Reinforcement Learning & Optimal Control

强化学习与最优控制

第2章 MDP与动态规划

目录

随机过程

- 随机过程是一个或多个事件、随机系统或者随机现象随时间发生演变的过程

$$\mathbb{P}[S_{t+1}|S_1, \dots, S_t]$$

- 概率论研究静态随机现象的统计规律
- 随机过程研究动态随机现象的统计规律

布朗运动

天气变化

随机过程

足球比赛

城市交通

生态系统

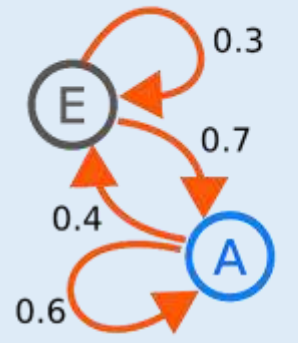
星系

马尔可夫过程

- 马尔可夫过程（Markov Process）是具有马尔可夫性质的随机过程

“The future is independent of the past given the present”

- 定义
  - 状态满足马尔可夫性质，当且仅当 $\mathbb{P}[S_{t+1}|S_t] = \mathbb{P}[S_{t+1}|S_1, \dots, S_t]$
- 性质：
  - 状态从历史（history）中捕获了所有相关信息
  - 当状态已知的时候，可以抛开历史不管
  - 也就是说，当前状态是未来的充分统计量

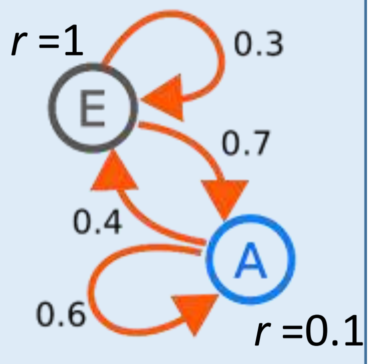


一个具有两个转换状态的马尔可夫链

马尔可夫奖励过程

- 在马尔可夫过程基础上加入奖励函数和折扣因子，就得到了马尔可夫奖励过程过程（Markov reward process）

“MRP=MP + Reward + discount”



一个具有两个转换状态的马尔可夫链

- 主要要素：
  - 状态转移是马尔可夫的 $\mathbb{P}[S_{t+1}|S_t] = \mathbb{P}[S_{t+1}|S_1, \dots, S_t]$
  - 基于每个时间步的状态 s，环境产生相应的奖励$r(s)$，随机变量记为R_{t}
  - 基于状态序列及其对应的奖励，可以得到序列的回报:

$$G_t = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k}$$

马尔可夫奖励过程-序列回报形式

- 为什么？序列回报形式

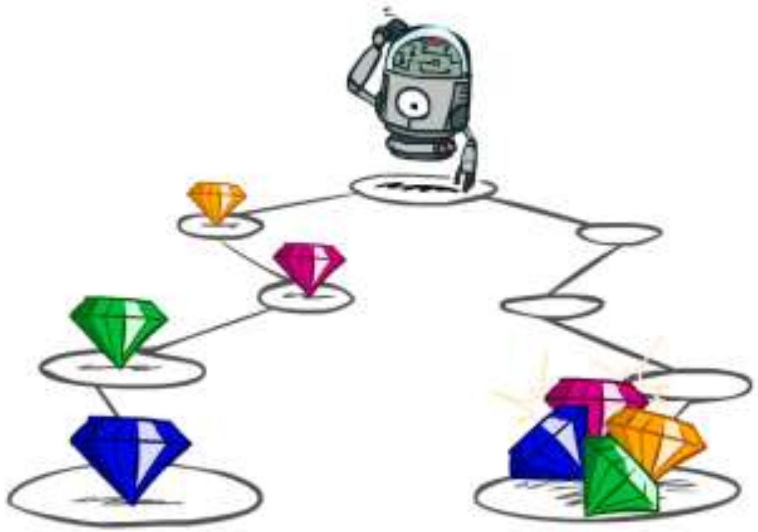
$$G_t = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k}$$

