装备强化小结

写在前面

笔者出道伊始，见识浅薄，着笔粗陋，或有错漏，忐企斧正。闲话不表，言归正传。

引子

但凡有装备（等级）系统的游戏，一般都会遇到装备强化（或称“升星”）的问题。对此的讨论历久不衰，加上自补的细枝末节，不外乎以下几个问题：

问题一、已知：各级强化后可能到达的状态及对应概率。计算：从某低级开始，在固定次数内，强化到某高级的概率。

问题二、已知：各级强化消耗、强化后可能到达的状态及对应概率。计算：从某低级到某高级所需进行的平均强化次数及消耗。

问题三、已知：各级强化消耗、强化后可能到达的状态及对应概率。计算：从某低级开始进行固定次数强化，所能到达的平均等级及平均消耗。

对此，同仁群策群力，公式结论、程序模拟……各种实现方法层出不穷。即便如此，仍有不少人理不清此些问题的内容和关系。特妄开此文，一则权作总结，二则以正视听。

理论准备：

1、一般随机过程

（1）、定义：

依赖于一个变动参数$\boldsymbol{t}$的一族随机变量$\{\boldsymbol{X(t)}, \boldsymbol{t} \in \boldsymbol{T}\}$。其中，变动参数$\boldsymbol{t}$的所有可取值的集合$\boldsymbol{T}$为参数空间。$\boldsymbol{X(t)}$的值所构成的集合$\boldsymbol{S}$成为随机过程的状态空间。例如，从时间$\boldsymbol{t=0}$开始记录某电话总机的呼叫次数，设$\boldsymbol{t=0}$时没有呼叫，至时刻$\boldsymbol{t}$的呼叫次数记作$\boldsymbol{N_t}$，则随机变量族$\{\boldsymbol{N_t}, \boldsymbol{t \geq 0}\}$是随机过程。

（2）、马氏过程：

如果已知在时间$\boldsymbol{t}$系统处于状态$\boldsymbol{X}$的条件下，在时刻$\boldsymbol{\tau(\tau > t)}$系统所处状态与时刻$\boldsymbol{t}$以前系统所处的状态无关，此过程便为马尔可夫过程（随机过程的一个子类）。例如，在布朗运动中，已知时刻$\boldsymbol{t}$下的运动状态条件下，微粒在$\boldsymbol{t}$后的运动情况和微粒在$\boldsymbol{t}$以前的情况无关。若$\boldsymbol{X(t)}$表示微粒在时刻$\boldsymbol{t}$的位置，则$\boldsymbol{X(t)}$是马尔可夫过程。

2、马尔科夫链

（1）、定义：

设$\{\mathbf{X}_{\boldsymbol{n}} = \boldsymbol{1,2,\ldots}\}$是一个随机变量序列，用“$\boldsymbol{X_n = i}$”表示时刻$\boldsymbol{n}$系统处于状态$\boldsymbol{i}$这一事件，称$\boldsymbol{p_{ij}(n) = p(\mathbf{X}_{n+1} = j \mid \mathbf{X}_n = i)}$为事件“$\boldsymbol{\mathbf{X}_n = i}$”出现的条件下，事件“$\boldsymbol{\mathbf{X}_{n+1} = j}$”出现的概率，又称它为系统的一步转移概率。若对任意的非负整数$\boldsymbol{i_1}$、$\boldsymbol{i_2}$、$\boldsymbol{i_3 \ldots i_{n-1}}$、$\boldsymbol{i}$、$\boldsymbol{j}$以及一切$\boldsymbol{n \geq 0}$，有$\boldsymbol{p(\mathbf{X}_{n+1} = j \mid \mathbf{X}_n = i, \mathbf{X}_k = i_k, k = 1,2,\ldots n-1) = p(\mathbf{X}_{n+1} = j \mid \mathbf{X}_n = i) = p_{ij}(n)}$，则称$\{\boldsymbol{\mathbf{X}_n}\}$是一个马尔可夫链。一步转移概率有以下的性质：

$$\boldsymbol{p_{ij} \geq 0, (i,j = 1,2,\ldots,n)}$$

$$\boldsymbol{\sum_{j=1}^{n} p_{ij} = 1\,, (i,j = 1,2,\ldots,n)}$$

把各个状态之间的一步转移概率排成矩阵，成为状态矩阵：

$$\boldsymbol{P = \begin{pmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{pmatrix}}$$

