pvp ttk

模型设置

定义 X_n 为在时刻 n 时当前人物的面板状态
定义 Y_n 为在时刻 n 时目标人物的面板状态（如防御属性等）
定义 $D_n(X_n, Y_n)$ 为根据各自面板和技能系数造成的实际伤害，是一个随机变量。
这是因为会存在会心与不会心的情况。

考虑更为简单的情况。

我们假设实战中双方各自的面板 X_n，Y_n 不随时间发生改变。
那么 D_n 就是一个只跟会心有关的随机变量，服从二元离散分布。

那么 $D_n \in \{d_n, d_n^*\}$，其中 d_n^* 为实际会心造成的伤害。

$$P\left(D_n=d_n\right)=1-p, \quad P\left(D_n=d_n^*\right)=p$$

且 $\{D_n\}_{n \geq 0}$ 相互独立

记 $S_n=\sum_{i=1}^n D_i$ 为第 n 步时造成的累计总伤害。
其中每一步的伤害增量 D_n 只依赖于当前的状态（技能，会心率）。

注意到 $S_n=S_{n-1}+D_n$，且 D_n 相互独立，显然满足马尔科夫链的性质：

$$P\left(S_n=s \mid S_{n-1}, S_{n-2}, \ldots, S_0\right)=P\left(S_n=s \mid S_{n-1}\right)$$

由于每步增量 D_n 相互独立，那么我们有以下：

$$P\left(S_n=s \mid S_{n-1}=s^{\prime}\right)=P\left(D_n=s-s^{\prime}\right)$$

在开始本篇内容之前，先计算几个概率。

1.累计伤害在第 n 次出伤时的概率分布

定义 $f_n(s)=P(S_n=s)$，其含义为在第 n 次出伤时，累计伤害刚好为 s 的概率。
定义 $g_n(d)=P(D_n=d)$，其含义为打出第 n 个技能时，该伤害恰好为 d 的概率。

那么由全概率公式，我们可以得到：

$$
\begin{aligned}
f_n(s) &= P(S_n = s) \\
&= \sum_{s'} P(S_n = s | S_{n-1} = s') P(S_{n-1} = s') \\
&= \sum_{s'} P(D_n = s - s') P(S_{n-1} = s') \\
&= \sum_{s'} g_n(s - s') f_{n-1}(s')
\end{aligned}
$$

事实上，这就是一个卷积公式，可以用卷积的方式写出递归公式。

$$f_n(s) = (f_{n-1} * g_n)(s)$$

那么回到标题上，累计伤害在第 n 次出伤时的概率分布 f_n 就可以直接计算了。

做点变量替换以便后续的应用：

$$f_n(s) = \sum_d f_{n-1}(s - d) g_n(d)$$

即

$$f_n(s' + d) += f_{n-1}(s') g_n(d)$$

```
def pmf_convolve(f, d0, d1, p, f_next):
    for s, f in f.items():
        f_next[s + d0] += f * (1 - p)
        f_next[s + d1] += f * p
    return f_next
```

2.累计伤害在第 n 次出伤时目标已被击杀的概率分布

有了 n 次出伤时的概率分布，就可以计算出在第 n 次出伤时能否击杀目标的概率。

记 HP 为目标的生命值。

那么第 n 步累计伤害击杀概率则为：

$$P(S_n \geq HP) = \sum_{s \geq HP} f_n(s)$$

3.第 n 步的首次击杀概率

仅仅用累计伤害分布 $f_n(s)$ 得到的是 **到第 n 步累计击杀概率**，但是在伤害过剩的情况下，就有可能早于 n 步就有击杀的概率。

我们考虑这样一个概率，前面打了 n-1 个伤害，目标没死，在打第 n 个技能的时候，目标生命值耗尽的概率。

还是定义停时：

$$\tau = \inf\{n : S_n \geq HP\}$$

首次击杀的概率分布

$$P(\tau = n) = P(S_n \geq HP, S_{n-1} < HP) = \sum_{s<HP} P(S_n \geq HP, S_{n-1} = s)$$

用条件概率公式展开

$$\begin{aligned} P(S_n \geq HP, S_{n-1} = s) &= P(S_n \geq HP | S_{n-1} = s) P(S_{n-1} = s) \\ &= P(D_n \geq HP - s) f_{n-1}(s) \\ &= \sum_{d \geq HP-s} f_{n-1}(s) g_n(d) \end{aligned}$$