- 需要构建序列之间的全序，也即是对任意两个序列，需要有孰好孰坏之分
  - 比大小，多还是少？奖励加和

$$[1,2,3] < [2,3,4]$$

  - 近期or远期？做时间衰减

$$[1,0,0] > [0,0,1]$$

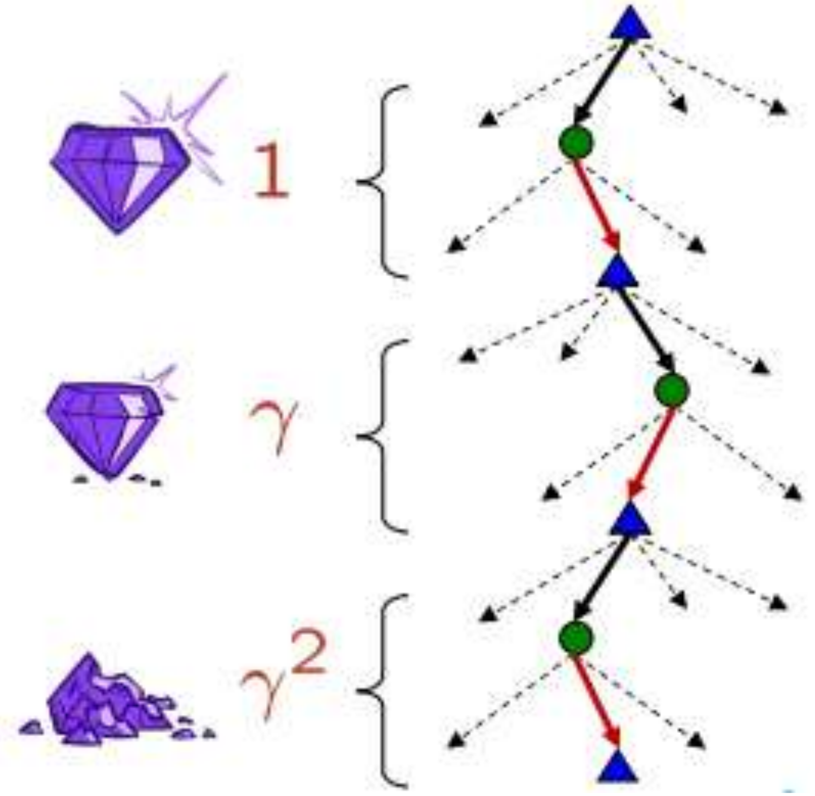


马尔可夫奖励过程-序列回报形式

- 为什么？序列回报形式

$$G_t = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k}$$

- 如何做衰减？
  - 在每个时间步，奖励乘上一个衰减因子 $\gamma \in [0,1]$
  - 可以考虑为每一个时间步都有1 - γ的概率会直接结束该序列，因此未来的奖励需要打折(求期望)
  - 该衰减因子也帮助算法收敛



马尔可夫决策过程

- 马尔可夫决策过程（Markov Decision Process，MDP）
  - 提供了一套为在结果部分随机、部分在决策者的控制下的决策过程建模的数学框架

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

$$\mathbb{P}[S_{t+1}|S_t, A_t]$$

- MDP形式化地描述了一种强化学习的环境
  - 环境完全可观测
  - 当前状态可以完全表征过程（马尔可夫性质）

MDP五元组

- MDP可以由一个五元组表示 $(S, A, \{P_{sa}\}, \gamma, R)$
  - S是状态的集合
    - 比如，迷宫中的位置，Atari游戏中的当前屏幕显示
  - A是动作的集合
    - 比如，向N、E、S、W移动，手柄操纵杆方向和按钮
  - P_{sa}是状态转移概率
    - 对每个状态$s \in S$和动作$a \in A$，P_{sa}是下一个状态在S中的概率分布
  - $\gamma \in [0,1]$是对未来奖励的折扣因子
  - $R: S \times A \mapsto \mathbb{R}$ 是奖励函数
    - 有时奖励只和状态相关

