{365}\times\ldots\times\frac{365-23+1}{365}\approx 0.51$
- 这就是Hash为什么那么容易被卡的原因……

三门问题

- 有三扇门，其中一扇后面是一辆车，其余两扇后面是山羊。
- 你先选一扇门，主持人会在其余两扇门选一扇后面是山羊的，将这扇门打开。
- 你可以改选剩下的另一扇门。
- 请问改选能够提高抽中车的概率？

- 请问改选能够提高抽中车的概率？
- 可以！
  - 有$\frac{2}{3}$的概率选中山羊门，改选就能win。
  - 有$\frac{1}{3}$的概率选中车门，改选lose。
- 改选胜率为$\frac{2}{3}$，不改选胜率为$\frac{1}{3}$。

淘汰赛

- 16名水平两两不同的选手参加淘汰赛，每轮随机抽签安排对阵。
- 依次进行1/8决赛、1/4决赛、半决赛、三四名决赛、冠亚军决赛。
- 假设强者总是能战胜弱者，请问冠亚季军恰好是实际水平第一、二、三名的概率是多少呢？

$$\frac{2}{3} \times \frac{6}{7} \times \frac{4}{6} \times \frac{14}{15} \times \frac{12}{14} \approx 0.3$$

所以一场考试没考好不能说明什么啊！

期望

- 期望值是随机变量的值乘以其概率的总和。
- 记随机变量X可以取到的值为$x_1, x_2, ..., x_m$，概率分别为$p_1, p_2, ..., p_m$。
- 随机变量X的期望值为：

$$E(x) = \sum_{i=1}^{m} x_i p_i$$

- 注意X的取值可能不可列，比如说X在$[0, 1]$中全体实数等概率随机。

期望的线性性质

- $E[X+Y]=E[X]+E[Y]$
- 即和的期望为期望的和
- $E[aX]=\sum_x ax*Pr\{X=x\}=a\sum_x x*Pr\{X=x\}=aE[X]$
- 那么$E[aX+Y]=aE[X]+E[Y]$

乘法原理

- 当两个随机变量X和Y独立且期望有定义时
- $E[XY] = \sum_x \sum_y xy * Pr\{X = x且Y = y\} = \sum_x \sum_y xy * Pr\{X = x\}Pr\{Y = y\} = \left(\sum_x x * Pr\{X = x\}\right)\left(\sum_y y * Pr\{Y = y\}\right) = E[X] * E[Y]$
- 当n个随机变量$X_1, X_2, \ldots, X_n$互相独立时
- $E[X_1 X_2 \ldots X_n] = \prod_{k=1}^{n} E[X_k]$

重点

- 当随机变量X可在自然数集$N = \{0,1,2,\dots\}$中取值时
- $E[X] = \sum_{i=0}^{\infty} i * Pr\{X = i\} = \sum_{i=0}^{\infty} i(Pr\{X \geq i\} - Pr\{X \geq i + 1\})$
- $= \sum_{i=1}^{\infty} Pr\{X \geq i\}$
- 每项$Pr\{X \geq i\}$被加了i次，被减了$i - 1$次，所以它被加了一次

经典

- 有$1,2,...,n$一共n个数。
- 每轮在1到n中随机选一个数，请问$1,2,...,n$都被选过的期望轮数。
- 期望线性性！
  - 和的期望 = 期望的和
- $Ans=\sum_{i=1}^{n}F(i)$，$F(i)$表示取出和前$i-1$个数不同的数期望要多少轮。
- $F(i)=\sum_{j=1}^{\infty}f(i,j)\times j$，$f(i,j)$表示第$j$次取出不同数的概率。
- $f(i,j)=(\frac{i-1}{n})^{j-1}\times\frac{n-i+1}{n}$.

- $Ans = \sum_{i=1}^{n} F(i)$.
- $F(i) = \sum_{j=1}^{\infty} f(i,j) \times j$.
- $f(i,j) = (\frac{i-1}{n})^{j-1} \times \frac{n-i+1}{n}$.

- 换种方式计算$F(i)$:
- $F(i) = \sum_{j=0}^{\infty} g(i,j)$，$g(i,j)$表示前$j$轮**还没有取出不同数的概率。**
- $g(i,j) = (\frac{i-1}{n})^{j}$.

$$
\begin{aligned}
F(i) &= \sum_{j=1}^{\infty}\left(\frac{i-1}{n}\right)^{j-1} \times \frac{n-i+1}{n} \times j \\
&= \sum_{j=0}^{\infty}\left(\frac{i-1}{n}\right)^{j} \\
&= \frac{n}{n-i+1}
\end{aligned}
$$

- $Ans = \sum_{i=1}^{n} \frac{n}{n-i+1}$。

51nod 13E B君的骗局

- 给出n个点m条边的图，你初始在S点，接着你会随机向相邻的点走过去。
- 给出3组3个点，如果你在两组都经历了2个点以上，则游戏结束。
- 求期望多少步游戏结束。
- 效率：$n \le 50$。

- 暴力：
  - 令$F[x][s]$表示当前你在x点，s为一个二进制数，表示9个点中经历了哪些点，$F[x][s]$即为这种情况期望走几步结束。
  - 直接对其高斯消元，效率为$O((2^9 \times n)^3)$。
- 容易发现，每次转移s都会“或”上一个点，即s不会变小。
- 于是可以分层高斯消元：
  - 从大到小计算状态为s的所有$F[x][s]$。
  - 对于$\tilde{s} > s$的$F[\tilde{x}][\tilde{s}]$都已经被计算出来了；
  - 高斯消元时可以直接作为常数项。
  - 故效率为$O(2^9 \times N^3)$。