每个状态$\boldsymbol{i}$对应状态矩阵$\boldsymbol{P}$的第$\boldsymbol{i}$行。

3、$\boldsymbol{k}$步转移概率与$\boldsymbol{k}$步转移矩阵

（1）、$\boldsymbol{k}$步转移概率

系统从状态$\boldsymbol{i}$恰好经过$\boldsymbol{k}$步转移到状态$\boldsymbol{j}$的概率。记作$p_{ij}^{(k)} = p(X_{k+1} = j \mid X_1 = i)$。

（2）、$\boldsymbol{k}$步转移矩阵

$$P^{(k)} = \left(p_{ij}^{(k)}\right)_{n\times n}$$

显然，$P^{(k)}$为概率矩阵，即有：

$$p_{ij}^{(k)} \geq 0, (i, j = 1, 2, \ldots, n)$$

$$\sum_{j=1}^{n} p_{ij}^{(k)} = 1\,, (i, j = 1, 2, \ldots, n)$$

（3）、切普曼·科尔莫格洛夫方程

$$p_{ij}^{(n)} = \sum_{k} p_{ik}^{(m)} p_{kj}^{(n-m)}$$

$$P^{(n)} = P^{(m)} P^{(n-m)}$$

由以上方程可以推出：

$$P^{(n)} = P^{(n-1)} P = P^n$$

建模及符号规范：

装备基础等级为$\boldsymbol{1}$，最高等级为$\boldsymbol{N}$。在等级$\boldsymbol{x}$的时候进行一次强化操作，所能到达的等级$\boldsymbol{y}$对应的概率为p_{xy}，即一步转移概率矩阵为：

$$P = \begin{pmatrix} p_{11} & \cdots & p_{1N} \\ \vdots & \ddots & \vdots \\ p_{N1} & \cdots & p_{NN} \end{pmatrix}$$

在等级$\boldsymbol{i}$进行一次强化操作的成本为c_i，那么成本向量就是：

$$\vec{c} = (c_1 c_2 c_3 \ldots c_N)$$

另外，为表达方便，还需要以下几个工具矩阵：

$$I_x = (0 \ldots 1 \ldots 0)，1 \times N，第x个元素为1$$

$$I_y = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}，N \times 1，第y个元素为1$$

$$E_y = \begin{pmatrix} 1 & \cdots & 0 \\ \vdots & 0 & \vdots \\ 0 & \cdots & 1 \end{pmatrix}，N \times N，单位矩阵对角线上第y个元素为0$$

并且规定：文中讨论涉及到的原始概率矩阵$\boldsymbol{P}$的各个状态（也即装备强化过程中的各个等级）都是互通的（或者可遍历的，即从任何一个状态开始，经过有限步，可以到达任何一个其他状态）。因此概率矩阵$\boldsymbol{P}$的最后一行，也就是顶级状态对应的强化概率向量，不应该是：

$$(0 \ldots 0 \ldots 1)$$

而应该是：

$$(\mathbf{1} \dots \mathbf{0} \dots \mathbf{0})$$

即在顶级状态进行强化，将必然到达底级状态。

【问题一】、已知：一步转移概率矩阵P。计算：从x级开始，在n次内，强化到y级的概率。

一、分析

前面已经知道，从等级x开始进行n次强化操作，恰好到达的等级y的概率为$p_{xy}^{(n)}$，但是在这个问题中应该注意的是，不论强化次数是否到达n，只要到达y级，就停止强化。这看似使问题复杂化，实际上，只需将y级转化成吸收态，就可轻易解决。因为，只要到达吸收态，以后是否再强化都不会影响结果。具体做法：

（1）、将P的第y行换作I_y'；

（2）、求P^n，取结果矩阵的第x行第y列元素，即为所求。

二、实现及举例

举例：一把 1 级的屠龙刀，最高可以升到 9 级，每次强化成功率 30%，失败率 70%。失败会退一级，最差退到 1 级。那么在 1000 次内强化到 9 级的概率为多少？