MDP的动态

- MDP的动态如下所示：
  - 从状态s_0开始
  - 智能体选择某个动作$a_0 \in A$
  - 智能体得到奖励$R(s_0, a_0)$
  - MDP随机转移到下一个状态$s_1 \sim P_{s_0 a_0}$
  - 这个过程不断进行

$$s_0 \xrightarrow{a_0, R(s_0, a_0)} s_1 \xrightarrow{a_1, R(s_1, a_1)} s_2 \xrightarrow{a_2, R(s_2, a_2)} s_3 \quad \cdots$$

  - 直到终止状态s_T出现为止，或者无止尽地进行下去
  - 智能体的总回报为

$$R(s_0, a_0) + \gamma R(s_1, a_1) + \gamma^2 R(s_2, a_2) + \cdots$$

MDP的动态性

- 在大部分情况下，奖励只和状态相关
  - 比如，在迷宫游戏中，奖励只和位置相关
  - 在围棋中，奖励只基于最终所围地盘的大小有关
- 这时，奖励函数为 $R(s): S \longmapsto \mathbb{R}$
- MDP的过程为

$$s_0 \xrightarrow{a_0, R(s_0)} s_1 \xrightarrow{a_1, R(s_1)} s_2 \xrightarrow{a_2, R(s_2)} s_3 \quad \cdots$$