代入公式，得到

$$P(\tau = n) = \sum_{s<HP} \sum_{d \geq HP-s} f_{n-1}(s) g_n(d)$$

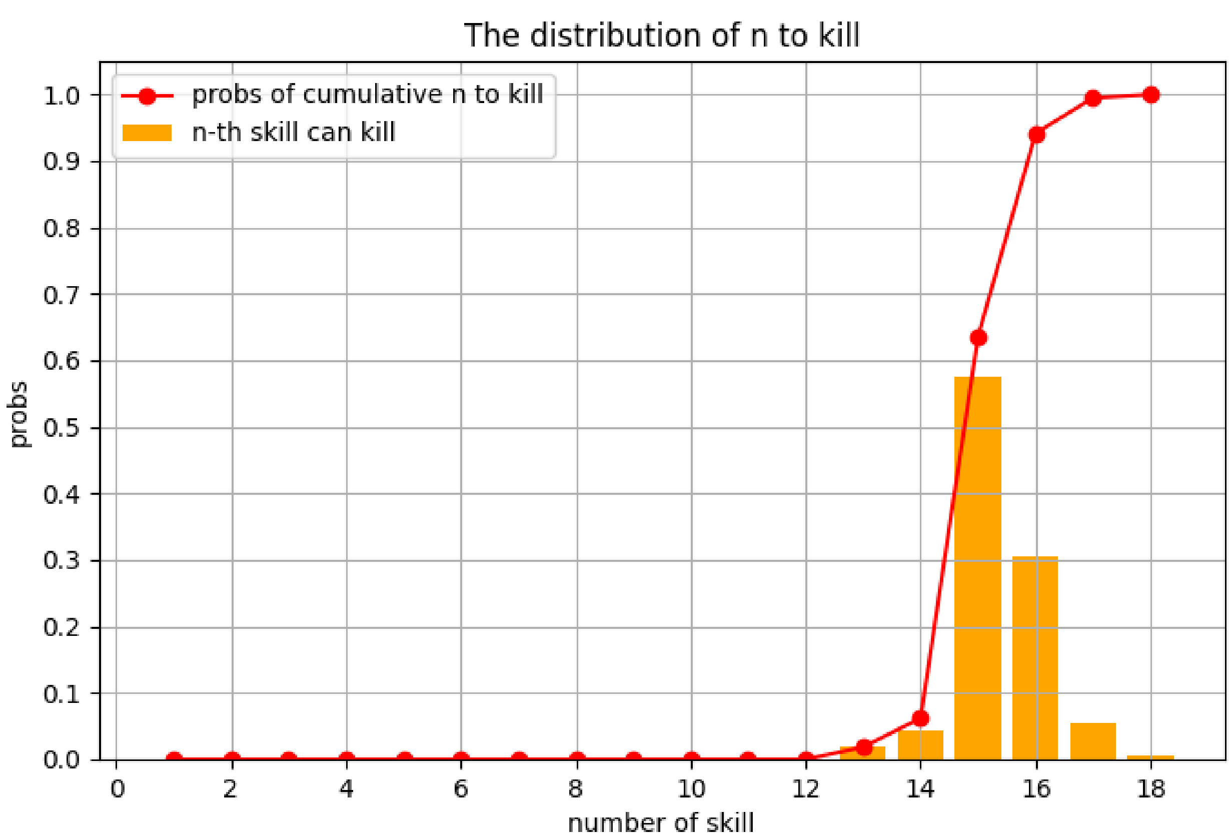
Example

做一个例子看看效果。

计算数据源于 **闪辰** 提供的段式计算器，借用其数据做个简单的分析。

技能名称	不会心	会心
劈风	163558	278080
劈风骤	59723	101540
骤风一次	42957	73035
抟风（奇穴100%）	170292	289529
横驱（1次）	73207	124466
横驱（1次）	73207	124466
飞鸿（共四次）	67824	115314
抟风爆（奇穴100%）	182707	310637
飞鸿（共四次）	67824	115314
横驱（1次）	73207	124466
飞鸿（共四次）	67824	115314
飞鸿（共四次）	67824	115314
单次破野	417604	710008
横驱（1次）	73207	124466
引窍（奇穴额外+200%增幅后）	317934	540549
（12层截阳绝脉）（奇穴20%增伤）	192014	326461
灵台	122426	208148
横驱（1次）	73207	124466

默认无治疗的情况给定一个连招循环，他的击杀概率随时间的变化



一个连招循环，他的出伤总量随时间变化的分布情况。
比如在连招的最末尾，他的出伤总量大致分布在280w左右，最低伤害也大于230w，给定的红线。

在n = 13的时候，接近连招的中段，累计出伤大致在180w左右，有小部分概率赌脸的情况能够越过红线。
在n = 16的时候，接近连招的末尾，累计出伤大致在260w左右，仅仅只有小部分概率，特别脸黑无法击杀。
在n = 18的时候，连招的末尾，累计出伤大致在280w左右，极其脸红的情况下能打到330w，不存在无法击杀的概率。