强化学习基础篇（三十五）探索与利用（Exploration and Exploitation

冯小龙 Garden001 2020-11-13

强化学习基础篇（三十五）探索与利用（Exploration and Exploitation）

1、探索与利用简介

在强化学习中，探索（Exploration ）的目的是找到更多有关环境的信息，而利用（Exploitation）的目的是利用已知的环境信息来最大限度地提高奖励。简而言之，探索是尝试还未尝试过的动作行为，而利用则是从已知动作中选择下一步的动作。

探索与利用之间的如何权衡，是强化学习的一个基本的问题。例如在很多情况，为了获得最佳的长期策略，可能需要做一些短期的牺牲。为了能够获得最佳的总体策略，往往需要收集到更多的信息。

有几种方式可以达到探索的目的：

- 第一种是朴素探索(Naive Exploration)，类似$\epsilon - greedy$算法在一定概率基础下随机选择一些action。
- 第二种是乐观初始估计(Optimistic Initialization)，优先选择当前被认为是最高价值的行为，除非新信息的获取推翻了该行为具有最高价值这一认知；
- 第三种是不确定优先(Optimism in the Face of Uncertainty):，更加倾向选择更加具有不确定性的状态/动作，这种方法就需要一种方法来衡量这种不确定性
- 第四种是概率匹配（Probability Matching): 根据当前估计的概率分布采样行为；
- 第五种是信息状态搜索(Information State Search)，将已探索的信息作为状态的一部分联合个体的状态组成新的状态，以新状态为基础进行前向探索。

根据搜索过程中使用的数据结构，可以将搜索分为：

- 依据状态行为空间的探索(State-Action Exploration)，其针对每一个当前的状态，以一定的算法尝试之前该状态下没有尝试过的行为。

- 参数化搜索（Parameter Exploration)。直接针对策略的函数近似，此时策略用各种形式的参数表达，探索即表现为尝试不同的参数设置。

  其优点是：得到基于某一策略的一段持续性的行为；

  其缺点是：对个体曾经到过的状态空间毫无记忆，也就是个体也许会进入一个之前曾经进入过的状态而并不知道其曾到过该状态，不能利用已经到过这个状态这个信息。

为了较简单的描述各类搜索的原理，下一节将使用一种与状态无关的Bandit来进行讲解。

2、与状态无关的k-bandit问题

k-bandit问题考虑的是如下的学习问题：你要重复地在k个选项或者动作中进行选择。每次做出选择后，都会得到一定数值的收益，收益由你选择的动作决定的平稳概率分布产生。目标是在某一段时间内最大化总收益的期望。