根据上面的结论，编写$Matlab$代码如下（程序前半部分用的是模拟方法，用以和后面的理论方法进行比较，亦作验证用）：

```
clc;clearall;format long;
%一步转移概率矩阵:
P=[0.7  0.3  0.0  0.0  0.0  0.0  0.0  0.0  0.0
0.7  0.0  0.3  0.0  0.0  0.0  0.0  0.0  0.0
0.0  0.7  0.0  0.3  0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.7  0.0  0.3  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.7  0.0  0.3  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.7  0.0  0.3  0.0  0.0
0.0  0.0  0.0  0.0  0.0  0.7  0.0  0.3  0.0
0.0  0.0  0.0  0.0  0.0  0.0  0.7  0.0  0.3
0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.0];
n_bot=input('输入基础等级:');
n_top=input('输入目标等级:');
n_up=input('输入强化次数:');
%一、模拟方法:
%1、概率累计矩阵:
Q=zeros(9,10);Q(:,2)=P(:,1);
for i=2:9
Q(:,i+1)=sum(P(:,1:i)');
end
%2、模拟总人数及合格人数:
N=1000000;n=0;
%3、循环模拟体:
for j=1:N
up=0;
```

```
x=1;
while up<=n_up
        a=rand();
for i=1:9
if a>=Q(x,i) && a<=Q(x,i+1)
                x=i;
up=up+1;
break;
end
end
if x==n_top
            n=n+1;
break;
end
end
end
%4、结果输出：
fprintf('一、模拟方法：共有%g人进行强化操作，其中有%g人在%g次内强化到了目标等级，所占比例为：%g。\n',N,n,n_up,n/N);
%二、理论方法：
T=P;alpha=zeros(1,9);alpha(n_top)=1;T(n_top,:)=alpha;R=T^n_up;
fprintf('二、理论方法：计算吸收化的一步转移概率矩阵的%g次方，取结果矩阵的第%g行第%g列的元素，结果：%g。\n',n_up,n_bot,n_top,R(n_bot,n_top));
```

运行结果：

```
输入基础等级：1
输入目标等级：9
输入强化次数：1000
一、模拟方法：共有1e+006人进行强化操作，其中有229253人在1000次内强化到了目标等级，所占比例为：0.229253。
二、理论方法：计算吸收化的一步转移概率矩阵的1000次方，取结果矩阵的第1行第9列的元素，结果：0.228557。
```

作为补充，计算“从2级开始，在500次内强化到8级”的概率。运行结果为：

```
输入基础等级：2
输入目标等级：8
输入强化次数：500
一、模拟方法：共有1e+006人进行强化操作，其中有261879人在500次内强化到了目标等级，所占比例为：0.261879。
二、理论方法：计算吸收化的一步转移概率矩阵的500次方，取结果矩阵的第2行第8列的元素，结果：0.263293。
```

【问题二】、已知：一步转移概率矩阵$\boldsymbol{P}$、成本向量$\vec{c}$。计算：从$\boldsymbol{x}$级到$\boldsymbol{y}$级所需进行的平均强化次数$\boldsymbol{T_{x\to y}}$及消耗$\boldsymbol{C_{x\to y}}$。

一、分析

1、计算$\boldsymbol{T_{x\to y}}$

首先需要计算进行$\boldsymbol{k}$次强化到达等级$\boldsymbol{y}$的概率。注意这个概率并不是$\boldsymbol{k}$步转移矩阵中的那个$\boldsymbol{p_{xy}^{(k)}}$。理由是，我们要求的这个事件的停止条件是“只要升到等级$\boldsymbol{y}$就停止”；而$\boldsymbol{k}$步转移

中的停止条件是“只要强化次数到达$\boldsymbol{k}$就停止”，而不管$\boldsymbol{k}$步之前是否到达过等级$\boldsymbol{y}$。但是，这个概率却可以藉由$\boldsymbol{k}$步转移方法计算得出。

“只要升到等级$\boldsymbol{y}$就停止，共强化了$\boldsymbol{k}$次”的言外之意是“前$\boldsymbol{k-1}$次都没有到达过等级$\boldsymbol{y}$”，也就是说，前$\boldsymbol{k-1}$次的各个一步转移概率矩阵中，第$\boldsymbol{y}$列元素必须都等于$\boldsymbol{0}$。

那么，事件“从等级$\boldsymbol{x}$强化到等级$\boldsymbol{y}$，只要升到等级$\boldsymbol{y}$就停止，共强化了$\boldsymbol{k}$次”的概率记作$\boldsymbol{p_{x\to y}^{(k)}}$，就有：

$$\boldsymbol{p_{x\to y}^{(k)} = I_x\left(PE_y\right)^{k-1}PI_y}$$

于是，强化次数$\boldsymbol{t_{x\to y}}$的数学期望$\boldsymbol{T_{x\to y}}$（即平均次数）为：

$$\boldsymbol{T_{x\to y} = \sum_{k=1}^{\infty} k\cdot p_{x\to y}^{(k)} = I_x\left[\sum_{k=1}^{\infty} k\cdot\left(PE_y\right)^{k-1}\right]PI_y}$$

可以证明（略）：$\boldsymbol{PE_y}$是收敛矩阵，即：

$$\boldsymbol{\lim_{n\to+\infty}\left(PE_y\right)^n = 0}$$

根据矩阵论的相关结论，推出$\boldsymbol{I-PE_y}$可逆。记：

$$\boldsymbol{S(n) = \left[\sum_{k=1}^{n} k\cdot\left(PE_y\right)^{k-1}\right]}$$

