强化学习笔记：价值函数与贝尔曼方程

2025 年 7 月 20 日

1 State Value (状态价值函数)

- **状态价值（State Value）**：在遵循策略 π 的前提下，从状态 s 出发所能获得的期望回报（Expected Return）。
- **回报（Return）**：从某个时刻开始，后续所有奖励的折扣总和，是一条具体轨迹的产出。
- **价值（Value）**：对所有可能轨迹的回报求期望，是一个统计量。

状态价值函数 (State-Value Function) 定义为：

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

其中：

- $G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$ 表示从时刻 t 开始的总折扣回报；
- $\mathbb{E}_\pi[\cdot]$ 表示在策略 π 下的期望。

---

2 Bellman Equation (贝尔曼方程)

状态价值函数的贝尔曼展开：

$$v_\pi(s) = \mathbb{E}_\pi\big[R_{t+1} + \gamma\, G_{t+1} \mid S_t = s\big]$$

进一步边缘化动作和后继状态，可得：

$$v_\pi(s) = \sum_a \pi(a \mid s) \Big(\mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a\, v_\pi(s')\Big)$$

或者写成期望形式：

$$v_\pi(s) = \sum_a \pi(a \mid s)\, \mathbb{E}\big[R_{t+1} + \gamma\, v_\pi(S_{t+1}) \mid S_t = s, A_t = a\big]$$

其中：

- $\mathcal{R}_s^a = \mathbb{E}[R_{t+1} \mid S_t = s, A_t = a]$；
- $\mathcal{P}_{ss'}^a = p(s' \mid s, a)$。

---

3 矩阵形式的贝尔曼方程

首先，为所有状态 $s \in \mathcal{S}$ 写出：

$$v_\pi(s) = \sum_a \pi(a \mid s)\,\mathcal{R}_s^a \;+\; \gamma \sum_a \pi(a \mid s) \sum_{s'} \mathcal{P}_{ss'}^a\, v_\pi(s')$$

定义：

- 价值向量 $\mathbf{v}_\pi$，其中第 s 项为 $v_\pi(s)$；
- 奖励向量 $\mathbf{r}_\pi$，其中第 s 项为 $r_\pi(s) = \sum_a \pi(a \mid s)\,\mathcal{R}_s^a$；
- 状态转移矩阵 $\mathbf{P}_\pi$，其中第 (s, s') 项为 $P_\pi(s, s') = \sum_a \pi(a \mid s)\,\mathcal{P}_{ss'}^a$。

于是矩阵形式：

$$\mathbf{v}_\pi = \mathbf{r}_\pi + \gamma\,\mathbf{P}_\pi\,\mathbf{v}_\pi$$

解得：

$$\mathbf{v}_\pi = (I - \gamma\,\mathbf{P}_\pi)^{-1}\,\mathbf{r}_\pi$$

---

4 价值函数的迭代求解方法

4.1 策略评估 (Policy Evaluation)

从初始值 $\mathbf{v}^{(0)}$ 开始迭代：

$$\mathbf{v}^{(k+1)} = \mathbf{r}_\pi + \gamma\,\mathbf{P}_\pi\,\mathbf{v}^{(k)}$$

直到满足：

$$\|\mathbf{v}^{(k+1)} - \mathbf{v}^{(k)}\|_\infty < \epsilon$$

4.2 策略迭代 (Policy Iteration)

1. 初始化策略 π_0；
2. 使用策略评估计算 v_{π_k}；
3. 策略改进：
$$\pi_{k+1}(s) = \arg\max_a \Big(\mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a\, v_{\pi_k}(s')\Big)$$
4. 若 $\pi_{k+1} = \pi_k$ 则停止，否则返回第 2 步。

4.3 值迭代 (Value Iteration)

从任意初始值 $v^{(0)}$ 开始迭代：

$$v^{(k+1)}(s) = \max_a \Big(\mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a\, v^{(k)}(s')\Big)$$

当收敛到 v^* 后，可通过

$$\pi^*(s) = \arg\max_a \Big(\mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a\, v^*(s')\Big)$$

提取得到最优策略。

---

5 Action Value (动作价值函数)

动作价值函数 (Action-Value Function) 定义为:

$$
\begin{aligned}
q_\pi(s,a) &= \mathbb{E}_\pi\big[G_t \mid S_t = s, A_t = a\big] \\
&= \mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a \, v_\pi(s')
\end{aligned}
$$

状态价值与动作价值的关系:

$$
v_\pi(s) = \sum_a \pi(a \mid s)\, q_\pi(s,a)
$$

---

6 贝尔曼最优方程 (Bellman Optimality Equation)

最优状态价值函数:

$$
v^*(s) = \max_a q^*(s,a)
$$

最优动作价值函数:

$$
q^*(s,a) = \mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a \, v^*(s')
$$

综合得:

$$
v^*(s) = \max_a \Big(\mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a \, v^*(s')\Big)
$$

$$
q^*(s,a) = \mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a \, \max_{a'} q^*(s',a')
$$

---

7 不动点定理与收缩映射

收缩映射 (Contraction Mapping) 定义:

在完备度量空间 (X,d) 中,若映射 $f : X \to X$ 满足

$$
d\big(f(x), f(y)\big) \le k\, d(x,y), \quad k \in [0,1),
$$

则称 f 为收缩映射。

巴拿赫不动点定理 (Banach Fixed-Point Theorem):

任何收缩映射 f 都存在唯一不动点 x^*,并且迭代

$$
x_{n+1} = f(x_n)
$$

必收敛于 x^*。

贝尔曼最优算子:

定义算子 $\mathcal{T}$ 为

$$
(\mathcal{T}v)(s) = \max_a \Big(\mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a \, v(s')\Big).
$$

该算子是 γ-收缩,因此存在唯一不动点 v^*,即最优价值函数,且值迭代

$$
v^{(k+1)} = \mathcal{T} v^{(k)}
$$

必收敛于 v^*。