蒙特卡洛法

苏州大学计算机科学与技术学院
主讲：刘 全

目 录

引言

动态规划法：

- 基于模型的MDP问题求解方法；
- 当环境模型已知，动态规划法无需环境采样，只需通过迭代计算，就可以得到问题的最优策略；
- 无模型强化学习状态转移概率是未知的，无法利用动态规划方法求解值函数。
- 通过值函数的原始定义来求解无模型强化学习问题：

$$v_\pi(s) = \mathbb{E}_\pi(G_t \mid S_t = s)$$

$$q_\pi(s,a) = \mathbb{E}_\pi(G_t \mid S_t = s, A_t = a)$$

5.1 蒙特卡洛法的基本概念（1）

- **经验方法**

　　通过大量采样获取数据来进行学习

- **MC方法**

　　MC正是基于经验方法，在环境模型未知的情况下，采用时间步有限的、完整的情节，根据经验进行学习，并通过平均采样回报来解决强化学习问题。

5.1 蒙特卡洛法的基本概念（2）

5.1.1 MC的核心要素

- **经验：**

从环境交互中获得的(s,a,r)序列，它是情节的集合，也就是样本集。

✓真实经验

✓模拟经验

模拟经验是通过模拟模型得到的，这里的模拟模型只需生成状态转移的一些样本，无需像DP那样需要环境中所有可能的状态转移概率。

5.1 蒙特卡洛法的基本概念（3）

- **情节：**

一段经验可以分为多个情节，每一情节都是一个完整的(s,a,r)，即必有终止状态，形如：

$$s_0, a_0, r_1, .., s_{T-1}, a_{T-1}, r_T, s_T$$

经常与情节混淆的是轨迹，轨迹可以不存在终止状态，形如：

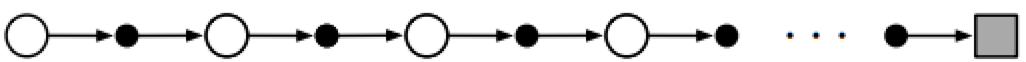
$$s_0, a_0, r_1, s_1, a_1, r_2, ...$$

序列 $\subseteq$ 情节 $\subseteq$ 经验（轨迹）

5.1 蒙特卡洛法的基本概念（4）

- **完整回报与目标值：**

因为只有到达终止状态才能计算回报，所以将情节的回报 G_t 称为完整回报，G_t 也称为MC的目标值。



5.1 蒙特卡洛法的基本概念（5）

5.1.2 MC的特点

- 无需知道状态转移概率 p，直接从环境中进行采样来处理无模型任务；
- 利用情节进行学习，并采用情节到情节（episode-by-episode）的离线学习（off-line）方式来求解最优策略 π_*。DP和后续介绍的时序差分算法则采用步到步（step-by-step）的在线学习（on-line）方式来求解最优策略；

5.1 蒙特卡洛法的基本概念（6）

✓离线学习：先完整地采集数据，然后以离线方式优化学习目标；

✓在线学习：边采集数据边优化学习目标。

- MC是一个非平稳问题，其表现在：某个状态采取动作之后的回报，取决于在同一个情节内后续状态所采取的动作，而这些动作通常是不确定的。如果说DP是在MDP模型中计算值函数，那么MC就是学习值函数。

5.1 蒙特卡洛法的基本概念（7）

- 在MC中，对每个状态的值函数估计都是独立的。对状态的值函数估计不依赖于其他任何状态，这也说明了MC不是自举过程；
- MC在估计每个状态的值函数时，其计算成本与状态总数无直接关，因为它只需要计算指定状态的回报，无需考虑所有的状态。

5.1 蒙特卡洛法的基本概念（8）

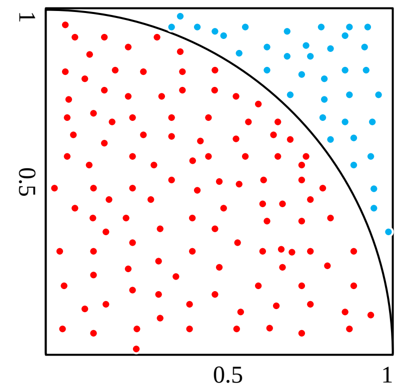
实际上，MC泛指任何包含大量随机成分的估计方法，通常利用采样数据来估算某一事件的发生概率。在数学领域中，它的应用可以用例5.1来说明。

例5.1 在边长为1米的正方形S_1内构建一个扇形S_2，利用扇形面积计算公式，可以计算出$S_2=\frac{1}{4}\pi r^2\approx\frac{1}{4}\times 3.14\times 1^2=0.785$。现在利用MC方法计算$S_2$的面积。均匀地向$S_1$内撒$n$个黄豆，经统计得知：有$m$个黄豆在$S_2$内部，那么有$\frac{S_2}{S_1}\approx\frac{m}{n}$，即$S_2\approx\frac{m}{n}S_1$，且$n$越大，计算得到的面积越精确。

5.1 蒙特卡洛法的基本概念（9）

在程序中分别设置 $n=100$、10000 和 1000000 共 3 组数据，统计结果如表所示。

n	S_1	S_2	m	m/n
100	1.0	0.785	79.0	0.79
10000	1.0	0.785	7839.0	0.7839
1000000	1.0	0.785	786018.0	0.786018



由实验数据可知，当 n 值越大时，得到的扇形面积越精确，即接近**0.785**。由此获得的**MC**采样公式为：

$$\mathbb{E}_{x\sim p}[f(x)]=\sum_{x}p(x)f(x)\approx\frac{1}{N}\sum_{x_i\sim p,i=1}^{N}f\left(x_i\right)$$

目　录

5.2 蒙特卡洛法预测（1）

根据状态值函数的初始定义，MC预测算法以情节中初始状态 s 的回报期望作为其值函数 $v_\pi(s)$ 的估计值，对策略 π 进行评估。在求解状态 s 的值函数时，先利用策略 π 产生 n 个情节 $s_0, a_0, r_1, .., s_{\mathrm{T}-1}, a_{\mathrm{T}-1}, r_{\mathrm{T}}, s_{\mathrm{T}}$ ，然后计算每个情节中状态 s 的折扣回报：

$$G_t^{(i)} = r_{t+1} + \gamma r_{t+2} + \ldots + \gamma^{\mathrm{T}-t\text{-}1} r_{\mathrm{T}}$$

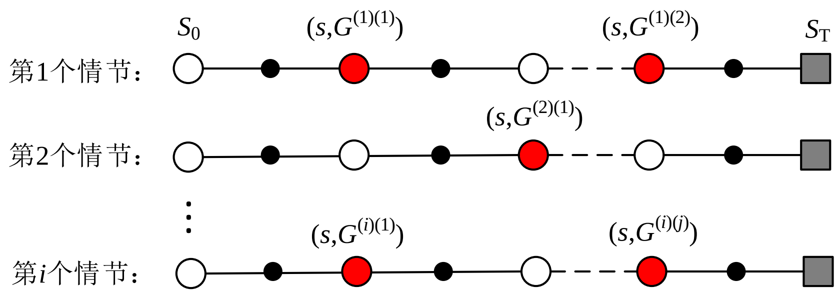
这里，$G_t^{(i)}$表示在第 i 个情节中，从 t 时刻到终点时刻T 的回报。该回报是基于某一策略下的状态值函数的无偏估计（由于 G_t 是真实获得的，所以属于无偏估计，但是存在高方差）。