两边右乘$\boldsymbol{PE_y}$，有：

$$\boldsymbol{S(n)PE_y = \left[\sum_{k=1}^{n} k\cdot\left(PE_y\right)^{k}\right]}$$

上面两个式子相减，得：

$$\boldsymbol{S(n)\left(I-PE_y\right) = \left[\sum_{k=1}^{n}\left(PE_y\right)^{k-1}\right] - n\left(PE_y\right)^n}$$

两边右乘$\boldsymbol{I-PE_y}$，得：

$$\boldsymbol{S(n)\left(I-PE_y\right)^2 = \left[\sum_{k=1}^{n}\left(PE_y\right)^{k-1}\right]\left(I-PE_y\right) - n\left(PE_y\right)^n\left(I-PE_y\right)}$$

即：

$$\boldsymbol{S(n)\left(I-PE_y\right)^2 = I + n\left(PE_y\right)^{n+1} - (n+1)\left(PE_y\right)^n}$$

所以：

$$\boldsymbol{S(n) = \left[I + n\left(PE_y\right)^{n+1} - (n+1)\left(PE_y\right)^n\right]\left(I-PE_y\right)^{-2}}$$

并且由：

$$\boldsymbol{n\left(PE_y\right)^{n+1} - (n+1)\left(PE_y\right)^n \to 0}$$

得到：

$$\boldsymbol{S(\infty) = \left(I-PE_y\right)^{-2}}$$

因此，最终结果是：

$$T_{x\to y}=I_x\left(I-PE_y\right)^{-2}PI_y$$

2、计算$C_{x\to y}$

下面计算强化总消耗。

在从等级x到等级y的强化过程中，如果计算出在各个等级上进行强化次数的数学期望：

$$\vec{r}=(r_1 r_2 r_3 \dots r_N)$$

那么，从等级x强化到等级y的总成本期望就是：

$$C_{x\to y}=\sum_{i=1}^{N}c_i\cdot r_i=\vec{c}\cdot\vec{r}$$

很显然：

$$T_{x\to y}=\sum_{i=1}^{N}r_i$$

剩下的问题就是计算$\vec{r}=(r_1 r_2 r_3 \dots r_N)$。

由归一化特点，必然有：

$$\sum_{k=1}^{\infty}p_{x\to y}^{(k)}=I_x\left[\sum_{k=1}^{\infty}\left(PE_y\right)^{k-1}P\right]I_y=1$$

即在事件序列“从等级x强化到等级y，只要升到等级y就停止，共强化了k次”中，装备处于等级y的积累概率为1，考虑到概率的意义，也即此过程中到达等级y的次数期望。那么对于

$$\sum_{k=1}^{\infty}\left(PE_y\right)^{k-1}P$$

其第y列自然全是1。第x行中第y个元素之外的各个数值就是装备从等级x强化到等级y过程中对应各个等级出现的次数期望。

另外，需要注意的是，由于计数发生在强化操作之后，也即对强化结果状态计数，为了得到强化前状态计数，只要将初始状态计数结果加1，结束状态计数结果减1即可。

所以：

$$\vec{r}=I_x\left\{\left[\sum_{k=1}^{\infty}\left(PE_y\right)^{k-1}\right]P+I\right\}E_y$$

