

概率与期望

何中天

2018 年 6 月 8 日

概率的解释

频率学派和贝叶斯学派。有点哲学色彩。

概率的解释

频率学派和贝叶斯学派。有点哲学色彩。
不过我们今天不讨论它们，我们来看公理化定义。

样本空间、事件和概率

样本空间 S 是一个集合，它的元素称为基本事件。样本空间的一个子集被称为事件，根据定义，所有基本事件互斥。

样本空间、事件和概率

样本空间 S 是一个集合，它的元素称为基本事件。样本空间的一个子集被称为事件，根据定义，所有基本事件互斥。

概率：如果有一种事件到实数的映射 $P()$ ，满足：

1. 对任何事件 A , $P(A) \geq 0$
2. $P(S) = 1$
3. 对两个互斥事件, $P(A \cup B) = P(A) + P(B)$

则可称 $P(A)$ 为事件 A 的概率。上述三条称为概率公理。

事件的关系与运算

包含：对于事件A与事件B，如果事件A发生，则事件B一定发生，称事件B包含事件A（或事件A包含于事件B）。

相等。

事件A与事件B的并事件（和事件）：某事件发生当且仅当事件A发生或事件B发生，记作 $A \cup B$ 。

事件A与事件B的交事件（积事件）：某事件发生当且仅当事件A发生且事件B发生，记作 $A \cap B$ 或 AB 。

事件A与事件B互斥：A交B为不可能事件，即事件A与事件B在任何一次试验中并不会同时发生。

条件概率:

定义B的关于A的条件概率(事件A发生条件下, 事件B发生的概率)

$$P(B|A) = \frac{P(A \cap B)}{P(A)}$$

条件概率:

定义B的关于A的条件概率(事件A发生条件下, 事件B发生的概率)

$$P(B|A) = \frac{P(A \cap B)}{P(A)}$$

独立:

如果A, B满足 $P(A \cap B) = P(A)P(B)$, 称A, B独立, 可以推出 $P(A) = P(A|B)$

随机变量是随试验结果变化而变化的变量。有离散型和连续型两种，以离散型为例。

数学期望（或均值），即随机变量在概率意义下的平均值：

$$E(X) = \sum_{i=1}^n X_i p_i$$

随机变量是随试验结果变化而变化的变量。有离散型和连续型两种，以离散型为例。

数学期望（或均值），即随机变量在概率意义下的平均值：

$$E(X) = \sum_{i=1}^n X_i p_i$$

期望的线性性：对于任意随机变量X和Y，满足

$$E[aX + bY] = aE[X] + bE[Y]$$

期望的线性性是始终成立的，无论两随机变量是否独立。

[CF280C] Game on Tree

给出一棵含 n 个白点的有根树,每次随机选择一个还没有被染黑的节点,将这个节点和这个节点子树中的所有点染黑. 问期望操作多少次后所有点都被染黑. $n \leq 100000$

[CF280C] Game on Tree

给出一棵含 n 个白点的有根树,每次随机选择一个还没有被染黑的节点,将这个节点和这个节点子树中的所有点染黑. 问期望操作多少次后所有点都被染黑. $n \leq 100000$