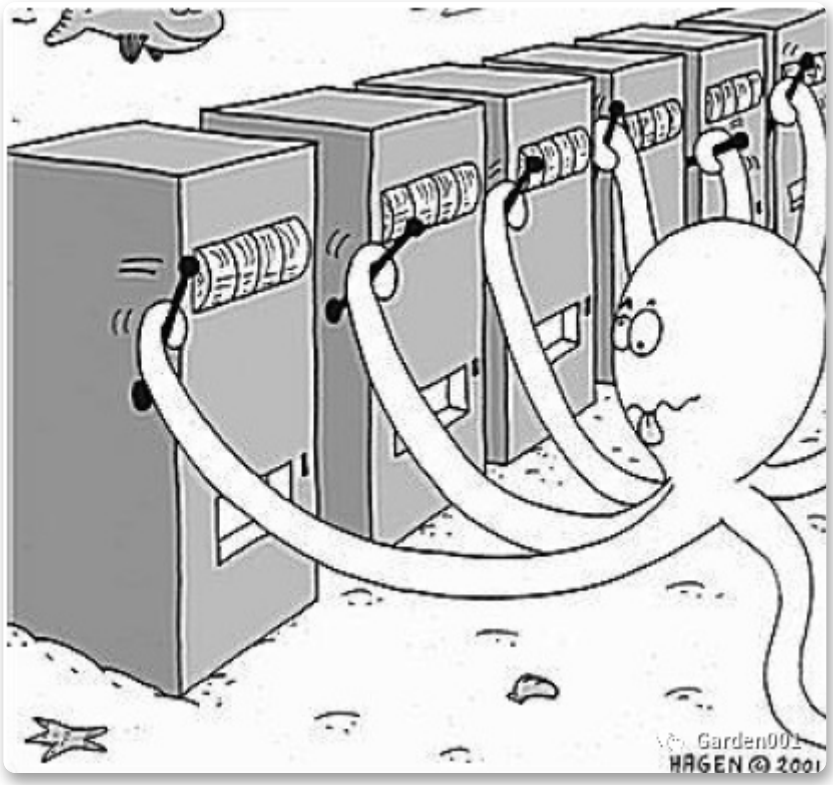


image.png

该问题定义是：

- multi-bandit可以看成是由行为空间和奖励组成的元组$，动作空间A是 m 个动作（即每个 $arms$ ），每一个行为对应拉下某一个拉杆。奖励

\mathcal{R}^{a}(r)=\mathbb{P}[r \mid a]$是未知的概率分布

- 在每个时间步t，智能体会选择动作$a_t \in A$，然后环境将生成奖励 $r_t \sim R^{a_t}$， 目标为最大化累计收益$\sum_{T-1}^{t} r_\tau$。

3、后悔值（Regret）

为了方便描述问题，我们先给出几个定义：

- action-value

  一个行为的价值等于该行为能得到的即时奖励期望，即该行为得到的所有即时奖励的平均值。

$$Q(a) = E[r|a]$$

- optimal value

  我们能够事先知道哪一个bandit能够给出最大即时奖励，那我们可以每次只选择对应的那个拉杆。如果用V^*表示这个最优价值，a^*表示能够带来最优价值的行为，则：

$$V^* = Q(a^*) = \max_{a \in \mathcal{A}} Q(a)$$

- Regret

  事实上我们不可能事先知道拉下哪个拉杆能带来最高奖励，因此每一次拉杆获得的即时奖励可能都与最优价值V^*存在一定的差距，我们定义这个差距为**「后悔值(regret):」**

$$l_t = \mathbb{E}[V^* - Q(a_t)]$$

- Total Regret

  每做出一个行为，都会产生一个后悔值,因此随着持续的拉杆行为，将所有的后悔值加起来，形成**「总后悔值(Total Regret)」**：

$$L_t = \mathbb{E}\left[\sum_{\tau=1}^{t} V^* - Q(a_\tau)\right]$$

这样，最大化累计奖励的问题就可以转化为最小化总后悔值了。之所以这样转换，是为了描述问题的方便，在随后的讲解中可以看到，较好的算法可以控制后悔值的

增加速度。而用最大化累计奖励描述问题不够方便直观。

4、Regret的推导

从另一个角度重写总后悔值。定义$N_t(a)$为到t时刻时已执行行为A的次数，定义差距（gap）Δa为最优动作a^*与行为a之间的差。那么总后悔值可以这样推导：

$$\begin{aligned} L_t &= \mathbb{E}\left[\sum_{\tau=1}^{t} V^* - Q(a_\tau)\right] \\ &= \sum_{a \in \mathcal{A}} \mathbb{E}[N_t(a)](V^* - Q(a)) \\ &= \sum_{a \in \mathcal{A}} \mathbb{E}[N_t(a)]\Delta_a \end{aligned}$$

这相当于把个行为的差距与该行为发生的次数乘起来，随后把行为空间的所有行为的这个乘积再相加得到，只不过这里是期望。

把总后悔值用计数和差距描述可以使我们理解到一个好的算法应该尽量减少那些差距较大的行为的次数。不过我们并不知道这个差距具体是多少，因为根据定义虽然最优价值V^*和每个行为的差距（gap）Δa为静态的，但我们并不清楚这两者的具体数值，我们所能使用的信息就是每次行为带来的即时奖励 r。那么我们如何利用每次行为的即时奖励呢？