同样，因为$I-PE_y$可逆，也可以直接计算出级数

$$\sum_{k=1}^{\infty}\left(PE_y\right)^{k-1}$$

的结果，此处不予赘述，而直接给出结果：

$$\vec{r}=I_x\left[\left(I-PE_y\right)^{-1}P+I\right]E_y$$

得到$\vec{r}$之后，就可以计算消耗总期望：

$$C_{x\to y}=\vec{c}\cdot\vec{r}$$

二、实现及举例

1、矩阵方法

经过前面的讨论，得知，比较可行的方法有两个：矩阵级数方法和逆矩阵方法。已知量有五个：

（1）、一步转移概率矩阵$\boldsymbol{P}$；

（2）、初始等级$\boldsymbol{x}$；

（3）、目标等级$\boldsymbol{y}$；

（4）、各等级强化消耗成本$\vec{\boldsymbol{c}}$；

（5）、矩阵级数解法中用以调整结果精度的项数$\boldsymbol{n}$。

只要确定以上参数，就可以得到结果。为此可以编写函数来实现此功能。***Matlab***代码如下：

```
function Result=Markov(P,x,y,c)
%P为一步转移概率矩阵，x为初始等级，y为目标等级，c为各个等级上进行一次升级操作的投入。
%返回结果Result=[upgrade_time,total_cost]:
% 1、upgrade_time为从x升级到y的升级次数期望；
% 2、total_cost为此过程中总消耗的期望。


%一、求P矩阵的阶数：
[P_size,~]=size(P);
%二、工具矩阵：
I=eye(P_size);
Ix=zeros(1,P_size);Ix(x)=1;
Iy=zeros(P_size,1);Iy(y)=1;
Ey=I;Ey(y,y)=0;
%三、计算过程：
if x==y
upgrade_time=0;
total_cost=0;
else
upgrade_time=Ix*(I-P*Ey)^-2*P*Iy;
total_cost=Ix*((I-P*Ey)^-1*P+I)*Ey*c';
end
Result=[upgrade_time,total_cost];
```

某网游装备强化规则如下：

1→2：强化成功率为 75%，强化失败强化仍为 1。

2→3：强化成功率为 60%，强化失败强化仍为 2。

3→4：强化成功率为 50%，强化失败掉回 2。

4→5：强化成功率为 45%，强化失败掉回 2。

5→6：强化成功率为 15%，强化失败掉回 2。

6→7：强化成功率为 20%，强化失败为 6。

7→8：强化成功率为 10%，强化失败掉回 6。

8→9：强化成功率为 1%，强化失败为 8。

9→10：强化成功率为 2%，强化失败为 8。

问，从 1 强化到 10：

1、平均需要强化多少次？

2、若在 1 到 9 级上，每次强化消耗的宝石数分别为 1，1，2，2，2，3，3，4，4。总共需要消耗多少个宝石？

编写程序如下：

```
clc;clearall;format short;
P=[0.25  0.75  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
0.00  0.40  0.60  0.00  0.00  0.00  0.00  0.00  0.00  0.00
0.00  0.50  0.00  0.50  0.00  0.00  0.00  0.00  0.00  0.00
0.00  0.55  0.00  0.00  0.45  0.00  0.00  0.00  0.00  0.00
0.00  0.85  0.00  0.00  0.00  0.15  0.00  0.00  0.00  0.00
0.00  0.00  0.00  0.00  0.00  0.80  0.20  0.00  0.00  0.00
0.00  0.00  0.00  0.00  0.00  0.90  0.00  0.10  0.00  0.00
0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.99  0.01  0.00
0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.98  0.00  0.02
1.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00];
x=input('输入基础等级： ');
y=input('输入目标等级： ');
c=[1 1 2 2 2 3 3 4 4 0];
[upgrade_time,total_cost]=Markov(P,x,y,c);
fprintf('从等级%g强化到等级%g，平均需要进行%g次强化。\n',x,y,upgrade_time);
fprintf('从等级%g强化到等级%g，一共需要消耗%g个宝石。\n',x,y,total_cost);
运行结果：
输入基础等级： 1
输入目标等级： 10
从等级1强化到等级10，平均需要进行5211.83次强化。
从等级1强化到等级10，一共需要消耗20532.9个宝石。
```

2、特殊方法

以上所有的推导计算过程都使用了矩阵这一数学工具，以实现“一般性”和“通用性”。当然，实际中遇到的概率矩阵$\boldsymbol{P}$往往比较简单，这时候就可以用比较初等的方法来求强化次数期望。

（1）、一般来说，凡强化成功，则等级$+1$；如果失败则有“等级不变、等级-1、等级降到底”这几种情况。

设从n级强化到$n+1$级，平均次数为x_n。综合起来：

当$n=1$时，$x_1=\frac{1}{a_1}$；

当$n>1$时，有：

a、等级$+1$，概率为a_n；

b、等级不变，概率为b_n，共经过x_n+1次强化到$n+1$级；

c、等级-1，概率为c_n，共经过$x_n+x_{n-1}+1$次强化到$n+1$级；

d、等级降到底1级，概率为d_n，共经过$1+\sum_{i=1}^{n}x_i$次强化到$n+1$级。

根据数学期望的线性性质，得：

$$x_n=a_n\cdot 1+b_n\cdot(x_n+1)+c_n\cdot(x_n+x_{n-1}+1)+d_n\cdot\left(1+\sum_{i=1}^{n}x_i\right)$$

记：

$$X_i=\sum_{i=1}^{n}x_i$$

就是从1级强化到n级的次数期望。

整理得：

$$x_n=1+\frac{b_n}{a_n}+\frac{c_n}{a_n}\cdot(x_{n-1}+1)+\frac{d_n}{a_n}\cdot(1+X_{n-1}),\ n>1$$