根据期望的线性性, 计算出每个点被选择的期望次数, 然后相加就是整棵树的了。

所以得出对于点 x : $E(x) = 1/dep[x]$ 。

[bzoj3036]绿豆蛙的归宿

随着新版百度空间的下线，Blog宠物绿豆蛙完成了它的使命，去寻找它新的归宿。

给出一个有向无环的连通图，起点为1终点为N，每条边都有一个长度。绿豆蛙从起点出发，走向终点。

到达每一个顶点时，如果有K条离开该点的道路，绿豆蛙可以选择任意一条道路离开该点，并且走向每条路的概率为 $1/K$ 。

现在绿豆蛙想知道，从起点走到终点的所经过的路径总长度期望是多少？

$$N \leq 100000, M \leq 2 \cdot N$$

由期望的线性性可得：经过路径期望总长度 = \sum 每条边期望经过次数 * 边权。

因为是有向无环图，所以 每条边的期望经过次数 = 该边起点的期望经过次数 * 从该起点出发经过该路径的概率。

由期望的线性性可得：经过路径期望总长度 = \sum 每条边期望经过次数 * 边权。

因为是有向无环图，所以 每条边的期望经过次数 = 该边起点的期望经过次数 * 从该起点出发经过该路径的概率。

于是问题转成了求每个点的期望经过次数。很显然，每个点的期望经过次数 = \sum 入边的期望经过次数。

于是发现边与点的期望值是相辅相成的关系，由于是有向无环图，所以拓扑排序就可以了。

[bzoj3143][Hnoi2013]游走

一个无向连通图，顶点从1编号到 N ，边从1编号到 M 。

小Z在该图上进行随机游走，初始时小Z在1号顶点，每一步小Z以相等的概率随机选择当前顶点的某条边，沿着这条边走到下一个顶点，获得等于这条边的编号的分数。当小Z到达 N 号顶点时游走结束，总分为所有获得的分数之和。

现在，请你对这 M 条边进行编号，使得小Z获得的总分的期望值最小。

$2 \leq N \leq 500$ 且是一个无向简单连通图。

其实这道题与绿豆蛙有相似之处，但是图变为了无向连通图。

首先明确一个简单的贪心：我们希望期望经过次数最大的边编号最小。然后就是如何求期望经过次数了。原理与上一题一样，由点推边。

其实这道题与绿豆蛙有相似之处，但是图变为了无向连通图。

首先明确一个简单的贪心：我们希望期望经过次数最大的边编号最小。然后就是如何求期望经过次数了。原理与上一题一样，由点推边。

对于一号点， $f[1] = 1 + \sum f[j]/deg[j]$, j 和1有边。

对于其他点， $f[i] = \sum f[j]/deg[j]$, j 和 i 有边。

但由于是无向图，每个点的值会互相影响，所以不能递推。

高斯消元求解。

[POJ2096] Collecting Bugs

一个软件有 s 个子系统，会产生 n 种bug。某人每天发现一个bug，这个bug属于一个子系统，属于一个分类。每个bug属于某个子系统的概率是 $1/s$ ，属于某种分类的概率是 $1/n$ 。问发现 n 种bug，每个子系统都发现bug的天数的期望。

[POJ2096] Collecting Bugs

一个软件有 s 个子系统，会产生 n 种bug。某人每天发现一个bug，这个bug属于一个子系统，属于一个分类。每个bug属于某个子系统的概率是 $1/s$ ，属于某种分类的概率是 $1/n$ 。问发现 n 种bug，每个子系统都发现bug的天数的期望。

设 $dp[i,j]$:已经找到了 i 个系统， j 种病毒，还需要的期望天数

[POJ2096] Collecting Bugs

一个软件有 s 个子系统，会产生 n 种bug。某人每天发现一个bug，这个bug属于一个子系统，属于一个分类。每个bug属于某个子系统的概率是 $1/s$ ，属于某种分类的概率是 $1/n$ 。问发现 n 种bug，每个子系统都发现bug的天数的期望。

设 $dp[i, j]$:已经找到了 i 个系统， j 种病毒，还需要的期望天数

$$\begin{aligned} dp[i, j] = & dp[i + 1, j] * \frac{s - i}{s} * \frac{j}{n} \\ & + dp[i, j + 1] * \frac{n - j}{n} * \frac{i}{s} \\ & + dp[i + 1, j + 1] * \frac{s - i}{s} * \frac{n - j}{n} \\ & + dp[i, j] * \frac{i}{s} * \frac{j}{n} \end{aligned}$$

给定 n 种物品，每次购买会随机买到一种，询问买到 n 种物品的期望次数。

给定 n 种物品，每次购买会随机买到一种，询问买到 n 种物品的期望次数。

当我们已经买到 k 种物品了，再继续买多少次能得到第 $k + 1$ 种物品。

给定 n 种物品，每次购买会随机买到一种，询问买到 n 种物品的期望次数。

当我们已经买到 k 种物品了，再继续买多少次能得到第 $k + 1$ 种物品。

这个值是 $\frac{n}{n-k}$ 次(可以解方程或者求无穷等比数列得出)。

那么 $ans = n \sum_{i=1}^n \frac{1}{n}$ 。

初始有 k 只生物，这种生物只能活一天，死的时候有 P_i 的概率产生 i 只新生物(也只能活一天)，询问 m 天后所有生物都死的概率(包括 m 天前死亡的情况)