5.2 蒙特卡洛法预测（2）

在MC中，每个回报都是对$v_\pi(s)$独立同分布的估计，通过对这些折扣回报求期望（均值）来评估策略π：

$$v_\pi(s)=\mathbb{E}_\pi(G_t \mid s\in S)=average(G_t^{(1)}+G_t^{(2)}+\cdots+G_t^{(i)}+\cdots+G_t^{(n)} \mid s\in S)$$

在一组采样（一个情节）中状态s可能多次出现，以更新图的方式表示，如下图所示。对同一情节中重复出现的状态s，有如下两种处理方法：



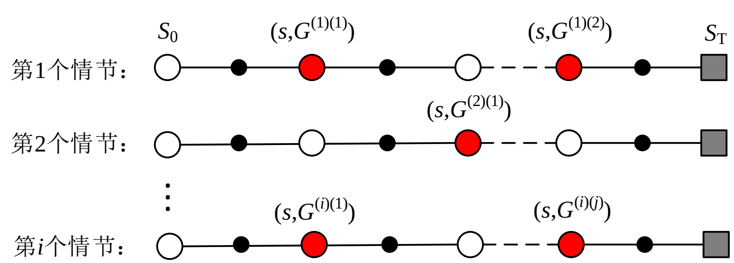
5.2 蒙特卡洛法预测（3）

- 首次访问（first-visit）：在对状态 s 的回报 $v_\pi(s)$ 进行估计时，只对每个情节中第1次访问到状态 s 的回报值作以统计：

$$V(s)=\frac{G_t^{(1)(1)}(s)+G_t^{(2)(1)}(s)+\ldots+G_t^{(i)(1)}(s)}{i}$$

- 每次访问（every-visit）：在对状态 s 的回报 $v_\pi(s)$ 进行估计时，对所有访问到状态 s 的回报值都作以统计：

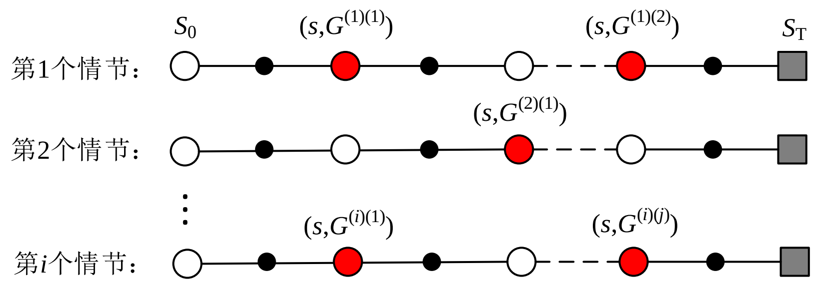
$$V(s)=\frac{G_t^{(1)(1)}(s)+G_t^{(1)(2)}(s)+\ldots+G_t^{(2)(1)}(s)+\ldots+G_t^{(i)(1)}(s)+\ldots+G_t^{(i)(j)}(s)}{N(s)}$$



5.2 蒙特卡洛法预测（4）

其中，i 表示第 i 个情节，j 表示第 j 次访问到状态 s ；$N(s)$ 表示状态 s 被访问过的总次数。根据大数定理，当MC采集的样本足够多时，计算出来的状态值函数估计值 $V_\pi(s)$ 就会逼近真实状态值函数 $v_\pi(s)$ 。

从右图中可以看出，MC更新图只显示在当前情节中，采样得到的状态转移，且包含了到该情节结束为止的所有状态转移；而DP更新图显示在当前状态下所有可能的状态转移，且仅包含单步转移。



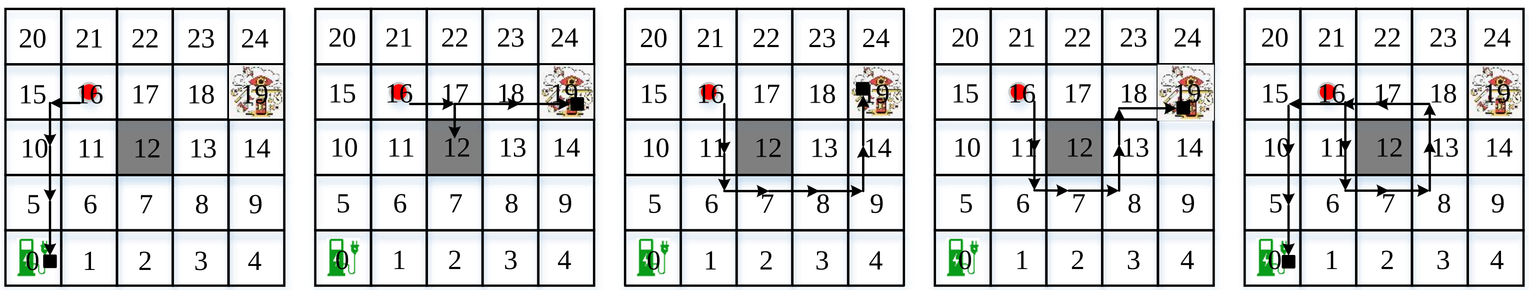
5.2 蒙特卡洛法预测（5）

基于首次访问的MC预测算法，如算法5.1所示。

算法 5.1 基于首次访问的 MC 预测算法
输　入：待评估的随机策略 $\pi(a\|s)$
初始化：
1.　对所有 $s \in \mathcal{S}$，初始化 $V(s) \in \mathbb{R}$， $V(s^{\mathrm{T}})=0$；状态 s 被统计的次数 $count(s)=0$
2.　**repeat** 对每一个情节 $k=0,1,2,\cdots$
3.　　根据策略 $\pi(a\|s)$，生成一个情节序列 $S_0,A_0,R_1,\cdots,S_{\mathrm{T}-1},A_{\mathrm{T}-1},R_{\mathrm{T}},S_{\mathrm{T}}$
4.　　$G \leftarrow 0$
5.　　**for** 本情节中的每一步 $t=\mathrm{T}-1$ **downto** 0 **do**
6.　　　$G \leftarrow \gamma G + R_{t+1}$
7.　　　**if** $S_t \notin \{S_0,S_1,\ldots,S_{t-1}\}$ **then**
8.　　　　$count(S_t) \leftarrow count(S_t)+1$
9.　　　　$V(S_t) \leftarrow V(S_t) + \frac{1}{count(S_t)}(G - V(S_t))$
10.　　　**end if**
11.　　**end for**
输　出：$v_\pi = V$

5.2 蒙特卡洛法预测（6）

例5.2 以例3.1扫地机器人为例。给出机器人经过下图的每条轨迹后，相对应的状态值。



1 **2** **3** **4** **5**

如图所示，选取了5个经过状态16的情节，5个情节依次设置为情节1、情节2、情节3、情节4和情节5。

5.2 蒙特卡洛法预测（7）

- **情节1：** $16 \to 15 \to 10 \to 5 \to 0$

$$G_{16}^{(1)(1)} = 0 + 0.8\times\left(0+0.8\times\left(0+0.8\times\left(1+0.8\times 0\right)\right)\right)=0.8^{3}\times 1 = 0.512$$