并且注意到$a_n+b_n+c_n+d_n=1$，进一步化简得：

$$x_n=\frac{1}{a_n}\cdot(1+c_n\cdot x_{n-1}+d_n\cdot X_{n-1}),\ n>1$$

由此，可以徒手迭代计算或者$EXCEL$拉表得到结果。

例如，某强化过程如下：

事件	1→2	2→3	3→4	4→5	5→6	6→7	7→8	8→9	9→10
a_n成功+1	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
b_n失败不变	0.6	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
c_n失败-1	0.0	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
d_n失败到 1	0.0	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1

制表结果（结果保留两位小数）：

事件	1->2	2->3	3->4	4->5	5->6	6->7	7->8	8->9	9->10
a_n成功+1	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
b_n失败不变	0.6	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
c_n失败-1	0	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
d_n失败到 1	0	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
左升右次数	2.50	4.38	6.41	9.02	12.59	17.52	24.36	33.87	47.10
1 升右次数	2.50	6.88	13.28	22.30	34.89	52.41	76.77	110.64	157.74

经检验和矩阵方法一致。

进一步，当$n>2$时，可以得到$\{X_n\}$的递推公式：

$$a_nX_n-(1-b_n)X_{n-1}+c_nX_{n-2}=1$$

在上面这个例子里，$a_n=a=0.4$，$b_n=b=0.3$，$c_n=c=0.2$。

那么递推公式就是：

$$X_n-1.75X_{n-1}+0.5X_{n-2}=2.5$$

容易计算出初始的两个值：

$$X_1=2.5,\ X_2=6.875$$

递归操作可以手动进行，可以$EXCEL$拉表，也可以借助软件（笔者用$Mathematica$，使用的函数为$RecurrenceTable$）。

输入：

$$RecurrenceTable[\{X[n]-1.75X[n-1]+0.5X[n-2]==2.5,X[1]==2.5,X[2]==6.875\},X,\{n,1,9\}]$$

结果：

{2.5,6.875,13.2813,22.3047,34.8926,52.4097,76.7706,110.644,157.741}

实际上，因为这是一个线性非齐次递推关系，完全可以用生成函数的方法给出解析解。下面仅给出部分提示：

$$(X_n+10)-1.75(X_{n-1}+10)+0.5(X_{n-2}+10)=0$$

设$X_n+10=T_n$，那么：

$$T_n-1.75T_{n-1}+0.5T_{n-2}=0，n>2$$

特征方程：

$$r^2-1.75r+0.5=0$$

剩下的部分就是纯计算，此处不赘述。

（2）、以上的情况比较简单，稍复杂一点的情况是这样的，在等级n进行一次强化操作，可能出现的结果有：1级、2级、…$n-1$级、n级和$n+1$级。从而：

设从n级强化到$n+1$级，平均次数为x_n，且：

$$X_i=\sum_{i=1}^{n}x_i$$

当$n=1$时，

$$X_1=x_1=\frac{1}{a_1}$$

当$n>1$时，有：

等级$+1$，到$n+1$级，概率为$p_{n,n+1}$；

等级$+0$，到$n+0$级，概率为$p_{n,n}$，共经过$X_n-X_{n-1}+1$次强化到$n+1$级；

等级-1，到$n-1$级，概率为$p_{n,n-1}$，共经过$X_n-X_{n-2}+1$次强化到$n+1$级；

……

等级$-k$，到$n-k$级，概率为$p_{n,n-k}$，共经过$X_n-X_{n-(k+1)}+1$次强化到$n+1$级；

……

等级$-(n-3)$，到3级，概率为$p_{n,3}$，共经过X_n-X_2+1次强化到$n+1$级；

等级$-(n-2)$，到2级，概率为$p_{n,2}$，共经过X_n-X_1+1次强化到$n+1$级；

等级$-(n-1)$，到1级，概率为$p_{n,1}$，共经过X_n+1次强化到$n+1$级。

然后根据数学期望的线性性质，得：

$x_n=X_n-X_{n-1}=$

$p_{n,n+1}+p_{n,n}\cdot(X_n-X_{n-1}+1)+p_{n,n-1}\cdot(X_n-X_{n-2}+1)+\cdots+$

$p_{n,n-k}\cdot\left(X_n-X_{n-(k+1)}+1\right)+\cdots+$

$p_{n,3}\cdot(X_n-X_2+1)+p_{n,2}\cdot(X_n-X_1+1)+p_{n,1}\cdot(X_n+1)$

化简得：

$$X_n=\frac{1}{p_{n,n+1}}\cdot\left[1+X_{n-1}-\sum_{\mathrm{i=2}}^{\mathrm{n}}\left(p_{n,i}\cdot X_{i-1}\right)\right]，n>1$$