初始有 k 只生物，这种生物只能活一天，死的时候有 P_i 的概率产生 i 只新生物(也只能活一天)，询问 m 天后所有生物都死的概率(包括 m 天前死亡的情况)

首先每只生物间互不影响，所以我们只考虑初始有1只的情况

初始有 k 只生物，这种生物只能活一天，死的时候有 P_i 的概率产生 i 只新生物(也只能活一天)，询问 m 天后所有生物都死的概率(包括 m 天前死亡的情况)

首先每只生物间互不影响，所以我们只考虑初始有1只的情况
 $dp[i]$:前 i 天内所有生物都死的概率。

初始有 k 只生物，这种生物只能活一天，死的时候有 P_i 的概率产生 i 只新生物(也只能活一天)，询问 m 天后所有生物都死的概率(包括 m 天前死亡的情况)

首先每只生物间互不影响，所以我们只考虑初始有1只的情况

$dp[i]$:前 i 天内所有生物都死的概率。

$$dp[i] = \sum_{j=0}^{n-1} p_j * (dp[i-1])^j$$

$$ans = (dp[m])^k$$

有 n 张 $+1$ 和 m 张 -1 ，可以中途停止拿牌，询问按照最优拿牌，最后的得分期望。

有 n 张 $+1$ 和 m 张 -1 ，可以中途停止拿牌，询问按照最优拿牌，最后的得分期望。

由于有决策存在，所以要满足最优子结构，而期望正是能用来表示该状态的好坏的。继续翻牌的期望小于0，莫不如不取。

有 n 张 $+1$ 和 m 张 -1 ，可以中途停止拿牌，询问按照最优拿牌，最后的得分期望。

由于有决策存在，所以要满足最优子结构，而期望正是能用来表示该状态的好坏的。继续翻牌的期望小于0，莫不如不取。

设 $dp[i, j]$:有 i 张 $+1$ 和 j 张 -1 的期望得分。

[BZOJ1419]/[SRM420 Div1 500pt] Red is good

有 n 张 $+1$ 和 m 张 -1 ，可以中途停止拿牌，询问按照最优拿牌，最后的得分期望。

由于有决策存在，所以要满足最优子结构，而期望正是能用来表示该状态的好坏的。继续翻牌的期望小于0，莫不如不取。

设 $dp[i, j]$:有 i 张 $+1$ 和 j 张 -1 的期望得分。

$$dp[i, j] = \max(0, (dp[i-1, j] + 1) * \frac{i}{i+j} + (dp[i, j-1] - 1) * \frac{j}{i+j})$$

$$dp[0, j] = 0 \quad dp[i, 0] = i$$

[51NOD1670] 打怪兽

给定 n 个怪兽的能量值(均在 $[0, n - 1]$)，人的初始能量值为0，若人的能量 \geq 怪兽的能量就能打败，求死亡的期望能量值

[51NOD1670] 打怪兽

给定 n 个怪兽的能量值(均在 $[0, n-1]$)，人的初始能量值为0，若人的能量 \geq 怪兽的能量就能打败，求死亡的期望能量值

$dp[i]$:第 i 轮结束后人依然没死的概率 $sum[i]$:能量值 $\leq i$ 的怪兽数

$dp[i] = dp[i-1] * \frac{sum[i-1]-i+1}{n-i+1}$ $dp[i-1] - dp[i]$:死亡时能量值为 $i-1$ 的概率
 $ans = \sum (dp[i-1] - dp[i]) * (i-1) = \sum dp[i]$

[UVA10529] Dumb Bones

要求摆放 n 块骨牌，但摆放时会有 p_l 的概率放的这块的左边所有挨着的骨牌都倒下，同理右边的概率为 $p_r(1 - p_l - p_r > 0)$ ，询问摆完 n 块的期望次数。

[UVA10529] Dumb Bones

要求摆放 n 块骨牌，但摆放时会有 p_l 的概率放的这块的左边所有挨着的骨牌都倒下，同理右边的概率为 p_r ($1 - p_l - p_r > 0$)，询问摆完 n 块的期望次数。

