第 4 章

Markov 决策过程

本章的目标、内容

1. 了解 Markov 决策过程的定义及相关概念。
2. 掌握 Bellman 方程。

4.1 定义和记号

在序贯决策问题中，决策目标是通过选择合适的动作使得长期奖赏最大化。每个动作都会影响到长期结果。奖赏有长期奖赏和即时奖赏，在决策时，可能会为了最大化长期奖赏而牺牲即时奖赏。

在智能体与环境交互的每个时间步 t，智能体执行动作 A_t，从环境中得到观测 O_t 和即时奖赏 R_t，而环境接受到动作，并给出观测 O_{t+1} 和即时奖赏 R_{t+1}。

从决策开始时刻到时间步 t，形成一个观测、奖赏，动作组成的序列，称为**历史**，记为

$$H_t = O_1, R_1, A_1, O_2, R_2, \cdots, A_{t-1}, O_t, R_t$$

下一时刻将会发生什么依赖于历史，具体而言是智能体选择的动作和环境反馈的观测和奖赏。而**状态**指的是用于决定下一时刻将会发生什么的信息。它可以看成是历史的函数，记为 $S_t = f(H_t)$。

状态分为环境状态 S_t^e 和智能体状态 S_t^a，环境状态对于智能体而言可能只是部分可观的，而智能体状态是指智能体用于决策的信息。一般地，本书中的状态指的是智能体状态。

如果状态只依赖于现在而与过去无关，则称一个状态是 Markov 的，特别地，有如下定义。

定义 4.1 Markov 性

状态 S_t 是 Markov 的，如果

$$P[S_{t+1}|S_t] = P[S_{t+1}|S_1, S_2, \cdots, S_t]$$

其中 P 是条件概率。

由 $\{S_1, S_2, \cdots, S_t, \cdots\}$ 组成的序列称为 Markov 链。

定义 4.2 Markov 决策过程

如果序贯决策过程所有状态是 Markov 的，且环境状态是完全可观的，即 $O_t = S_t^e = S_t^a$，则称该过程是 Markov 决策过程。如果环境状态是部分可观的，则称其为部分可观 Markov 决策过程。

一个 Markov 决策过程，可以用四元组模型 $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$ 表示，分别为状态集、动作集、状态转移概率和奖赏。

不特别声明，奖赏、动作和状态都是随机变量。假设在时间步 t 时，状态为 s，动作为 a，下一个状态为 s'，则状态转移概率 $\mathcal{P}_{ss'}^a$ 和即时奖赏 $\mathcal{R}_s^a$ 分别为

$$\begin{aligned} \mathcal{P}_{ss'}^a &= P[S_{t+1} = s' | S_t = s, A_t = a] \\ \mathcal{R}_s^a &= E[R_{t+1} | S_t = s, A_t = a] \end{aligned} \tag{4.1}$$

4.2 有限 Markov 决策过程

在有限 MDP 中，状态、动作、奖赏都是有限个可能取值。随机变量 R_t 和 A_t 具有依赖于上一个状态和动作的离散概率分布。假设上一个状态和动作分别为 s, a，下一个状态为 s' 且奖赏为 r 的概率为：

$$p(s', r|s, a) \triangleq \Pr\{S_{t+1} = s', R_t = r | S_t = s, A_t = a\}$$

则状态转移概率为：

$$p(s'|s, a) = \sum_{r \in \mathcal{R}} p(s', r|s, a)$$

状态-动作的期望奖赏为：

$$r(s, a) = E[R_{t+1}|S_t = s, A_t = a] = \sum_{r \in \mathcal{R}} r \sum_{s' \in \mathcal{S}} p(s', r|s, a)$$

状态-动作-状态的期望奖赏为：

$$r(s,a,s')=E[R_{t+1}|S_{t+1}=s',S_t=s,A_t=a]=\sum_{r\in\mathcal{R}}rp(r|s,a,s')=\sum_{r\in\mathcal{R}}r\frac{p(s',r|s,a)}{p(s'|s,a)}$$

决策目标用期望回报或期望折扣回报表征。其中折扣回报定义如下：

$$G_t=R_{t+1}+\gamma R_{t+2}+\cdots+\gamma^{T-t-1}R_T$$

其中，T 是终止时间步，γ 是折扣因子。折扣因子与经济学的折现率类似，它影响每个时刻奖赏的作用，γ 越小意味着未来的奖赏对当前决策的作用小。回报可以看作 $\gamma=1$ 时的折扣回报。

决策任务可能是片段式的或连续式的：

（1）每个**片段**从一个状态出发，终止于结束状态，比如象棋中的结束状态可能是输、赢、和棋。T 是一个随机变量，因此每个片段即使出发状态相同，可能终止的时间步数不一样。