这也是一个线性递推，当然也可以$EXCEL$写个宏来实现。为了验证，给出$Matlab$模拟代码和结果：

```
clc;clear all;
P=[0.1  0.9  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0
0.1  0.1  0.8  0.0  0.0  0.0  0.0  0.0  0.0  0.0
0.1  0.1  0.1  0.7  0.0  0.0  0.0  0.0  0.0  0.0
0.1  0.1  0.1  0.1  0.6  0.0  0.0  0.0  0.0  0.0
0.1  0.1  0.1  0.1  0.1  0.5  0.0  0.0  0.0  0.0
0.1  0.1  0.1  0.1  0.1  0.1  0.4  0.0  0.0  0.0
0.1  0.1  0.1  0.1  0.1  0.1  0.1  0.3  0.0  0.0
0.1  0.1  0.1  0.1  0.1  0.1  0.1  0.1  0.2  0.0
```

```
0.1  0.1  0.1  0.1  0.1  0.1  0.1  0.1  0.1  0.1
0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.0];
x=input('输入基础等级: ');
y=input('输入目标等级: ');
c=ones(1,10);
[upgrade_time,total_cost]=Markov(P,x,y,c);
fprintf('一、矩阵方法: 从等级%g强化到等级%g, 平均需要进行%g次强化。\n',x,y,upgrade_time);
X(1)=0;X(2)=1/P(1,2);
for n=2:9
    S=0;
for i=2:n
        S=S+P(n,i)*X(i);
end
X(n+1)=1/P(n,n+1)*(1+X(n)-S);
end
fprintf('二、递推方法: 从等级%g强化到等级%g, 平均需要进行%g次强化。\n',x,y,X(y)-X(x));
```

运行结果:

```
输入基础等级: 1
输入目标等级: 10
一、矩阵方法: 从等级1强化到等级10, 平均需要进行3117.36次强化。
二、递推方法: 从等级1强化到等级10, 平均需要进行3117.36次强化。
```

【问题三】、已知: 一步转移概率矩阵P、成本向量c。计算: 从x级开始进行n次强化, 所能到达的平均等级Y及平均消耗C_x^n。

一、分析

1、根据一开始的知识预备中"k步转移概率与k步转移矩阵"部分, $p_{xy}^{(n)}$表示从等级x开始, 进行n次强化, 恰好到达等级y的概率。那么:

$$Y=\sum_{i=1}^{N} i\cdot p_{xi}^{(n)}$$

因为P^n也是概率矩阵, 所以P^n的各行和都是1, 从而, 从小于顶级的任何一级开始, 不论经过任何次数强化, 所能到达的平均等级一定小于顶级。这是因为:

如果:

$$x_i<1, \text{且} \sum_{i}^{n} x_i = 1$$

那么:

$$\sum_{i}^{n} a_i\cdot x_i < max\{a_i\}$$

实际上:

$$\sum_{i}^{n} a_i \cdot x_i < \sum_{i}^{n} max\{a_i\} \cdot x_i < max\{a_i\} \cdot \sum_{i}^{n} x_i = max\{a_i\}$$

2、下面计算平均消耗C_x^n。从x级开始，

进行 1 次强化，消耗：

$$C_x^1 = c(x) = I_x \cdot \vec{c}$$

进行 2 次强化，消耗：

$$C_x^2 = I_x \cdot \vec{c} + I_x P \cdot \vec{c}$$

进行 3 次强化，消耗：

$$C_x^3 = I_x \cdot \vec{c} + I_x P \cdot \vec{c} + \sum_{i=1}^{N} P(x,i) \sum_{j=1}^{N} P(i,j) c(j) = I_x \cdot \vec{c} + I_x P \cdot \vec{c} + I_x P^2 \cdot \vec{c}$$

进行4次强化，消耗：

$$C_x^4 = I_x \cdot \vec{c} + I_x P \cdot \vec{c} + I_x P^2 \cdot \vec{c} + \sum_{i=1}^{N} P(x,i) \sum_{j=1}^{N} P(i,j) \sum_{k=1}^{N} P(j,k)\, c(k)$$

$$= I_x \cdot \vec{c} + I_x P \cdot \vec{c} + I_x P^2 \cdot \vec{c} + I_x P^3 \cdot \vec{c}$$