- 累积奖励为

$$R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \cdots$$

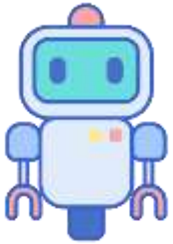
回顾：与动态环境的交互中学习

- 有监督、无监督学习

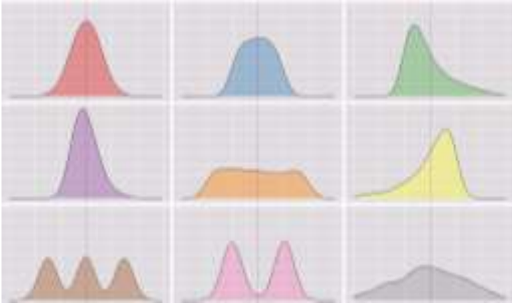
泛化模型

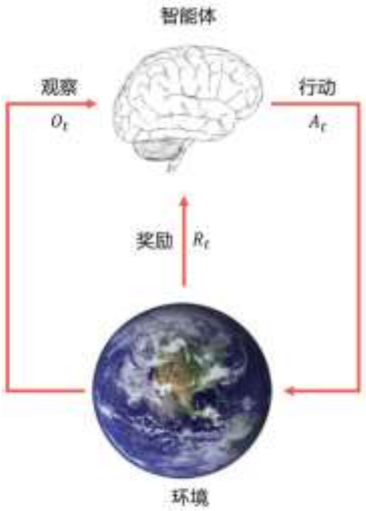
固定数据分布（IID假设）

- 强化学习



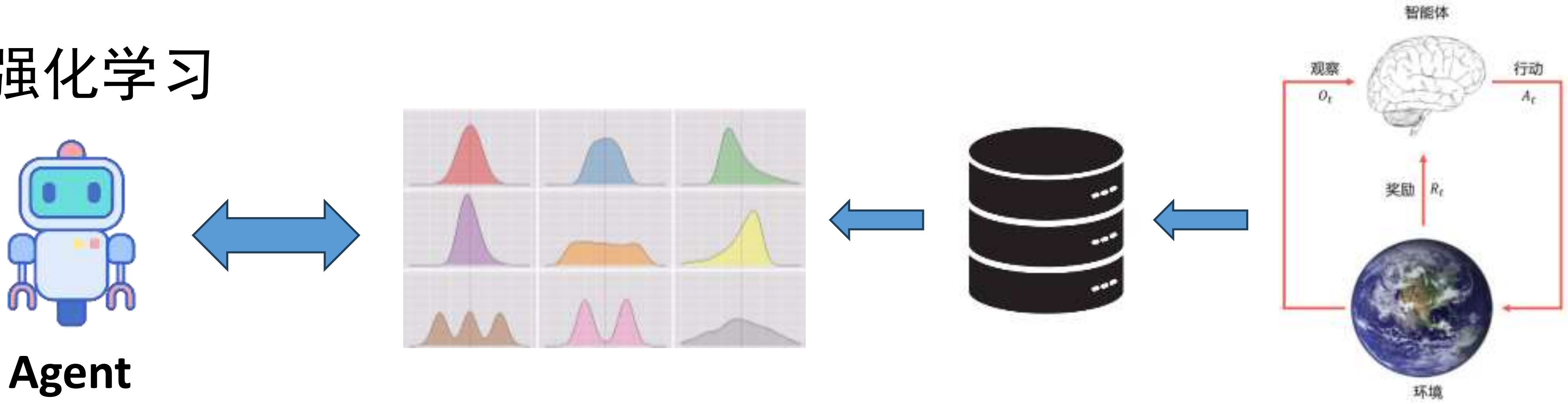
Agent





与动态环境交互产生的数据分布

■ 强化学习



- 给定同一个动态环境（即MDP），不同的策略采样出来的“状态-行动”对的分布是不同的

- 占用度量（Occupancy Measure）

$$\rho^{\pi}(s,a)=\mathbb{E}\left[\sum_{t=0}^{T}\gamma^{t}\mathbb{I}(S_t=s,A_t=a)|\pi\right],\quad \forall s\in\mathcal{S},a\in\mathcal{A}$$

$$=\sum_{t=0}^{T}\gamma^{t}P(S_t=s,A_t=a|s_0,\pi)$$

占用度量和策略

占用度量（Occupancy Measure）

$$\rho^{\pi}(s,a) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^{t} \mathbb{I}(S_t = s, A_t = a) | \pi\right], \quad \forall s \in \mathcal{S}, a \in \mathcal{A}$$

- 定理1：和同一个动态环境交互的两个策略π_1和π_2得到的占用度量ρ^{π_1}和ρ^{π_2}满足

$$\rho^{\pi_1} = \rho^{\pi_2} \text{ iff } \pi_1 = \pi_2$$

- 定理2：给定一占用度量ρ，可生成该占用度量的唯一策略是

$$\pi_\rho = \frac{\rho(s,a)}{\sum_{a'} \rho(s,a')}$$

U. Syed, M. Bowling, and R. E. Schapire. Apprenticeship learning using linear programming. ICML 2008.

占用度量和策略

■ 占用度量（Occupancy Measure）

$$\rho^{\pi}(s,a)=\sum_{t=0}^{T}\gamma^{t}P(S_t=s,A_t=a|s_0,\pi)$$

■ 状态占用度量

$$\begin{aligned}
\rho^{\pi}(s) &= \sum_{t=0}^{T}\gamma^{t}P(S_t=s|s_0,\pi) \\
&= \sum_{t=0}^{T}\gamma^{t}P(S_t=s|s_0,\pi)\sum_{a}\pi(A_t=a|S_t=s) \\
&= \sum_{a}\sum_{t=0}^{T}\gamma^{t}P(S_t=s,A_t=a|s_0,\pi) \\
&= \sum_{a}\rho^{\pi}(s,a)
\end{aligned}$$

U. Syed, M. Bowling, and R. E. Schapire. Apprenticeship learning using linear programming. ICML 2008.

占用度量和累计奖励

占用度量（Occupancy Measure）

$$\rho^{\pi}(s,a)=\sum_{t=0}^{T}\gamma^{t}P(S_t=s,A_t=a|s_0,\pi)$$

策略累计奖励

$$\begin{aligned}\eta(\pi)&=\mathbb{E}[r(S_0,A_0)+\gamma r(S_1,A_1)+\gamma^2 r(S_2,A_2)+\cdots|s_0,\pi]\\&=\sum_{s,a}\sum_{t=0}^{T}\gamma^{t}P(S_t=s,A_t=a|s_0,\pi)r(s,a)\\&=\sum_{s,a}\rho^{\pi}(s,a)r(s,a)=\mathbb{E}_{\pi}[r(s,a)]\end{aligned}$$