（2）**连续式任务**指的是智能体与环境交互过程不能分解成一系列片段的任务。T 可以是 $+\infty$。

下面给出策略、状态值函数和状态-动作对值函数的定义如下：

（1）策略 $\pi(a|s)$ 表示在状态 s 下，执行动作 a 的概率。

（2）状态值函数 $v_\pi(s)$ 表示从当前状态出发的期望回报，其数学表达式如下：

$$v_\pi(s)=E_\pi[G_t|S_t=s]$$

显然结束状态的值为 0。通常，状态值函数简称为值函数。

（3）状态-动作对值函数 $q_\pi(s,a)$ 表示从当前状态出发且动作为 a 时的期望回报。其数学表达式如下：

$$q_\pi(s,a)=E_\pi[G_t|S_t=s,A_t=a]$$

状态-动作对值函数也称为 Q-值函数。

4.3　Bellman 方程

由值函数定义，可得出如下单步递推方程：

$$\begin{aligned}v_\pi(s)&=E_\pi[R_{t+1}+\gamma G_{t+1}|S_t=s]\\&=\sum_a\pi(a|s)\sum_{s',r}p(s',r|s,a)[r+\gamma v_\pi(s')]\end{aligned}\tag{4.2}$$

这个方程也称为**值函数的 Bellman 方程**。

回传图（**backup diagram**）是有效的分析工具，它能帮助理清当前状态和子状态（或后继状态）之间的关系。在回传图中，空心圆表示状态，实心圆表示状态-动作对，空心圆和实心圆交替出现。图 4.1 给出了 v_π 的回传图。其中，第一层状态 s 有三个动作；第二层最右边的状态-动作对 (s,a) 有两个子状态；第三层最右边的状态为 s'，它是 (s,a) 的一个子状态。

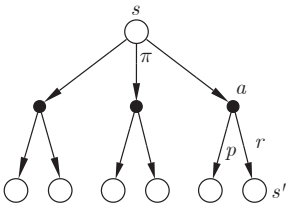

图 4.1　v_π 的回传图

Bellman 方程可以表示为矩阵形式。令

$$\boldsymbol{v}=\begin{bmatrix} v(1)\\ v(2)\\ \vdots\\ v(n)\end{bmatrix}$$

其中，$v(i)$ 表示第 i 个状态的值函数。

$$\boldsymbol{R}=\begin{bmatrix} R(1)\\ R(2)\\ \vdots\\ R(n)\end{bmatrix}$$

其中，$R(i)$ 表示状态 i 对应的即时奖赏；概率矩阵 $\boldsymbol{P}=[P_{ij}]$ 表示状态 i 转移到状态 j 的概率，则有：

$$\boldsymbol{v}=\boldsymbol{R}+\gamma\boldsymbol{P}\boldsymbol{v} \tag{4.3}$$

因而有 $\boldsymbol{v}=(I-\gamma\boldsymbol{P})^{-1}\boldsymbol{R}$。通过式 (4.3)，理论上可以计算出值函数。但是其计算复杂度为 $O(n^3)$，其中 n 为状态数。这种方法只能用于小规模问题。第 5 章将用动态规划方法求解规模较大的问题。

类似地，也有 Q-值函数的 Bellman 方程：

$$\begin{aligned} q_\pi(s,a)&=E_\pi[R_{t+1}+\gamma G_{t+1}|S_t=s,A_t=a]\\ &=\sum_{s',r}p(s',r|s,a)\big[r+\gamma\sum_{a'}\pi(a'|s')q_\pi(s',a')\big]\end{aligned} \tag{4.4}$$

图 4.2 给出了 q_π 的回传图。在图 4.2 中，第一层实心圆表示当前状态-动作对 (s,a)，它有两个子状态；第二层最右边的状态为 s' 有两个动作，其中一个动作是 a'；第三层最右边的状态-动作对为 (s',a')。