- **情节2：** $16 \to 17 \to 17 \to 18 \to 19$

$$G_{16}^{(2)(1)} = 0 + 0.8\times\left(-10+0.8\times\left(0+0.8\times\left(3+0.8\times 0\right)\right)\right)=0.8\times(-10)+0.8^{3}\times 3$$

$$= -8.464$$

- **情节3：** $16 \to 11 \to 6 \to 7 \to 8 \to 9 \to 14 \to 19$

$$G_{16}^{(3)(1)} = 0 + 0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(3+0.8\times 0\right)\right)\right)\right)\right)\right)$$

$$=0.8^{6}\times 3 = 0.786$$

5.2 蒙特卡洛法预测（8）

也可以直接利用关于回报的定义计算：

$$G_{16}^{(3)(1)} = 0 + 0.8 \times 0 + 0.8^2 \times 0 + 0.8^3 \times 0 + 0.8^4 \times 0 + 0.8^5 \times 0 + 0.8^6 \times 3$$

$$= 0.8^6 \times 3 = 0.786$$

- 情节4：$16 \rightarrow 11 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 13 \rightarrow 18 \rightarrow 19$

$$G_{16}^{(4)(1)} = 0 + 0.8 \times \Big(0 + 0.8 \times \Big(0 + 0.8 \times \Big(0 + 0.8 \times \Big(0 + 0.8 \times \Big(0 + 0.8 \times \big(3 + 0.8 \times 0\big)\Big)\Big)\Big)\Big)\Big)$$

$$= 0.8^6 \times 3 = 0.786$$

5.2 蒙特卡洛法预测（9）

- **情节5：**

  首次访问状态16：

$$16 \to 11 \to 6 \to 7 \to 8 \to 13 \to 18 \to 17 \to 16 \to 15 \to 10 \to 5 \to 0$$

$$\begin{aligned} G_{16}^{(5)(1)} &= 0 + 0.8 \times \\ &\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times\left(0+0.8\times(1+0.8\times 0)\right)\right)\right)\right)\right)\right)\right)\right)\right)\right) \\ &=0.8^{11}\times 1=0.086 \end{aligned}$$

  第2次访问状态16：

$$G_{16}^{(5)(2)} = 0 + 0.8\times\left(0+0.8\times\left(0+0.8\times\left(1+0.8\times 0\right)\right)\right)=0.8^{3}\times 1=0.512$$

5.2 蒙特卡洛法预测（10）

在情节5中状态16被访问了两次。利用首次访问的MC预测方法时，计算累计回报值只使用该情节中第一次访问状态16的回报，即 $G_{16}^{(5)(1)}$ 。所以使用这5条情节，利用首次访问方式计算状态16的估计值为：

$$V(S_{16})=(G_{16}^{(1)(1)}+G_{16}^{(2)(1)}+G_{16}^{(3)(1)}+G_{16}^{(4)(1)}+G_{16}^{(5)(1)})/5=-1.2588$$

而利用每次访问方式计算状态16的估计值为：

$$V(S_{16})=(G_{16}^{(1)(1)}+G_{16}^{(2)(1)}+G_{16}^{(3)(1)}+G_{16}^{(4)(1)}+G_{16}^{(5)(1)}+G_{16}^{(5)(2)})/6=-0.9637$$

目 录

5.3 蒙特卡洛评估（1）

- 由最优策略的两种求解方式可知，利用动作值函数是一种更适合于无模型求解最优策略的方法。
- **将估计状态值函数的MC预测问题转化为估计动作值函数的MC评估问题，对状态-动作对**(s,a)进行访问而不是对状态s进行访问。
- 根据策略π进行采样，记录情节中(s,a)的回报G_t，并对(s,a)的回报求期望，得到策略π下的动作值函数$q_\pi(s,a)$的估计值。
- **MC评估方法对每一组状态-动作对** (s,a) 的评估方法也分为首次访问和每次访问两种。

5.3 蒙特卡洛评估（2）

- 为了保证算法中值函数和策略的收敛性，**DP**算法会对所有状态进行逐个扫描。在**MC**评估方法中，根据动作值函数计算的性质，必须保证每组状态-动作对(s,a)都能被访问到，即得到所有(s,a)的回报值，才能保证样本的完备性。
- 针对该问题，我们设定探索始点（**exploring starts**）：每一组(s,a)都有非**0**的概率作为情节的起始点(s_0,a_0)。

5.3 蒙特卡洛评估（3）

- 实际上，探索始点在实际应用中是难以达成的，需要配合无限采样才能保证样本的完整性。
- 通常的做法是采用那些在每个状态下所有动作都有非0概率被选中的随机策略。
- 这里我们先从简单的满足探索始点的MC控制算法开始讨论，然后引出基于同策略和异策略方法的MC控制算法。

目 录

5.4 蒙特卡洛控制（1）

预测和控制的思想是相同的，它们都是基于带奖励过程的马尔可夫链来对目标进行更新，其区别在于：

✓**MC预测：**求解在给定策略 π 下，状态 s 的状态值函数 $v_\pi(s)$ 。

✓**MC控制：**基于**GPI**，包含策略评估和策略改进两部分。这里的策略评估是求解在给定策略π下，状态-动作对(s,a)的动作值函数$q_\pi(s,a)$。其策略迭代过程如下所示：

$$\pi_0 \xrightarrow{\text{E}} q_{\pi_0} \xrightarrow{\text{I}} \pi_1 \xrightarrow{\text{E}} q_{\pi_1} \xrightarrow{\text{I}} \cdots \xrightarrow{\text{I}} \pi_* \xrightarrow{\text{E}} q_*$$

5.4 蒙特卡洛控制（2）

5.4.1 基于探索始点的蒙特卡洛控制

当采样过程满足探索始点时，对任意策略 π_k，MC算法都可以准确地计算出指定 (s,a) 的动作值函数。一旦得到了动作值函数，可以直接利用基于动作值函数的贪心策略来对策略进行更新。此时对于所有 $s \in \mathcal{S}$，任意策略 π_k 以及更新后的 π_{k+1} 都将满足策略改进原理：

$$
\begin{aligned}
q_{\pi_k}\left(s,\pi_{k+1}(s)\right) &= q_{\pi_k}\left(s,\arg\max_a q_{\pi_k}(s,a)\right) && \text{根据 } \pi_{k+1}(s)=\arg\max_a q_{\pi_k}(s,a) \\
&= \max_a q_{\pi_k}(s,a) && \text{将最优策略提到公式外面} \\
&\geq q_{\pi_k}\left(s,\pi_k(s)\right) && \text{使用上一个策略的动作值函数} \\
&\geq v_{\pi_k}(s)
\end{aligned}
$$

5.4 蒙特卡洛控制（3）

上式表明，采用基于动作值函数的贪心策略，改进后的策略 π_{k+1} 一定优于或等价于 π_k 。这一过程保证了MC控制算法能够收敛到最优动作值函数和最优策略。

无穷采样假设，使用基于探索始点的蒙特卡洛（Monte Carlo based on Exploring States，MCES）控制算法来进行规避。MCES控制算法通过情节到情节的方式对策略进行评估和更新，每一情节结束后，使用观测到的回报进行策略评估，然后在该情节访问到的每一个状态上进行策略改进。