我们考虑将两堆摆好的骨牌合并,即 $[1..i-1]$ 和 $[i+1..n]$ 已经被摆好,现在我们要把第 i 块放进去, $[1..i-1]$ 被摆好的期望为 E_1 , $[i+1..n]$ 为 E_2
分为往左倒的期望次数 $\times E_1$ + 往右倒的期望次数 $\times E_2$ + 不倒的期望次数 $\times 1$

[UVA10529] Dumb Bones

要求摆放 n 块骨牌，但摆放时会有 p_l 的概率放的这块的左边所有挨着的骨牌都倒下，同理右边的概率为 p_r ($1 - p_l - p_r > 0$)，询问摆完 n 块的期望次数。

我们考虑将两堆摆好的骨牌合并,即 $[1..i-1]$ 和 $[i+1..n]$ 已经被摆好,现在我们要把第 i 块放进去, $[1..i-1]$ 被摆好的期望为 E_1 , $[i+1..n]$ 为 E_2
分为往左倒的期望次数 $\times E_1$ + 往右倒的期望次数 $\times E_2$ + 不倒的期望次数 $\times 1$

不倒的期望次数很明显就是概率的倒数 $\frac{1}{1-p_l-p_r}$ ，不倒的期望代表了什么？代表了到这步之前都是倒的！

也就是说往左倒和往右倒的期望次数和为 $\frac{1}{1-p_l-p_r} - 1 = \frac{p_l+p_r}{1-p_l-p_r}$

那么往左倒的期望次数为 $\frac{p_l + p_r}{1 - p_l - p_r} * \frac{p_l}{p_l + p_r} = \frac{p_l}{1 - p_l - p_r}$, 同理往右倒的期望次数为 $\frac{p_r}{1 - p_l - p_r}$

那么往左倒的期望次数为 $\frac{p_l + p_r}{1 - p_l - p_r} * \frac{p_l}{p_l + p_r} = \frac{p_l}{1 - p_l - p_r}$, 同理往右倒的期望次数为 $\frac{p_r}{1 - p_l - p_r}$

$E = \frac{p_l}{1 - p_l - p_r} E_1 + \frac{p_r}{1 - p_l - p_r} E_2 + \frac{1}{1 - p_l - p_r}$ 注意还要加上合并两块之前, 两块各自摆好的期望次数 $E_1 + E_2$

那么往左倒的期望次数为 $\frac{p_l+p_r}{1-p_l-p_r} * \frac{p_l}{p_l+p_r} = \frac{p_l}{1-p_l-p_r}$, 同理往右倒的期望次数为 $\frac{p_r}{1-p_l-p_r}$

$E = \frac{p_l}{1-p_l-p_r} E_1 + \frac{p_r}{1-p_l-p_r} E_2 + \frac{1}{1-p_l-p_r}$ 注意还要加上合并两块之前, 两块各自摆好的期望次数 $E_1 + E_2$

$dp[i]$: 摆好连续 i 块骨牌的期望次数

$$dp[i] = \min(dp[i], \frac{1-p_r}{1-p_l-p_r} dp[j] + \frac{1-p_l}{1-p_l-p_r} dp[i-j-1] + \frac{1}{1-p_l-p_r})$$

那么往左倒的期望次数为 $\frac{p_l+p_r}{1-p_l-p_r} * \frac{p_l}{p_l+p_r} = \frac{p_l}{1-p_l-p_r}$, 同理往右倒的期望次数为 $\frac{p_r}{1-p_l-p_r}$

$E = \frac{p_l}{1-p_l-p_r} E_1 + \frac{p_r}{1-p_l-p_r} E_2 + \frac{1}{1-p_l-p_r}$ 注意还要加上合并两块之前, 两块各自摆好的期望次数 $E_1 + E_2$

$dp[i]$: 摆好连续 i 块骨牌的期望次数

$$dp[i] = \min(dp[i], \frac{1-p_r}{1-p_l-p_r} dp[j] + \frac{1-p_l}{1-p_l-p_r} dp[i-j-1] + \frac{1}{1-p_l-p_r})$$