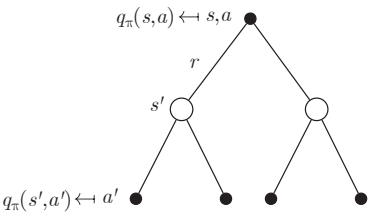

图 4.2　q_π 的回传图

4.4　最优策略

对于策略 π 与 π'，如果对于任意状态有 $v_\pi(s) \geqslant v_{\pi'}(s)$，称 π 是**优**于 π'，此时，记为 $\pi \geqslant \pi'$。至少存在一个策略比其他策略好，这个策略称为**最优策略**，记为 π^*。有

$$v^*(s) = \max_\pi v_\pi(s), \quad \forall s$$

$$q^*(s,a) = \max_\pi q_\pi(s,a), \quad \forall s$$

最优策略下的状态值函数与状态-动作对值函数的关系：

$$q^*(s,a) = E[r_{t+1} + \gamma v^*(s_{t+1}) | s_t = s, a_t = a]$$

最优策略下的值函数的 Bellman 方程如下：

$$\begin{aligned} v^*(s) &= \max_{a \in \mathcal{A}(s)} q^*(s,a) \\ &= \max_a E[r_{t+1} + \gamma v^*(s_{t+1}) | S_t = s, A_t = a] \\ &= \max_a \sum_{s',r} p(s',r|s,a)[r + \gamma v^*(s')] \end{aligned} \tag{4.5}$$

图 4.3(a) 给出了 v^* 的回传图。在图 4.3(a) 中，第一层空心圆表示当前状态 s，它有三个动作；第二层有三个状态-动作对，最右边的状态-动作对为 (s,a)，它有两个状态；第三层最右边的状态为 s'。注意在最优策略下第一层运算取得最大，这是按照 v^* 的定义来的。

最优策略下的 Q-值函数的 Bellman 方程为：

$$
\begin{aligned}
q^*(s,a) &= E[R_{t+1} + \gamma v^*(s_{t+1}) | S_t = s, A_t = a] \\
&= E[r_{t+1} + \gamma \max_{a'} q^*(s_{t+1}, a') | S_t = s, A_t = a] \\
&= \sum_{s',r} p(s', r|s, a)[r + \gamma \max_{a'} q^*(s', a')]
\end{aligned} \tag{4.6}
$$

图 4.3(b) 给出了 q^* 的回传图。在图 4.3(b) 中，第一层实心圆表示当前状态-动作对 (s, a)，它有两个子状态；第二层最右边的状态为 s' 有两个动作，其中一个动作是 a'；第三层最右边的状态-动作对为 (s', a')。注意在最优策略下，最后一层（即第三层）运算取得最大，这是按照 q^* 的定义来的。

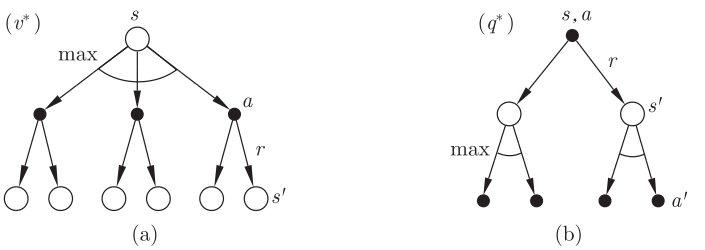

图 4.3　v^* 和 q^* 的回传图

例　图 4.4 给出了一个 MDP 问题：共有 5 个状态 $\{s_1, s_2, s_3, s_4, s_5\}$。在每个箭头旁边的英文单词表示动作，它的下面表示执行该动作得到的奖赏。比如在状态 s_1 时，如果执行动作 Facebook，则下一个状态是 s_1，能得到的奖赏是 -1。在状态 s_4 时，如果执行动作 Pub，则能得到奖赏为 1，它转入到状态 s_2、s_3 和 s_4 的概率分别为 0.2、0.4 和 0.4。

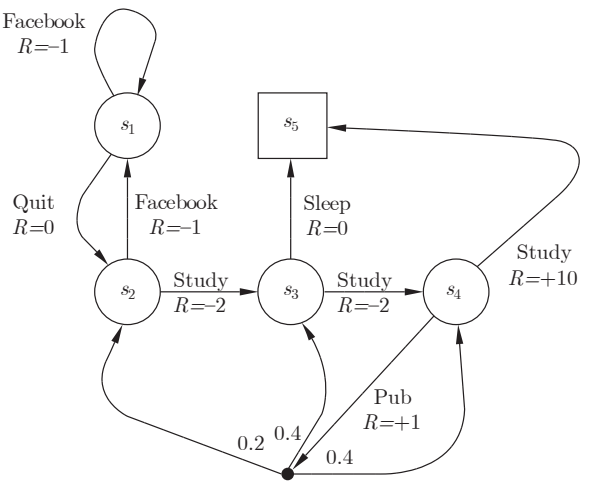

图 4.4　例子的状态转移图

1. 考虑策略 $\pi(a|s)=0.5$，$\gamma=1$ 时，计算出值函数

计算过程如下：为简单计，令 $v_i=v(s_i)$。由状态 s_1 的回传图可见：

$$v_1=0.5(-1+v_1)+0.5v_2$$

从而有 $v_1+1=v_2$;