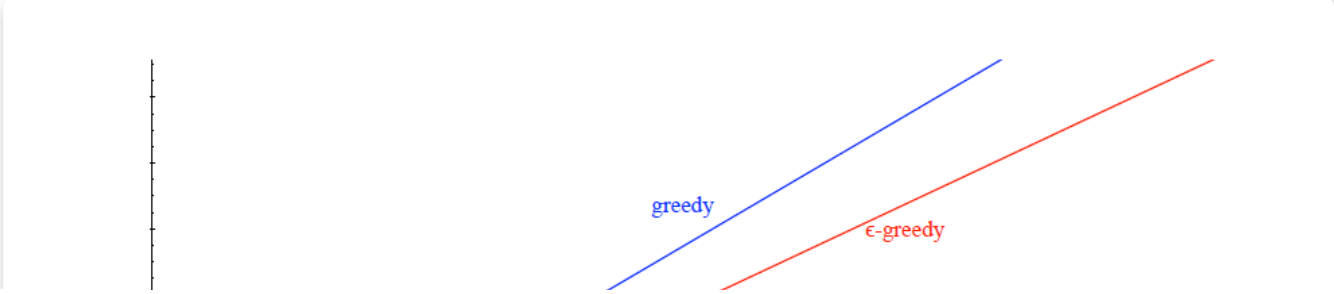
我们使用每次的即时奖励来计算得到t时刻止某一行为的平均价值：

$$\hat{Q}_t(a) = \frac{1}{N_t(a)} \sum_{t=1}^{T} r_t \mathbf{1}(a_t = a)$$

这个方法也叫蒙特卡罗评估，以此来近似该行为的实际价值 $Q(a)$：$\hat{Q}_t(a) \approx Q(a)$

5、Total Regret的直观理解

我们先直观了解下不同形式的随机策略其总后悔值随着时间的变化曲线：



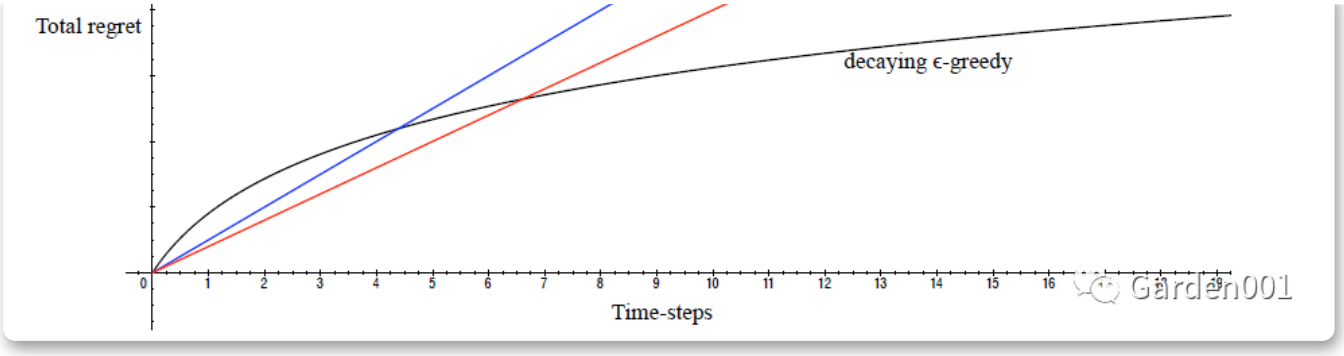


image.png

（1）对于$\epsilon - greedy$探索方法，总后悔值会呈线性增长，这是一个好的算法所不能接受的。这是因为每一个时间步，该探索方法有一定的几率选择最优行为，但同样也有一个固定小的几率采取完全随机的行为，如采取随机行为，那将一直会带来一定后悔值，如果持续以虽小但却固定的几率采取随机行为，那么总的后悔值会一直递增，导致呈现与时间之间的线性关系。类似的softmax探索方法与此类似。

（2）对于greedy探索方法，其总后悔值也是线性的，这是因为该探索方法的行为选择可能会锁死在一个不是最佳的行为上。

（3）目标就是找到一种探索方法，使用该探索方法时随着时间的推移其总后悔值增加得越来越少，后续将依次介绍几种较好的探索方法。