换个角度：时域→空间域

U. Syed, M. Bowling, and R. E. Schapire. Apprenticeship learning using linear programming. ICML 2008.

MDP中策略的目标

- 策略学习的目标：选择能够最大化累积奖励期望的动作

$$\max_{\pi} \mathbb{E}\left[r(S_0,A_0)+\gamma r(S_1,A_1)+\gamma^2 r(S_2,A_2)+\cdots \mid s_0,\pi\right]$$
$$=\sum_{s,a}\rho^{\pi}(s,a)r(s,a)=\mathbb{E}_{\pi}\left[r(s,a)\right]$$

- 如何实现上述目标？
  - 策略π和其占用度量ρ^{π}的对应关系是黑盒的，因此以上优化目标并没有直接对π更新方向的指导
  - 在每一个状态s下，策略改变了动作的选择后，策略整体是否变得更优秀了？

目录

策略值函数估计（Policy Evaluation）

- 给定环境MDP和策略π，策略值函数估计如下

状态价值

$$V^{\pi}(s)=\mathbb{E}[r(S_0,A_0)+\gamma r(S_1,A_1)+\gamma^2 r(S_2,A_2)+\cdots|S_0=s,\pi]$$

$$=\mathbb{E}_{a\sim\pi(s)}\left[r(s,a)+\gamma\sum_{s'\in\mathcal{S}}P_{s,a}(s')V^{\pi}(s')\right]$$

$$=\mathbb{E}_{a\sim\pi(s)}[Q^{\pi}(s,a)]$$

动作价值

$$Q^{\pi}(s,a)=\mathbb{E}[r(S_0,A_0)+\gamma r(S_1,A_1)+\gamma^2 r(S_2,A_2)+\cdots|S_0=s,A_0=a,\pi]$$

$$=r(s,a)+\gamma\sum_{s'\sim\mathcal{S}}P_{s,a}(s')V^{\pi}(s')$$

策略提升（Policy Improvement）

■ 对于两个策略π，π'，如果满足如下性质， π'是π的策略提升：

$$Q^{\pi}(s,\pi'(s)) \geq V^{\pi}(s), \quad \forall s \in \mathcal{S}$$

■ 一种特例：给定环境MDP和两个策略π，π'，如满足：

1. 在某个状态s下，两策略的输出不同，且有

$$\pi'(s) \neq \pi(s) \qquad\qquad Q^{\pi}(s,\pi'(s)) > Q^{\pi}(s,\pi(s)) = V^{\pi}(s)$$

2. 在其他所有状态s’下，两策略输出相同，即

$$\pi'(s') = \pi(s') \qquad\qquad Q^{\pi}(s',\pi'(s')) > Q^{\pi}(s',\pi(s')) = V^{\pi}(s')$$

那么π'是π的一种策略提升

策略提升定理（Policy Improvement Theorem）

- 对于两个策略π，π'，如果满足如下性质， π'是π的策略提升：

$$Q^{\pi}(s,\pi'(s)) \geq V^{\pi}(s), \quad \forall s \in \mathcal{S}$$

- 进而， π和π'满足：对任何状态s，有

$$V^{\pi'}(s) \geq V^{\pi}(s), \quad \forall s \in \mathcal{S}$$

也即是π'的策略价值（期望回报）超过π，π'比π更加优秀

策略提升定理（Policy Improvement Theorem）

- 对于两个策略π，π'，如果满足如下性质， π'是π的策略提升：

$$Q^{\pi}(s,\pi'(s)) \geq V^{\pi}(s), \quad \forall s \in \mathcal{S} \quad \Rightarrow \quad V^{\pi'}(s) \geq V^{\pi}(s), \quad \forall s \in \mathcal{S}$$