5.4 蒙特卡洛控制（4）

MC控制算法主要分为以下两个阶段：

✓策略评估：根据当前策略 π 进行采样，得到一条完整情节，估计每一组(s,a) 的动作值函数；

✓策略改进：基于动作值函数$q_\pi(s,a)$，直接利用贪心策略对策略进行改进。

5.4 蒙特卡洛控制（5）

算法5.2给出了基于探索始点的蒙特卡洛控制——MCES算法。

算法 5.2 MCES 控制算法

输　入：待评估的确定策略 $\pi(s)$

初始化：

1.　对所有 $s \in \mathcal{S}^{+}$， $a \in \mathcal{A}(s)$ ，初始化 $Q(s,a) \in \mathbb{R}$， $Q(s^{\mathrm{T}},a)=0$

2.　对所有 $s \in \mathcal{S}^{+}$， $a \in \mathcal{A}(s)$ ，状态-动作对 (s,a) 被统计的次数 $count(s,a)=0$

3.　**repeat** 对每一个情节 $k=0,1,2,\cdots$

4.　　　以非 0 概率随机选取初始状态-动作对 (S_0,A_0)

5.　　　根据策略 $\pi(s)$，从初始状态-动作对 (S_0,A_0) 开始，生成一个情节序列：

$$S_0,A_0,R_1,\cdots,S_{\mathrm{T}-1},A_{\mathrm{T}-1},R_{\mathrm{T}},S_{\mathrm{T}}$$

6.　　　$G \leftarrow 0$

7.　　　**for** 本情节中的每一步 $t=\mathrm{T}-1$ **downto** 0 **do**

8.　　　　　$G \leftarrow \gamma G + R_{t+1}$

9.　　　　　**if** $S_t,A_t \notin \{S_0,A_0,S_1,A_1,\ldots,S_{t-1},A_{t-1}\}$ **then**

10.　　　　　　　$count(S_t,A_t) \leftarrow count(S_t,A_t)+1$

11.　　　　　　　$Q(S_t,A_t) \leftarrow Q(S_t,A_t)+\dfrac{1}{count(S_t,A_t)}\big(G-Q(S_t,A_t)\big)$

12.　　　　　**end if**

13.　　　　　$\pi(S_t) \leftarrow \arg\max_{a} Q(S_t,a)$

14.　　　**end for**

输　出：$\pi_{*}=\pi$

5.4 蒙特卡洛控制（6）

- 虽然可以通过探索性始点来弥补无法访问到所有状态-动作对的缺陷，但这一方法并不合理，唯一普遍的解决方法就是保证Agent能够持续不断地选择所有可能的动作，这也称为无探索始点方法，该方法分为同策略方法与异策略方法两种。

5.4 蒙特卡洛控制（7）

5.4.2 同策略蒙特卡洛控制

- 在同策略MC控制算法中，策略通常是软性的（soft），即对于所有的 $s \in \mathcal{S}$、$a \in \mathcal{A}(s)$，均有 $\pi(a|s) > 0$。
- 策略都必须逐渐变得贪心，以逐渐逼近一个确定性策略。
- 同策略方法使用 ε-贪心策略（ε-greedy policy），其公式为：

$$\pi(a|s) \leftarrow \begin{cases} 1-\varepsilon+\dfrac{\varepsilon}{|\mathcal{A}(s)|} & \text{当} a = A^* \text{时，以概率} 1-\varepsilon+\dfrac{\varepsilon}{|\mathcal{A}(s)|} \text{选择具有最大价值的动作} \\ \dfrac{\varepsilon}{|\mathcal{A}(s)|} & \text{当} a \neq A^* \text{时，以概率} \dfrac{\varepsilon}{|\mathcal{A}(s)|} \text{随机选择动作} \end{cases}$$

5.4 蒙特卡洛控制（8）

- GPI并没有要求必须使用贪心策略，只要求采用的优化方法逐渐逼近贪心策略即可。
- 根据策略改进定理，对于一个ε -软性策略π ，任何根据q_π生成的ε-贪心策略都是对其所做的改进。下面对ε-软性策略改进定理进行证明。

5.4 蒙特卡洛控制（9）

➢假设 π' 为 ε**-greedy**策略，对任意状态 $s \in \mathcal{S}$ 有：

$$
\begin{aligned}
q_\pi(s,\pi'(s)) &= \sum_a \pi'(a|s) q_\pi(s,a) \\
&= \frac{\varepsilon}{|\mathcal{A}(s)|}\sum_a q_\pi(s,a) + (1-\varepsilon)\max_a q_\pi(s,a) \\
&\geq \frac{\varepsilon}{|\mathcal{A}(s)|}\sum_a q_\pi(s,a) + (1-\varepsilon)\frac{\pi(a|s)-\frac{\varepsilon}{|\mathcal{A}(s)|}}{1-\varepsilon} q_\pi(s,a) \\
&= \frac{\varepsilon}{|\mathcal{A}(s)|}\sum_a q_\pi(s,a) - \frac{\varepsilon}{|\mathcal{A}(s)|}\sum_a q_\pi(s,a) + \sum_a \pi(a|s) q_\pi(s,a) \\
&= v_\pi(s)
\end{aligned}
$$

所以，根据策略改进定理，$\pi' \geq \pi$ 。

5.4 蒙特卡洛控制（10）

基于同策略首次访问ε-greedy策略的MC算法，如算法5.3所示。

算法 5.3 基于同策略首次访问 ε-greedy 策略的 MC 算法
输　入：待评估的 ε-greedy 策略 $\pi(a\|s)$
初始化：
1. 对所有 $s \in \mathcal{S}^+$， $a \in \mathcal{A}(s)$ ，初始化 $Q(s,a) \in \mathbb{R}$， $Q(s^{\mathrm{T}},a)=0$
2. 对所有 $s \in \mathcal{S}^+$， $a \in \mathcal{A}(s)$ ，状态-动作对 (s,a) 被统计的次数 $count(s,a)=0$
3. $\varepsilon \leftarrow (0,1)$ 为一个逐步递减的较小的实数
4. **repeat** 对每一个情节 $k=0,1,2,\cdots$
5. 　　根据策略 $\pi(a\|s)$，生成一个情节序列 $S_0,A_0,R_1,\cdots,S_{\mathrm{T}-1},A_{\mathrm{T}-1},R_{\mathrm{T}},S_{\mathrm{T}}$
6. 　　$G \leftarrow 0$
7. 　　**for** 本情节中的每一步 $t=\mathrm{T}-1$ **downto** 0 **do**

5.4 蒙特卡洛控制（11）

$G \leftarrow \gamma G + R_{t+1}$

if $S_t, A_t \notin \{S_0, A_0, S_1, A_1, \ldots, S_{t-1}, A_{t-1}\}$ **then**

$count(S_t, A_t) \leftarrow count(S_t, A_t) + 1$

$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \dfrac{1}{count(S_t, A_t)}\big(G - Q(S_t, A_t)\big)$

