K臂老虎机问题

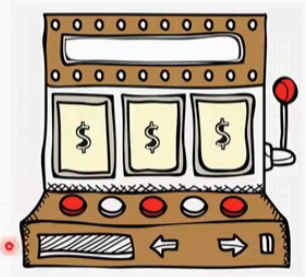
在多臂老虎机（multi-armed bandit，MAB）问题（见图 2-1）中，有一个拥有 K 根拉杆的老虎机，拉动每一根拉杆都对应一个关于奖励的概率分布 $\mathcal{R}$。我们每次拉动其中一根拉杆，就可以从该拉杆对应的奖励概率分布中获得一个奖励 r。我们在各根拉杆的奖励概率分布未知的情况下，从头开始尝试，目标是在操作 T 次拉杆后获得尽可能高的累积奖励。由于奖励的概率分布是未知的，因此我们需要在“探索拉杆的获奖概率”和“根据经验选择获奖最多的拉杆”中进行权衡。“采用怎样的操作策略才能使获得的累积奖励最高”便是多臂老虎机问题。如果是你，会怎么做呢？

一个**行动**真正的**价值**是选择该行动时的**平均回报**

$q = E\ (reward)$ *Sample average*

实际上，我们只能**有限采样t次**

$$Q_{t+1} = \frac{sum\ of\ rewards\ prior\ to\ t+1}{number\ of\ sample\ times\ prior\ to\ t+1}$$



中心极限定理+大数定律

估计期望奖励

批量式

$$Q_k = \frac{1}{k}\sum_{i=1}^{k} r_i$$

增量式

$$\begin{aligned} Q_k &= \frac{1}{k}\sum_{i=1}^{k} r_i \\ &= \frac{1}{k}\left(r_k + \sum_{i=1}^{k-1} r_i\right) \\ &= \frac{1}{k}(r_k + (k-1)Q_{k-1}) \\ &= \frac{1}{k}(r_k + kQ_{k-1} - Q_{k-1}) \\ &= Q_{k-1} + \frac{1}{k}[r_k - Q_{k-1}] \end{aligned}$$

探索与利用的平衡

由于事先不知道每个杆所对应的奖励，因此需要不断拉杆探索计算期望，但当次数是有限的时候，就需要去平衡探索和利用的次数，因为最终是想要奖励尽可能高，如果一直探索而不去选择一个最优的，就会使奖励偏低。

$\epsilon-Greedy$算法

完全贪婪算法即在每一时刻采取期望奖励估值最大的动作（拉动拉杆），这就是纯粹的利用，而没有探索，所以我们通常需要对完全贪婪算法进行一些修改，其中比较经典的一种方法为 ϵ-贪婪（ϵ-Greedy）算法。ϵ-贪婪算法在完全贪婪算法的基础上添加了噪声，每次以概率$1-\epsilon$选择以往经验中期望奖励估值最大的那根拉杆（利用），以概率 ϵ 随机选择一根拉杆（探索），公式如下：

$$a_t=\begin{cases}\arg\max_{a\in\mathcal{A}}\hat{Q}(a), & \text{采样概率：}1-\epsilon\\ \text{从 }\mathcal{A}\text{ 中随机选择}, & \text{采样概率：}\epsilon\end{cases}$$

随着探索次数的不断增加，我们对各个动作的奖励估计得越来越准，此时我们就没必要继续花大力气进行探索。所以在 ϵ-贪婪算法的具体实现中，我们可以令 ϵ 随时间衰减，即探索的概率将会不断降低。但是请注意，ϵ 不会在有限的步数内衰减至 0，因为基于有限步数观测的完全贪婪算法仍然是一个局部信息的贪婪算法，永远距离最优解有一个固定的差距。

上置信界算法（UCB）

上置信界（upper confidence bound，UCB）算法是一种经典的基于不确定性的策略算法，它的思想用到了一个非常著名的数学原理：**霍夫丁不等式**（Hoeffding's inequality）。在霍夫丁不等式中，令$X_1,\dots,X_n$为n个独立同分布的随机变量，取值范围为$[0,1]$，其经验期望为$\bar{x}_n=\frac{1}{n}\sum_{j=1}^{n}X_j$，则有

$$\mathbb{P}\left\{\mathbb{E}[X]\geq\bar{x}_n+u\right\}\leq e^{-2nu^2}$$

现在我们将霍夫丁不等式运用于多臂老虎机问题中。将$\hat{Q}_t(a)$代入 $\bar{x}_t$，不等式中的参数 $u=\hat{U}_t(a)$ 代表不确定性度量。给定一个概率 $p=e^{-2N_t(a)U_t(a)^2}$，根据上述不等式，$Q_t(a)<\hat{Q}_t(a)+\hat{U}_t(a)$ 至少以概率 $1-p$ 成立。当p很小时，$Q_t(a)<\hat{Q}_t(a)+\hat{U}_t(a)$就以很大概率成立，$\hat{Q}_t(a)+\hat{U}_t(a)$便是期望奖励上界。此时，上置信界算法便选取期望奖励上界最大的动作，即$a=\underset{a\in\mathcal{A}}{\arg\max}\left[\hat{Q}(a)+\hat{U}(a)\right]$。那其中$\hat{U}_t(a)$具体是什么呢？根据等式$e^{-2N_t(a)U_t(a)^2}$，解之即得 $\hat{U}_t(a)=\sqrt{\frac{-\log p}{2N_t(a)}}$。因此，设定一个概率$p$后，就可以计算相应的不确定性度量$\hat{U}_t(a)$了。更直观地说，UCB 算法在每次选择拉杆前，先估计每根拉杆的期望奖励的上界，使得拉动每根拉杆的期望奖励只有一个较小的概率p超过这个上界，接着选出期望奖励上界最大的拉杆，从而选择最有可能获得最大期望奖励的拉杆。