- 证明：

$$\begin{aligned}
V^{\pi}(s) &\leq Q^{\pi}(s,\pi') \\
&= \mathbb{E}_{a\sim\pi'(a|s),s'\sim\mathcal{T}'(s'|s,a)}[r(s,a)+\gamma V^{\pi}(s')] \\
&\leq \mathbb{E}_{a\sim\pi'(a|s),s'\sim\mathcal{T}'(s'|s,a)}[r(s,a)+\gamma Q^{\pi}(s',\pi')] \\
&= \mathbb{E}_{a,a'\sim\pi'}[r(s,a)+\gamma r(s',a')+\gamma^2 V^{\pi}(s'')] \\
&\leq \ldots \\
&\leq \mathbb{E}_{a,a',a''\ldots\sim\pi'}[r(s,a)+\gamma r(s',a')+\gamma^2 r(s'',a'')+\ldots] \\
&= V^{\pi'}(s)
\end{aligned}$$

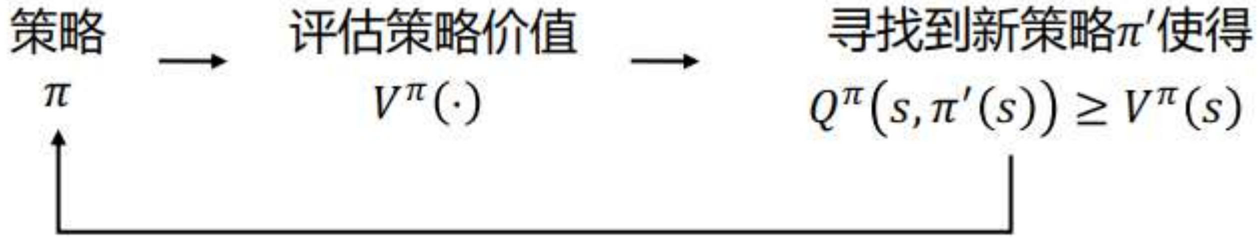
理论证明：https://yuanz.web.illinois.edu/teaching/IE498fa19/lec_16.pdf

策略提升定理（Policy Improvement Theorem）

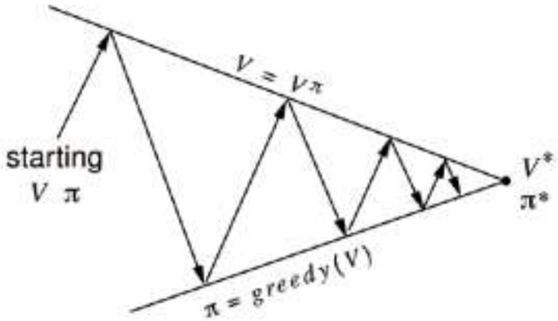
- 对于两个策略π，π'，如果满足如下性质，π'是π的策略提升：

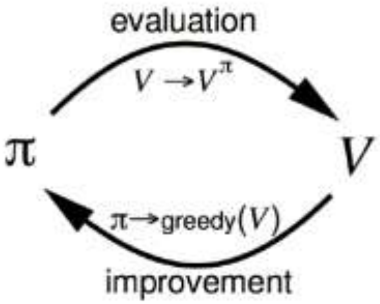
$$Q^{\pi}(s,\pi'(s)) \geq V^{\pi}(s), \quad \forall s \in \mathcal{S} \quad \Rightarrow \quad V^{\pi'}(s) \geq V^{\pi}(s), \quad \forall s \in \mathcal{S}$$

- 启示：



找到(s,a)使得 $Q^{\pi}(s,a) \geq V^{\pi}(s)$





策略评估指导策略提升

ε-Greedy策略提升定理

- 对于m个动作的ε-Greedy策略定义：

$$\pi(a|s)=\begin{cases}\epsilon/m+1-\epsilon & \text{if } a^*=\operatorname{argmax}_{a\in\mathcal{A}}Q(s,a)\\ \epsilon/m & \text{otherwise}\end{cases}$$