$A^* \leftarrow \arg\max_a Q(S_t, a)$

for $a \in \mathcal{A}(S_t)$ **do**

if $a == A^*$ **then**

$\pi(a \mid S_t) \leftarrow 1 - \varepsilon + \dfrac{\varepsilon}{|A(S_t)|}$

else

$\pi(a \mid S_t) \leftarrow \dfrac{\varepsilon}{|A(S_t)|}$

end if

end for

end if

end for

输　出：$q_* = Q$；　$\pi_* = \pi$

5.4 蒙特卡洛控制（12）

例5.3 利用同策略首次访问ε-贪心策略MC算法，给出扫地机器人的最优策略。扫地机器人通过多次实验，不断的更新Q值，最终收敛到最优策略，并得到一条回报最大的路径。同策略蒙特卡洛首次访问控制算法中，对动作值函数Q的计算，也是通过对每一情节中第一次访问到的该状态-动作对的回报进行平均，然后选择使该动作值函数Q最大的动作，作为该状态下应该采取的动作。表5.1给出5个代表性状态，基于同策略首次访问MC算法的扫地机器人最优策略的求解过程。

5.4 蒙特卡洛控制（13）

表5.1 基于同策略首次访问MC算法的扫地机器人最优策略计算过程（$\varepsilon_0 = 0.1$）

	S_5	S_{10}	S_{18}	S_{20}	S_{24}
Q_0	0.00;0.00;*.**;0.00	0.00;0.00;*.**;0.00	0.00;0.00;0.00;0.00	*.**;0.00;*.**;0.00	*.**;0.00;0.00;*.**
π_0	0.933;0.033;0.00;0.033	0.033;0.933;0.00;0.033	0.025;0.025;0.925;0.025	0.00;0.950;0.00;0.050	0.00;0.950;0.050;0.00
A_0	1;0;0;0	0;1;0;0	0;0;1;0	0;1;0;0	0;1;0;0
Q_1	0.00;0.00;*.**;0.00	-0.07;0.00;*.**;0.00	0.00;0.00;0.00;3.00	*.**;0.05;*.**;-4.14	*.**;0.00;0.00;*.**
π_1	0.933;0.033;0.00;0.033	0.033;0.933;0.00;0.033	0.025;0.025;0.025;0.925	0.00;0.950;0.00;0.050	0.00;0.950;0.050;0.00
A_1	1;0;0;0	0;1;0;0	0;0;0;1	0;1;0;0	0;1;0;0
⋮	⋮	⋮	⋮	⋮	
Q_{7500}	0.29;1.00;*.**;0.27	0.05;0.63;*.**;0.58	1.56;0.53;0.38;3.00	*.**;0.35;*.**;0.72	*.**;3.00;1.59;*.**
π_{7500}	0.033;0.933;0.00;0.033	0.033;0.933;0.00;0.033	0.025;0.025;0.025;0.925	0.00;0.050;0.00;0.950	0.00;0.950;0.050;0.00
A_{7500}	0;1;0;0	0;1;0;0	0;0;0;1	0;0;0;1	0;1;0;0

5.4 蒙特卡洛控制（14）

⋮	⋮	⋮	⋮	⋮	
Q_{12500}	0.31;1.00;*.**;0.30	0.11;0.65;*.**;-0.61	1.60;0.67;0.58;3.00	*.**;0.40;*.**;0.81	*.**;3.00;1.64;*.**
π_{12500}	0.033;0.933;0.00;0.033	0.033;0.933;0.00;0.033	0.025;0.025;0.025;0.925	0.00;0.050;0.00;0.950	0.00;0.950;0.050;0.00
A_{12500}	0;1;0;0	0;1;0;0	0;0;0;1	0;0;0;1	0;1;0;0
⋮	⋮	⋮	⋮	⋮	
Q_{19999}	0.31;1.00;*.**;0.30	0.11;0.66;*.**;-0.60	1.61;0.67;0.64;3.00	*.**;0.43;*.**;0.94	*.**;3.00;1.67;*.**
π_{19999}	0.033;0.933;0.00;0.033	0.033;0.933;0.00;0.033	0.025;0.025;0.025;0.925	0.00;0.050;0.00;0.950	0.00;0.950;0.050;0.00
A_{19999}	0;1;0;0	0;1;0;0	0;0;0;1	0;0;0;1	0;1;0;0
⋮	⋮	⋮	⋮	⋮	
Q_{20000}	0.31;1.00;*.**;0.30	0.11;0.66;*.**;-0.60	1.61;0.67;0.64;3.00	*.**;0.43;*.**;0.94	*.**;3.00;1.67;*.**
π_{20000}	0.033;0.933;0.00;0.033	0.033;0.933;0.00;0.033	0.025;0.025;0.025;0.925	0.00;0.050;0.00;0.950	0.00;0.950;0.050;0.00
A_{20000}	0;1;0;0	0;1;0;0	0;0;0;1	0;0;0;1	0;1;0;0
π_*	0;1;0;0	0;1;0;0	0;0;0;1	0;0;0;1	0;1;0;0

5.4 蒙特卡洛控制（15）

另外，常用ε-柔性策略公式还有以下**4**种：

✓随机贪心策略：基于随机数，用一个较小的阈值ε来控制策略的探索性：

$$\pi(a \mid S_t) \leftarrow \begin{cases} A^* & \text{如果 random( )} > \varepsilon \\ \text{rand}(A) & \text{如果 random( )} \le \varepsilon \end{cases}$$

当随机数大于ε时，选择最大动作值函数对应的动作；当随机数小于或等于ε时，随机地选择动作$rand(A)$。

5.4 蒙特卡洛控制（15）

✓ Boltzmann探索：定义t 时刻选择动作A_t 的概率，其公式为：

$$\pi(A_t \mid S_t) = \frac{e^{Q_t(s_t, A_t)/\tau_t}}{\sum_a e^{Q_t(s_t, a)/\tau_t}}$$

其中，$\tau_t \geq 0$表示温度参数，控制探索的随机性。当$\tau_t \to 0$时，选择贪心动作；当$\tau_t \to \infty$时，随机选择动作。

5.4 蒙特卡洛控制（16）

✓**最大置信上界法：**在选择动作时，一方面要考虑其估计值最大，另外一方面也要考虑探索长时间没有访问到的动作，以免错过更好的动作。

$$A_t = \arg\max_a \left[Q_t(s,a) + c\sqrt{\frac{\ln t}{N_t(s,a)}} \right]$$

其中，$\ln t$ 表示 t 的自然对数，$N_t(s,a)$ 表示当前状态 s ，在时刻 t 之前动作 a 被选择的次数。c 是一个大于**0**的数，用来控制探索的程度。如果 $N_t(s,a)=0$，则动作 a 就被认为是当前状态 s 下满足最大化条件的动作。

5.4 蒙特卡洛控制（17）

- ✓ 乐观初始值方法。给值函数赋予一个比实际价值大得多些的乐观初始值。这种乐观估计会鼓励不断地选取收益接近估计值的动作。但无论选取哪一种动作，收益都比最初始的估计值小，因此在估计值收敛之前，所有动作都会被多次尝试。既使每一次都按照贪心法选择动作，系统也会进行大量的探索。

5.4 蒙特卡洛控制（18）

5.4.3 异策略与重要性采样

- 异策略MC方法：常用的无探索始点蒙特卡洛方法。
- 重要性采样，因为几乎所有的异策略方法都使用到重要性采样。
- 重要性采样：利用来自其他分布的样本，估计当前某种分布期望值的通用方法。