由状态 s_2 的回传图有：

$$v_2=0.5(-1+v_1)+0.5(-2+v_3)$$

由状态 s_3 的回传图有：

$$v_3=0.5(-2+v_4)+0.5(0+v_5)$$

由状态 s_4 的回传图有：

$$v_4=0.5(1+0.2v_2+0.4v_3+0.4v_4)+0.5(10+v_5)$$

由状态 s_5 的回传图有：

$$v_5=0$$

联立方程组，可得到 $v(s_1)=-2.3$，$v(s_2)=-1.3$，$v(s_3)=2.7$，$v(s_4)=7.4$，$v(s_5)=0$。

2. 当 $\gamma=1$ 时，计算最优策略 π^*

计算过程如下，为简单计，令 $v_i=v^*(s_i)$。注意到：$v_5^*=0$。

对于状态 s_1，令 $a=\pi^*(\text{Facebook}|s_1)$，$b=\pi^*(\text{Quit}|s_1)$，有：

$$v_1=a(-1+v_1)+bv_2$$

可得 $v_1\leqslant v_2$，由于最优策略，在所有策略里面都取最大值，因此必有 $v_1=v_2$。对于状态 s_2，令 $a=\pi^*(\text{Facebook}|s_2)$，$b=\pi^*(\text{Study}|s_2)$，有

$$v_2=a(-1+v_1)+b(-2+v_3)$$

由于 $v_1=v_2$，故 $b>0$，有：

$$v_1=v_3-2-\frac{a}{b}$$

由于最优策略在所有策略里面都取最大值，因此必有 $v_1=v_3-2$。

对于状态 s_3，令 $a=\pi^*(\text{Sleep}|s_3)$，$b=\pi^*(\text{Study}|s_3)$：

$$v_3=b(-2+v_4)$$

注意，按照最优策略的定义 $v_4\geqslant 10$，因此有：

$$v_3=-2+v_4$$

对于状态 s_4，令 $a=\pi^*(\text{Pub}|s_4)$，$b=\pi^*(\text{Study}|s_4)$，有：

$$v_4=a(1+0.2v_2+0.4v_3+0.4v_4)+b(10+v_5)$$

当 $a=0,b=1$ 时，故 $v_4\geqslant 10$。由于 $v_1=v_3-2$，$v_1=v_2$，$v_3=-2+v_4$，因此有：

$$v_3=-\frac{0.6}{b}+8.6$$

当 $b=1$ 时，v_3 取最大值 8。

从而有 $v(s_1)=6$，$v(s_2)=6$，$v(s_3)=8$，$v(s_4)=10$，$v(s_5)=0$。

4.5 小结

本章介绍了 Markov 过程的基本概念、要素和模型。给出了值函数和 Q-值函数的定义，采用回传图给出了 Bellman 方程。最后，介绍了最优策略这一重要概念，并给出了 Bellman 最优方程。

4.6 习题

（1）试举一个生活中的 Markov 过程的实例，并给出其要素和模型。

（2）用 Q-值函数 $Q(s,a)$ 表示出值函数 $v(s)$，并用值函数表示出 Q-值函数。

（3）给出 Bellman 最优方程。

（4）编程实现：用动态规划求解 6 个城市的旅行商问题。

（5）编程实现：如图 4.5 所示的一个正方形格子世界表示的一个简单 Markov 决策过程，每个小格子对应于环境中的一个状态。智能体可以从每个小格子移出，有 4 个可能的动作（上、下、左、右）来移动，奖赏为 0；如果跳出格子外，必须回到原位，奖赏为 −1；也可以在每个小格子，保持不动奖赏为 −1；此外，在小格子 A，智能体无论采取哪种动作都只能跳到 A′，奖赏为 10；

在小格子 B，智能体无论采取哪种动作都只能跳到 B′，奖赏为 5。①假设在每个小格子，智能体以等概率选择一个方向进行移动，试求每个状态（或小格子）对应的值函数；②求最优策略及最优策略下的值函数。

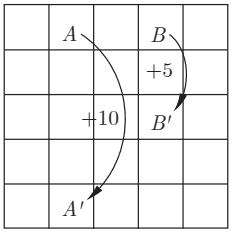

图 4.5　格子世界

参考文献

[1] Richard S，Andrew G. Reinforcement Learning: An Introduction[M]. Cambridge, Massachusetts: MIT Press, 2017.

[2] David Silver. UCL Course on RL[OL]. http://www0.cs.ucl.ac.uk/staff/D.Silver/web/Teaching.html.