这个复杂度可以利用dp值(期望)的单调性从 $O(N^2)$ 降到 $O(N)$, $O(N^2)$ 即可通过。

[Noi2005] 聪聪和可可

给定一张无向图,两个点A和B,每次A都向着离B最近的方向走,如果有两个点都可以,走编号小的那个,A如果走一次没遇到可以在这个时间内再走一步,B每次可以留在原来的点或者等概率走相邻的点,询问A遇到B的期望时间

[Noi2005] 聪聪和可可

给定一张无向图,两个点A和B,每次A都向着离B最近的方向走,如果有两个点都可以,走编号小的那个,A如果走一次没遇到可以在这个时间内再走一步,B每次可以留在原来的点或者等概率走相邻的点,询问A遇到B的期望时间

每次走最近相同走编号最小,我们先 $O(N^2)$ 处理出 $path[i, j]$:A在i,B在j,A下一步走到哪里

[Noi2005] 聪聪和可可

给定一张无向图,两个点A和B,每次A都向着离B最近的方向走,如果有两个点都可以,走编号小的那个,A如果走一次没遇到可以在这个时间内再走一步,B每次可以留在原来的点或者等概率走相邻的点,询问A遇到B的期望时间

每次走最近相同走编号最小,我们先 $O(N^2)$ 处理出 $path[i, j]$:A在i,B在j,A下一步走到哪里

$deg[i]$:点i的度数

$dp[i, j]$:A在i,B在j相遇的期望时间

[Noi2005] 聪聪和可可

给定一张无向图,两个点A和B,每次A都向着离B最近的方向走,如果有两个点都可以,走编号小的那个,A如果走一次没遇到可以在这个时间内再走一步,B每次可以留在原来的点或者等概率走相邻的点,询问A遇到B的期望时间

每次走最近相同走编号最小,我们先 $O(N^2)$ 处理出 $path[i, j]$:A在i,B在j,A下一步走到哪里

$deg[i]$:点i的度数

$dp[i, j]$:A在i,B在j相遇的期望时间

$$dp[i, j] = \frac{\sum_{k=\text{与}j\text{相邻的点以及}j\text{本身}} dp[path[path[i, j], j], k]}{deg[j]+1} + 1$$

$$dp[i, i] = 0$$

$$path[i, j] = j \text{ 或 } path[path[i, j], j] = j, \quad dp[i, j] = 1$$

记忆化搜索一下

[BZOJ2318] [SPOJ4060] game with probability Problem

有 n 枚石子,A先手拿,每次抛硬币,正面就取出一枚石子,否则不操作
A有 p 的概率抛出自己期望的那一面,B有 q 的概率抛出自己期望的那一面
询问A获胜的概率

[BZOJ2318] [SPOJ4060] game with probability Problem

有 n 枚石子,A先手拿,每次抛硬币,正面就取出一枚石子,否则不操作
A有 p 的概率抛出自己期望的那一面,B有 q 的概率抛出自己期望的那一面
询问A获胜的概率

f_i 表示剩 i 个石头、 A 先手的获胜概率。

g_i 表示剩 i 个石头、 A 后手的获胜概率。

如果想选, $f_i = p * g_{i-1} + (1 - p) * g_i$, $g_i = q * f_{i-1} + (1 - q) * f_i$

[BZOJ2318] [SPOJ4060] game with probability Problem

有 n 枚石子,A先手拿,每次抛硬币,正面就取出一枚石子,否则不操作
A有 p 的概率抛出自己期望的那一面,B有 q 的概率抛出自己期望的那一面
询问A获胜的概率

f_i 表示剩 i 个石头、 A 先手的获胜概率。

g_i 表示剩 i 个石头、 A 后手的获胜概率。

如果想选, $f_i = p * g_{i-1} + (1 - p) * g_i$, $g_i = q * f_{i-1} + (1 - q) * f_i$

然后对于不想选的情况, 那么 $p = 1 - p$, $q = 1 - q$ 就行了。

整理得:

$$f_i = \frac{p * g_{i-1} + (1-p) * q * f_{i-1}}{1 - (1-p) * (1-q)}$$

$$g_i = \frac{q * f_{i-1} + (1-q) * p * g_{i-1}}{1 - (1-p) * (1-q)}$$

然后剩 i 个石头时A想不想选与 f_{i-1} 、 g_{i-1} 的大小关系有关。

然后剩 i 个石头时A想不想选与 f_{i-1} 、 g_{i-1} 的大小关系有关。
 $f_{i-1} > g_{i-1}$ 都不想选。 $f_{i-1} < g_{i-1}$ 都想选。