5.4 蒙特卡洛控制（19）

重要性采样

以离散型数据为例，假设 $f(x)$ 是一个服从 $p(x)$ 分布的函数，其期望公式为：$\mathbb{E}_{x\sim p}[f(x)]=\sum_{x}p(x)f(x)$

其中，$x\sim p$ 表示 x 服从 $p(x)$ 分布，也可记为 $f(x)\sim p$ 。

通常情况下，可以在服从 $p(x)$ 分布的离散型数据中进行采样，得到样本集 $\{x_1,x_2,\ldots,x_N\}$，则 $f(x)$ 在 $p(x)$ 分布下的期望为：

$$\mathbb{E}_{x\sim p}[f(x)]\approx\frac{1}{N}\sum_{x_i\sim p,i=1}^{N}f(x_i)$$

5.4 蒙特卡洛控制（20）

在有些任务中，为了得到分布函数$p(x)$，需要采集大量的样本才能拟合原期望，或存在部分极端、无法代表分布的样本。针对这些任务，在服从$p(x)$分布的数据中采样存在困难的问题，根据重要性采样原则，可以将该任务转化为从服从简单分布$q(x)$的数据中进行采样，得到的样本集为$\{x_1, x_2, \ldots, x_N\}$。此时$f(x)$在$p(x)$分布下的期望为：

$$\mathbb{E}_{x\sim p}[f(x)] = \sum_x p(x) f(x) = \sum_x q(x)\left[\frac{p(x)}{q(x)} f(x)\right] = \mathbb{E}_{x\sim q}\left[\frac{p(x)}{q(x)} f(x)\right]$$

其中，$\mathbb{E}_{x\sim q}\left[\frac{p(x)}{q(x)} f(x)\right]$表示函数$\frac{p(x)}{q(x)} f(x)$在$q(x)$分布下的期望。

5.4 蒙特卡洛控制（21）

根据MC采样思想，在采样数据足够多时，$f(x)$在 $p(x)$分布下的期望近似为：

$$\mathbb{E}_{x\sim p}[f(x)]=\mathbb{E}_{x\sim q}\left[\frac{p(x)}{q(x)}f(x)\right]\approx\frac{1}{N}\sum_{x_i\sim q,i=1}^{N}\frac{p(x_i)}{q(x_i)}f(x_i)=\frac{1}{N}\sum_{x_i\sim q,i=1}^{N}\omega(x_i)f(x_i)$$

这里$\omega(x)$为重要性采样比率（importance-sampling ratio），有$\omega(x)=p(x)/q(x)$。

由此，我们将求解$f(x)$在$p(x)$分布下的函数期望问题，转换为求解包含重要性采样比率ω的$f(x)$在$q(x)$分布下的函数期望。

5.4 蒙特卡洛控制（22）

重要性采样的特点：

✓ $q(x)$ 与 $p(x)$ 具有相同的定义域。

✓采样概率分布 $q(x)$ 与原概率分布 $p(x)$ 越接近，方差越小；反之，方差越大。

通常采用加权重要性采样来减小方差，即用 $\sum_{j=1}^{N}\omega(x_j)$

替换 N：

✓ 普通重要性采样的函数估计为：$\mathbb{E}_{x\sim p}[f(x)]\approx\sum_{x_i\sim q,i=1}^{N}\omega(x_i)f(x_i)/N$

✓ 加权重要性采样的函数估计为：$\mathbb{E}_{x\sim p}[f(x)]\approx\sum_{x_i\sim q,i=1}^{N}\omega(x_i)f(x_i)/\sum_{j=1}^{N}\omega(x_j)$

5.4 蒙特卡洛控制（23）

➢ 基于重要性采样的异策略方法

异策略方法目标策略和行为策略是不同的。这里假设目标策略为 π，行为策略为 b，所有情节都遵循行为策略 b，并利用行为策略 b 产生的情节来评估目标策略。这样需要满足覆盖条件，即目标策略 π 中的所有动作都会在行为策略 b 中被执行。也就是说，所有满足 $\pi(a|s)>0$ 的 (s,a) 均有 $b(a|s)>0$ 。根据轨迹在两种策略下产生的相对概率来计算目标策略 π 的回报值，该相对概率称为重要性采样比率，

5.4 蒙特卡洛控制（24）

记为 ρ。

以 S_t 作为初始状态，其采样得到的后续状态-动作对序列为：$A_t,\ S_{t+1},\ldots,S_{\mathrm{T}-1},A_{\mathrm{T}-1},S_{\mathrm{T}}$。在任意目标策略 π 下发生的概率如下所示：

$$P(A_t,\ S_{t+1},A_{t+1}\ ,\ldots,S_{\mathrm{T}}\left|S_t,A_{t:\mathrm{T}-1}\sim\pi\right.)=\pi(A_t\,|\,S_t)\,p(S_{t+1}\,|\,S_{t+1},A_t)\pi(A_{t+1}\,|\,S_{t+1})\ldots p(S_{\mathrm{T}}\,|\,S_{\mathrm{T}-1},A_{\mathrm{T}-1})$$
$$=\prod_{k=t}^{\mathrm{T}-1}\pi(A_k\,|\,S_k)\,p(S_{k+1}\,|\,S_k\,,A_k)$$

在任意行为策略 b 下发生后的概率如下所示：

$$P(A_t,\ S_{t+1},A_{t+1}\ ,\ldots,S_{\mathrm{T}}\left|S_t,A_{t:\mathrm{T}-1}\sim b\right.)=b(A_t\,|\,S_t)\,p(S_{t+1}\,|\,S_{t+1},A_t)\pi(A_{t+1}\,|\,S_{t+1})\ldots p(S_{\mathrm{T}}\,|\,S_{\mathrm{T}-1},A_{\mathrm{T}-1})$$
$$=\prod_{k=t}^{\mathrm{T}-1}b(A_k\,|\,S_k)\,p(S_{k+1}\,|\,S_k\,,A_k)$$

5.4 蒙特卡洛控制（25）

其中，p 表示状态转移概率，T 是该情节的终止时刻。注意：公式累乘符号的上标为 $\mathrm{T}-1$，因为最后一个动作发生在 $\mathrm{T}-1$ 时刻。$S_t, A_{t:\mathrm{T}-1} \sim \pi$ 表示该情节服从目标策略 π。

这样某一情节在目标策略和行为策略下发生的相对概率为：

$$\rho_{t:\mathrm{T}-1} = \frac{\prod_{k=t}^{\mathrm{T}-1} \pi(A_k \mid S_k) p(S_{k+1} \mid S_k, A_k)}{\prod_{k=t}^{\mathrm{T}-1} b(A_k \mid S_k) p(S_{k+1} \mid S_k, A_k)} = \prod_{k=t}^{\mathrm{T}-1} \frac{\pi(A_k \mid S_k)}{b(A_k \mid S_k)}$$

5.4 蒙特卡洛控制（26）

其中，$\rho_{t:T-1}$ 表示某一情节从 t 到 T 时刻的重要性采样比率，也就是基于两种策略采取动作序列 $A_t, A_{t+1}, \ldots, A_{T-1}$ 的相对概率，与重要性采样比率 $\omega(x)$ 相对应。从上式可以看出，尽管情节的形成依赖于状态转移概率 p，但由于分子分母中同时存在 p，可以被消去，所以重要性采样比率仅仅依赖于两个策略，而与状态转移概率无关。