……

由归纳法不难得到，进行n次强化，消耗：

$$C_x^n = \left(I_x \sum_{i=1}^{n} P^{i-1} \right) \cdot \vec{c}$$

当然，如果$I - P$可逆，则有：

$$C_x^n = I_x (I - P^n)(I - P)^{-1} \cdot \vec{c}$$

二、实现及举例

根据以上结论，编写函数来计算Y和C_x^n，如下：

```
function [Y,Cxn]=Enhance(P,x,n,c)
%已知：一步转移概率矩阵P，起始等级x，强化次数n和成本向量
%返回：n次强化总消耗的期望Y,所能到达的平均等级Y
[P_size,~]=size(P);Q=zeros(P_size);
I=eye(P_size);Ix=zeros(1,P_size);Ix(x)=1;
%计算Y：
if x==P_size
    Y=P_size;
else
    R=P^n;
    Y=[1:1:P_size]*R(x,:)';
end
%计算Cxn：
for i=1:n
Q=Q+P^(i-1);
end
Cxn=Ix*Q*c';
```

以本文问题二中第一个强化规律为例，给出几个结果：

1、从各个等级开始进行 100 次强化操作：

从1级开始，进行100次强化，平均可到5.54652级，平均消耗215.295个宝石。
从2级开始，进行100次强化，平均可到5.57453级，平均消耗217.683个宝石。
从3级开始，进行100次强化，平均可到5.60919级，平均消耗220.69个宝石。
从4级开始，进行100次强化，平均可到5.68491级，平均消耗225.34个宝石。
从5级开始，进行100次强化，平均可到5.86418级，平均消耗236.603个宝石。
从6级开始，进行100次强化，平均可到7.62958级，平均消耗350.111个宝石。
从7级开始，进行100次强化，平均可到7.65554级，平均消耗354.087个宝石。
从8级开始，进行100次强化，平均可到7.93628级，平均消耗397.966个宝石。
从9级开始，进行100次强化，平均可到7.88812级，平均消耗394.317个宝石。
从10级开始，进行100次强化，平均可到10级，平均消耗212.515个宝石。

2、从 1 级开始进行不同次数的强化操作：

从1级开始，进行10次强化，平均可到3.01434级，平均消耗14.3024个宝石。
从1级开始，进行20次强化，平均可到3.34948级，平均消耗31.0738个宝石。
从1级开始，进行30次强化，平均可到3.67534级，平均消耗49.3438个宝石。
从1级开始，进行40次强化，平均可到3.98912级，平均消耗69.0848个宝石。
从1级开始，进行50次强化，平均可到4.2888级，平均消耗90.2522个宝石。
从1级开始，进行60次强化，平均可到4.57306级，平均消耗112.79个宝石。
从1级开始，进行70次强化，平均可到4.84112级，平均消耗136.634个宝石。
从1级开始，进行80次强化，平均可到5.09267级，平均消耗161.714个宝石。
从1级开始，进行90次强化，平均可到5.32771级，平均消耗187.959个宝石。
从1级开始，进行100次强化，平均可到5.54652级，平均消耗215.295个宝石。

补充：

在文档设计伊始，曾有第四个问题：

已知：各级强化消耗、强化后可能到达的状态及对应概率、总强化投入。计算：从某低级开始，可强化到的平均等级。即：

已知：一步转移概率矩阵$\boldsymbol{P}$、成本向量$\vec{c}$、最大强化投入$\boldsymbol{C}$。计算：从$\boldsymbol{x}$级开始，可强化到的平均等级$\boldsymbol{Y}$。

当然，使用模拟方法可以很容易得出结果。不过，由于学识所限，在寻找这个问题的理论解法的时候遇到了一点困难，一时卡住没有思路。继而转念考虑这个问题是否有讨论的必要。在向业内资深人士询问之后，发现，实际上，利用第二个问题的结论就可以实现这第四个问题所要达到的目的。

实际上，我们只要计算出从$\boldsymbol{x}$级开始到其他各级所需要的平均消耗$\overline{\boldsymbol{C}}$，然后比较$\boldsymbol{C}$和$\overline{\boldsymbol{C}}$，即可得到所能强化到的平均整数等级$\overline{\boldsymbol{Y}}$。

另外，第四个问题的结果$\boldsymbol{Y}$是整数的概率很小，而玩家强化的预期不可能是一个非整数的等级，所以说作为整数的$\overline{\boldsymbol{Y}}$更有实际意义和参考价值。

至此，略去总结，全文完。

写在后面

讫文赘言，笔者出道伊始，见识浅薄，着笔粗陋，或有错漏，忐企斧正。

QQ：707509279