然后剩 i 个石头时A想不想选与 f_{i-1} 、 g_{i-1} 的大小关系有关。

$f_{i-1} > g_{i-1}$ 都不想选。 $f_{i-1} < g_{i-1}$ 都想选。

然而这样就没法用矩阵乘法了。

当 n 很大时，其实概率已经基本不动了，让 $n = \min(n, 1000)$ 就好了。

NOI2012 迷失游乐园

N 个点的基环树（或树）。

问从每个点出发一直走下去，不能重复经过某个点，走过的路径长度的数学期望是多少？ $N \leq 100000$ 。

图中至多只有一个环，并且环长不超过20。

前50%的数据， $N-1$ 条边(树)。

后50%的数据， N 条边(基环树)。

有60%的数据， $N \leq 1000$ 。

先考虑简单情形树形结构，无根树转成有根树来处理。

先考虑简单情形树形结构，无根树转成有根树来处理。

$\text{son}[x]$ 为 x 的儿子数量。

$\text{down}[x]$ 表示从 x 这个点出发，向叶子们走的期望长度。

$\text{up}[x]$ 表示从 x 这个点出发，经过父亲，走到某个叶子的期望长度。

先考虑简单情形树形结构，无根树转成有根树来处理。

$\text{son}[x]$ 为 x 的儿子数量。

$\text{down}[x]$ 表示从 x 这个点出发，向叶子们走的期望长度。

$\text{up}[x]$ 表示从 x 这个点出发，经过父亲，走到某个叶子的期望长度。

先算出来每个结点的 $\text{down}[x]$

$$\text{down}[x] = \frac{\sum (\text{down}[y] + \text{len}(x \rightarrow y))}{\text{son}[x]}$$

先考虑简单情形树形结构，无根树转成有根树来处理。

$\text{son}[x]$ 为 x 的儿子数量。

$\text{down}[x]$ 表示从 x 这个点出发，向叶子们走的期望长度。

$\text{up}[x]$ 表示从 x 这个点出发，经过父亲，走到某个叶子的期望长度。

先算出来每个结点的 $\text{down}[x]$

$$\text{down}[x] = \frac{\sum(\text{down}[y] + \text{len}(x \rightarrow y))}{\text{son}[x]}$$

再dfs来求 $\text{up}[x]$

$$\text{up}[x] = \text{len}(x \rightarrow f) + \frac{\text{son}[f] * \text{down}[f] - \text{down}[x] - \text{len}(f \rightarrow x) + \text{up}[f]}{\text{son}[f] - 1 + 1}$$

先考虑简单情形树形结构，无根树转成有根树来处理。

$son[x]$ 为 x 的儿子的数量。

$down[x]$ 表示从 x 这个点出发，向叶子们走的期望长度。

$up[x]$ 表示从 x 这个点出发，经过父亲，走到某个叶子的期望长度。

先算出来每个结点的 $down[x]$

$$down[x] = \frac{\sum (down[y] + len(x \rightarrow y))}{son[x]}$$

再dfs来求 $up[x]$

$$up[x] = len(x \rightarrow f) + \frac{son[f] * down[f] - down[x] - len(f \rightarrow x) + up[f]}{son[f] - 1 + 1}$$

$up[x]$ 和 $down[x]$ 求出来后答案就很好求了

$$ans = \sum_{i=1}^n \frac{down[i] * son[i] + up[i]}{son[i] + 1}$$

环上的点的down值的求法和树一样。

环上的点的down值的求法和树一样。

环上的点的up值其实就是沿着环走到其他的任意一个环上的点（顺时针逆时针两种走法），然后再向下走的期望长度。 $O(n + k^2)$

环上的点的down值的求法和树一样。

环上的点的up值其实就是沿着环走到其他的任意一个环上的点（顺时针逆时针两种走法），然后再向下走的期望长度。 $O(n + k^2)$

实际上求up可以不用枚举走到哪个点，只要减掉转一圈回到自己的那部分就行了。 $O(n)$