- 令$f_{u,i}$表示从起点走i步到达u点的概率，P_u为起点走到u的概率，E_u为起点走到u的期望。

$$
\begin{aligned}
P_u &= \sum_{i=0}^{\infty} f_{u,i} \\
&= f_{u,0} + \sum_{v} P(v \to u) \sum_{i=0} f(v,i) \\
&= f_{u,0} + \sum_{v} P(v \to u) P(v)
\end{aligned}
$$

$$
\begin{aligned}
E_u &= \sum_{i=0}^{\infty} i \times f_{u,i} \\
&= \sum_{i=1}^{\infty} i \times f_{u,i} \\
&= \sum_{v} P(v \to u) \sum_{i=0}^{\infty} (i+1) \times f_{v,i} \\
&= \sum_{v} P(v \to u)(P_v + E_v)
\end{aligned}
$$

- 这里的$f_{u,0}$就表示它的初始状态，可以理解为常数项。
- 这里还有一些常识，如果是求u到终点的概率，则P_u恒为1，所以不需要得到$P(u)$。

02 DP例题

[POJ Monthly, POJ 2151] Check the difficulty of problems

- 出题人有N道题目，有M支队伍
- 给出每支队伍做出每道题目的概率，求每一支至少做出一道题目且至少有一支队伍做出K道题目的概率。

- 如何求总概率？P=1-P(有至少一支队伍没有做出题目)-P(所有队伍都做出1~K-1道题目)。
- 如何求有至少一支队伍没有做出题目的概率？P=1-P(所有队伍都做出题目的概率)
- 如何求每支队伍做出题目的概率？
- f[i][j][k]表示第i支队伍做了前j道题目，做出了其中k道题目。

[Codeforces698C] LRU

- 有一个容器，每次可能从给定集合中选出一个数
- 如果它已经在容器中，只更新它的放入时间，不把它放入；
- 否则将其放入容器中，同时如果容器中的数超过了容器的容量，最早被放入的数将被移除。
- 给出每个数字被选中的概率，求上述操作进行无限次后每个数分别出现在容器中的概率。

- 因为操作进行了无限次，所以可以认为最后的容器容量必定是满的。
- 考虑容器最后的状态，由此进行状态压缩型动态规划，考虑最后集合中数字的顺序，将一个数加入集合的概率等于它被选的概率，由于会选到集合里已有的数，所以要除以集合中已选数字不被选的概率。
- 注意特判被选中概率等于零的数。

[Codeforces601C] Kleofáš and the n-thlon

- M（M≤1000）个人参加N（N≤100）场比赛，所有人在每场比赛中的排名分别为1..N的排列，最后的总排名为把每个人每场比赛中的排名加起来，得到总分，将总分从小到大排序的结果。
- 总排名可能存在并列的现象。给出某个人在这N场比赛中的排名，求他的总排名期望。

- 设这个人的总分为K，并且一个另外的人达到总分为i的概率为P(i)，
- 根据那个经典变换，那么答案即为∑P(0..K-1)*(M-1)。
- 设立状态f[i][j]表示另外人第i轮后获得j分的概率，差分优化DP即可。

[Codeforces722E] Research Rover

- 给出一个N*M的方格阵，从(1,1)出发，到(N,M)结束，从(x,y)只能走到(x+1,y)或(x,y+1)。
- 方格阵上还有K个特殊点，初始时给出的分数t每经过一个特殊点就会变成ceil(t/2)。
- 求到(N,M)时得分的期望。
- 保证(1,1)和(N,M)不是特殊点。N,M≤100000， K≤2000， t≤1000000。

题解1：

- N,M的范围较大，不考虑将其记入动态规划状态。把(1,1)和(N,M)也看成特殊点，把所有的特殊点按照横坐标排序，$F[i][j]$表示从(1,1)出发到达第i个特殊点、已经经过j个特殊点的方案数。明显，j需考虑的取值范围只有log级别。
- 考虑转移，$F[i][j]=F[k][j-1]*E[k][i]$；$E[i][j]$表示第i个特殊点到第j个特殊点不经过任何其他特殊点的路径数，可以$O(K^3)$预处理，成为复杂度的瓶颈。
- 更换转移的思路，$F[i][j]=D[1][i]-\sum F[k][j]*D[k][i](k<i)-\sum F[i][l](l<j)$。（$D[i][j]$表示第i个特殊点到第j个特殊点的路径数，可以直接用组合数计算）
- 时间复杂度降为$O(K^2 \log t)$。

题解2：

- 考虑令dp[i][j]为从起点到点i之前经过了至少j个减分点，到点i的数学期望。此时转移方程为：
- dp[i][j] = Σ(dp[k][j - 1] - dp[k][j]) * D[k][i]。
- 这样相当于是前面恰好走过j个点 + 可能走过大于一个点的方式转移过来，这样可以保证计数的不重不漏。

CF865C

- 一个游戏关卡有n(n≤50)个任务，若在m秒内按顺序完成所有任务则算作通过当前关卡。
- 每个关卡有三个属性ai,bi,pi(1≤ai<bi≤100,80≤pi≤99)，表示有pi%的概率用ai完成任务i,有1−pi%的概率用bi秒完成任务i。每完成一个任务后可以选择继续下一个任务或重新开始当前关卡。
- 问最优策略下通过当前关卡的期望时间。

- 二分答案ans
- f[i][j]表示从i到最后，用了j秒，总时间的期望
- f[i][j]=pi*f[i+1][j+ai]+(1-pi)*f[i+1][j+bi]
- 若大于ans，则从新开始更优，故方程再和ans取min
- 最后f[0][0]=ans，则答案应该更大，若f[0][0]<ans，则答案应该更小

非常感谢您的观看

THANK YOU FOR YOUR WATCHING