行为策略中的回报期望是不能直接用于评估目标策略的。根据重要性采样原则，需要使用比例系数 $\rho_{t:T-1}$ 对回报进行调整，使其矫正为正确的期望值：

5.4 蒙特卡洛控制（27）

$$\mathbb{E}[G_t \mid S_t = s] = v_b(s) \quad \Rightarrow \quad v_\pi(s) = \mathbb{E}[\rho_{t:\mathrm{T}-1} G_t \mid S_t = s]$$

假设遵循行为策略 b 采样得到了一系列情节。为方便计算，将这些情节首位相连，并按时刻状态出现的顺序进行编号。例如第1个情节在时刻100状态结束，则第2个情节的编号就在时刻101状态开始，以此类推。在每次访问方法中，存储所有访问过状态 s 的时间步，记为 $\mathcal{T}(s)$，并以 $|\mathcal{T}(s)|$ 表示状态 s 被访问过的总次数。在首次访问方法中，$\mathcal{T}(s)$ 只包括这些情节中第一次访问到 s 的时间步。此外，以 $\mathrm{T}(t)$ 表示在 t 时刻后的第一个终止时刻，以 G_t 表示从 t 到 $\mathrm{T}(t)$ 时刻的回报，

5.4 蒙特卡洛控制（28）

以 $\rho_{t:\mathrm{T}(t)-1}$ 表示回报 G_t 的重要性采样比率（在增量式计算中常简写为 W_i）。

根据重要性采样思想，状态值函数的计算方法分为两种：

✓ 普通重要性采样（ordinary importance sampling）

将回报按照权重缩放后进行平均。属于无偏估计，具有方差无界的特点。其计算如下所示：

$$V(s)=\frac{\sum_{t\in\mathcal{T}(s)}\rho_{t:\mathrm{T}(t)-1}G_t}{\left|\mathcal{T}(s)\right|}$$

5.4 蒙特卡洛控制（29）

✓ 加权重要性采样（weighted importance sampling）

将回报进行加权平均。属于有偏估计，具有方差较小的特点。其计算如下所示：

$$V(s)=\frac{\sum_{t\in\mathcal{T}(s)}\rho_{t:\mathrm{T}(t)-1}G_t}{\sum_{t\in\mathcal{T}(s)}\rho_{t:\mathrm{T}(t)-1}}$$

分母为**0**时，$V(s)=0$ 。

两种方法的主要差异在于偏差和方差的不同。

5.4 蒙特卡洛控制（30）

✓ 普通重要性采样的偏差与方差

采用某种方法估计值函数，当估计结果的期望恒为$v_{\pi}(s)$时，该方法是无偏估计，但其方差可能是无界的。当 $\rho=10$ 时，表明该轨迹在目标策略下发生的可能性是行为策略下的**10**倍，$V(s)=10G_t$ ，根据普通重要性采样得到的估计值 $V_{\pi}(x)$ 是回报值的**10**倍，这就存在了高方差。

5.4 蒙特卡洛控制（31）

✓ 加权重要性采样的偏差与方差

由于比例系数ρ被消去，所以加权重要性采样的估计值就等于回报值，与重要性采样比例无关。因为该回报值是仅有的观测结果，所以是一个合理的估计，但它的期望是$v_b(s)$而非 $v_\pi(s)$ ，所以该方法属于有偏估计。此外，由于加权估计中回报的最大权重是1，所以其方差会明显低于普通估计。

由于重要性采样比率涉及到所有状态的转移概率，因此有很高的方差，从这一点来说，MC算法不太适合于处理异策略问题。异策略MC只有理论研究价值，实际应用的效果并不明显，难以获得最优动作值函数。

5.4 蒙特卡洛控制（32）

➤ 经典的增量式计算

假设有一组实数数据，其形式为：$x_1, x_2, \cdots, x_k, x_{k+1}, \cdots$ 。

令x_k 为第 k 个数据的数值，u_k为前 k 个数据的平均值，即有：

$$u_k = \frac{1}{k}\sum_{i=1}^{k} x_i$$

根据数学中的迭代思想，引入增量式计算方法，以简化求解过程。增量式推导如下所示：

$$\begin{aligned} u_k &= \frac{1}{k}\sum_{i=1}^{k} x_i \\ &= \frac{1}{k}\left(x_k + \sum_{i=1}^{k-1} x_i\right) \\ &= \frac{1}{k}\left(x_k + (k-1)u_{k-1}\right) \\ &= u_{k-1} + \frac{1}{k}\left(x_k - u_{k-1}\right) \end{aligned}$$

5.4 蒙特卡洛控制（33）

- 根据上式的规律，构建经典增量式公式：

$$\text{NewEstimate} \leftarrow \text{OldEstimate} + \text{StepSize}\left(\text{Target} - \text{OldEstimate}\right)$$

- 将**Target**称为目标值，从单次更新过程来看，**OldEstimate**是朝着**Target**移动的。从整个更新结果来看，**OldEstimate**是朝着真实目标值移动的。在自举方法中，**Target**也常被称为自举估计值。

- 增量式计算方法是一种基于样本**Target**的随机近似过程，拆分了均值求解过程，减少了存储消耗，简化了计算过程。

5.4 蒙特卡洛控制（34）

➢ MC的增量式

✓同策略MC：使用传统增量式计算公式，不涉及重要性采样，t 时刻状态 S_t 的状态值函数更新递归式为：

$$V(S_t) \leftarrow V(S_t) + \alpha\left(G_t - V(S_t)\right)$$

◆公式右侧的$V(S_t)$为历史状态值函数的均值，表示估计值，即**OldEstimate**；

◆G_t为 t 时刻的回报，表示目标值，即**Target**；

◆α 为步长，当α是固定步长时，该式称为恒定-α MC。

5.4 蒙特卡洛控制（35）

✓异策略MC：

假设已经获得了状态s的回报序列$G_1, G_2, \cdots, G_{n-1}$，每个回报都对应一个随机重要性权重$W_i$ $\left(W_i = \rho_{t:T(t)-1}\right)$。当获得新的回报值$G_n$时，希望以增量式的方式，在状态值函数估计值$V_n$的基础上估计$V_{n+1}$。

在普通重要性采样中，仅仅需要对回报赋予权重W_i，其增量式与经典增量式方程基本一致：

$$V(s) \leftarrow V(s) + \alpha\left(WG - V(s)\right)$$

5.4 蒙特卡洛控制（36）

在加权重要性采样中，需要为每一个状态计算前 n 个回报的累积权重 C_n：

$$C_n=\sum_{k=1}^{n}W_k \quad\Rightarrow\quad C_n=C_{n-1}+W_n \qquad\qquad (C_0=0)$$

推导过程为：

$$\begin{aligned}
V_{n+1}&=\sum_{k=1}^{n}W_kG_k/C_n=\left(\sum_{k=1}^{n-1}W_kG_k+W_nG_n\right)/C_n\\
&=\frac{\sum_{k=1}^{n-1}W_kG_k}{C_n}+\frac{W_nG_n}{C_n}=\frac{\sum_{k=1}^{n-1}W_kG_k}{\sum_{k=1}^{n-1}W_k}*\frac{\sum_{k=1}^{n-1}W_k}{C_n}+\frac{W_nG_n}{C_n}\\
&=V_n\frac{C_n-W_n}{C_n}+\frac{W_nG_n}{C_n}\\
&=V_n+\frac{W_n}{C_n}\left(G_n-V_n\right) \qquad\qquad (n\geq 1)
\end{aligned}$$

5.4 蒙特卡洛控制（37）

5.4.5 异策略蒙特卡洛控制

异策略MC控制算法与异策略MC预测算法的原理一致，动作值函数更新递归式为：

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{W}{C(S_t, A_t)}[G - Q(S_t, A_t)]$$

算法5.4给出了用于估算最优策略的，异策略每次访问MC控制算法。

5.4 蒙特卡洛控制（38）

算法 5.4 异策略每次访问 MC 控制算法

初始化：

1. 对所有 $s \in \mathcal{S}^{+}$， $a \in \mathcal{A}(s)$ ， 初始化 $Q(s,a) \in \mathbb{R}$， $Q(s^{\mathrm{T}},a)=0$， $C(s,a)=0$
2. $\varepsilon \leftarrow (0,1)$ 为一个逐步递减的较小的实数
3. $\pi(s) \leftarrow \arg\max_{a} Q(s,a)$

---

4. **repeat** 对每一个情节 $k=0,1,2,\cdots$
5. $b \leftarrow$ 任意软性策略
6. 根据策略 $b(s)$，从初始状态-动作对 (S_0,A_0) 开始，生成一个情节序列 $S_0,A_0,R_1,\cdots,S_{\mathrm{T}-1},A_{\mathrm{T}-1},R_{\mathrm{T}},S_{\mathrm{T}}$
7. $G \leftarrow 0$
8. $W \leftarrow 1$

5.4 蒙特卡洛控制（38）

9. **for** 本情节中的每一步 $t = \mathrm{T}-1$ **downto** 0 **do**

10. $G \leftarrow \gamma G + R_{t+1}$

11. $C(S_t, A_t) \leftarrow C(S_t, A_t) + W$

12. $Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \dfrac{W}{C(S_t, A_t)}\left[G - Q(S_t, A_t)\right]$

13. $\pi(S_t) \leftarrow \arg\max\limits_a Q(S_t, a)$

14. **if** $A_t = \pi(S_t)$ **then** $W \leftarrow W \dfrac{1}{b(A_t \mid S_t)}$

else break

15. **end for**

输　出：$\pi_* = \pi$

5.4 蒙特卡洛控制（39）

例5.4 对于扫地机器人问题，通过行为策略来生成情节，然后利用每次访问和重要性采样比率计算动作值函数 $Q(S_t, A_t)$，如果行为策略采样的动作不是目标策略采取的动作，则会结束该循环开始新一轮循环。这样也就产生很多无用的数据，使得学习效率不高。

5.4 蒙特卡洛控制（40）

异策略每次访问MC控制算法的 Q 值更新表

	S_5	S_{10}	S_{18}	S_{20}	S_{24}
Q_0	0.00;0.00;*.**;0.00	0.00;0.00;*.**;0.00	0.00;0.00;0.00;0.00	*.**;0.00;*.**;0.00	*.**;0.00;0.00;*.**
π_0	0.17;0.17;0.00;0.67	0.17;0.17;0.00;0.67	0.12;0.12;0.62;0.12	0.00;0.25;0.00;0.75	0.00;0.25;0.75;0.00
Q_1	0.00;0.00;*.**;0.00	0.00;0.00;*.**;0.00	1.92;0.00;0.00;0.00	*.**;0.00;*.**;0.00	*.**;3.00;0.00;*.**
π_1	0.17;0.17;0.00;0.67	0.17;0.17;0.00;0.67	0.63;0.12;0.12;0.12	0.00;0.25;0.00;0.75	0.00;0.75;0.25;0.00
：	：	：	：	：	
Q_{7500}	0.96;1.00;*.**;0.98	1.22;0.80;*.**;1.21	1.92;1.89;1.89;3.00	*.**;1.18;*.**;1.23	*.**;3.00;1.92;*.**
π_{7500}	0.10;0.79;0.00;0.10	0.79;0.10;0.00;0.10	0.08;0.08;0.08;0.77	0.00;0.16;0.00;0.84	0.00;0.84;0.16;0.00
：	：	：	：	：	

5.4 蒙特卡洛控制（41）

Q_{12500}	0.98;1.00;*.**;0.98	1.22;0.80;*.**;1.22	1.92;1.90;1.90;3.00	*.**;1.2;*.**;1.23	*.**;3.00;1.91;*.**
π_{12500}	0.06;0.88;0.00;0.06	0.12;0.44;0.00;0.44	0.05;0.05;0.05;0.86	0.00;0.09;0.00;0.91	0.00;0.91;0.09;0.00
:	:	:	:	:	
Q_{19999}	0.98;1.00;*.**;0.98	1.22;0.80;*.**;1.22	1.92;1.91;1.91;3.00	*.**;1.21;*.**;1.23	*.**;3.00;1.92;*.**
π_{19999}	0.00;1.00;0.00;0.00	0.50;0.00;0.00;0.50	0.00;0.00;0.00;1.00	0.00;0.00;0.00;1.00	0.00;1.00;0.00;0.00
Q_{20000}	0.50;1.00;*.**;0.45	0.62;0.72;*.**;0.58	1.78;1.61;1.69;3.00	*.**;1.02;*.**;1.21	*.**;3.00;1.92;*.**
π_{20000}	0.00;1.00;0.00;0.00	0.50;0.00;0.00;0.50	0.00;0.00;0.00;1.00	0.00;0.00;0.00;1.00	0.00;1.00;0.00;0.00
π_*	0.00;1.00;0.00;0.00	0.50;0.00;0.00;0.50	0.00;0.00;0.00;1.00	0.00;0.00;0.00;1.00	0.00;1.00;0.00;0.00

目 录

5.5 小结（1）

本章介绍了从经验中学习价值函数和最优策略的蒙特卡洛方法，这些“经验”主要体现在从多个情节采样数据。与DP方法相比，其优势主要在以下3个方面：

- MC方法不需要完整的环境动态模型，而可以直接通过与环境的交互来学习最优的决策行为。
- MC方法可以使用数据仿真或采样模型。在很多应用中，构建DP方法所需要的显式状态概率转移模型通常很困难，但是通过仿真采样得到多情节序列数据却很简单。

5.5 小结（2）

- MC方法可以简单、高效地聚焦于状态的一个小的子集，它可以只评估关注的区域而不评估其他的状态。

5.6 习题（1）

1、举例说明蒙特卡洛首次访问和每次访问的异同点。

2、蒙特卡洛方法可以解决哪些强化学习问题。

3、给出蒙特卡洛估计$q_\pi(s,a)$值的回溯图。

4、修改异策略蒙特卡洛控制算法，使之可以递增计算加权的平均值，请给出伪代码。

5、（编程）通过蒙特卡洛法计算：第3章习题2（图3.12）扫地机器人在等概率策略的情况下，分别给出实验次数为5000次和50000次时，每个状态的价值 。

The End