- 如果另一个ε-Greedy策略π'是基于Q^{π}的提升，那么有：　$V^{\pi'}(s)\geq V^{\pi}(s)$

$$\begin{aligned}\underbrace{V^{\pi'}(s)\geq Q^{\pi}(s,\pi'(s))}_{\text{如前面多步推导得出}} &= \sum_{a\in A}\pi'(a|s)Q^{\pi}(s,a)\\ \text{m actions}\quad &= \frac{\epsilon}{m}\sum_{a\in A}Q^{\pi}(s,a)+(1-\epsilon)\max_{a\in A}Q^{\pi}(s,a)\\ &\geq \frac{\epsilon}{m}\sum_{a\in A}Q^{\pi}(s,a)+(1-\epsilon)\sum_{a\in A}\frac{\pi(a|s)-\epsilon/m}{1-\epsilon}Q^{\pi}(s,a)\\ &= \sum_{a\in A}\pi(a|s)Q^{\pi}(s,a)=V^{\pi}(s)\end{aligned}$$

注意：
1. ε不能大
2. π并不是ε-Greedy(Q^{π})，π'才是

思考：对于随机策略，将策略的动作选择分布移向价值更高的动作

目录

策略值函数估计（Policy Evaluation）

- 给定环境MDP和策略π，策略值函数估计如下

状态价值 $V^{\pi}(s) = \mathbb{E}[r(S_0, A_0) + \gamma r(S_1, A_1) + \gamma^2 r(S_2, A_2) + \cdots | S_0 = s, \pi]$

$$= \mathbb{E}_{a \sim \pi(s)}\left[r(s,a) + \gamma \sum_{s' \in \mathcal{S}} P_{s,a}(s') V^{\pi}(s')\right]$$ **Bellman等式**

$$= \mathbb{E}_{a \sim \pi(s)}[Q^{\pi}(s,a)]$$

动作价值 $Q^{\pi}(s,a) = \mathbb{E}[r(S_0, A_0) + \gamma r(S_1, A_1) + \gamma^2 r(S_2, A_2) + \cdots | S_0 = s, A_0 = a, \pi]$

$$= r(s,a) + \gamma \sum_{s' \sim \mathcal{S}} P_{s,a}(s') V^{\pi}(s')$$

$$= r(s,a) + \gamma \sum_{s' \in \mathcal{S}} P_{s,a}(s') \sum_{a' \in \mathcal{A}} \pi(a'|s') Q^{\pi}(s',a')$$ **Bellman等式**

MDP中策略的目标

- 策略的目标：选择能够最大化累积奖励期望的动作

$$\max_{\pi} \mathbb{E}[r(S_0,A_0)+\gamma r(S_1,A_1)+\gamma^2 r(S_2,A_2)+\cdots|s_0,\pi]$$

  - $\gamma \in [0,1]$是未来奖励的折扣因子，使得和未来奖励相比起来智能体更重视，即时奖励（以金融为例，今天的$1比明天的$1更有价值）
  - $r(s)$ 和$r(s, a)$的设定是类似的，只需设$r(s) = r(s, a)$

- 给定一个确定性策略：$\pi(\cdot):\mathcal{S}\rightarrow\mathcal{A}$

- 给策略定义价值函数

$$V^{\pi}(s)=\mathbb{E}[r(S_0,A_0)+\gamma r(S_1,A_1)+\gamma^2 r(S_2,A_2)+\cdots|S_0=s,\pi]$$

$$Q^{\pi}(s)=\mathbb{E}[r(S_0,A_0)+\gamma r(S_1,A_1)+\gamma^2 r(S_2,A_2)+\cdots|S_0=s,A_0=a,\pi]$$

寻找优化策略的方法：策略迭代和价值迭代

- 价值函数和策略相关

$$V^{\pi}(s) = R(s) + \gamma \sum_{s' \in S} P_{s,\pi}(s')V^{\pi}(s')$$

$$\pi(s) = \arg\max_{a \in A} \sum_{s' \in S} P_{s,a}(s')V^{\pi}(s')$$

- 可以对最优价值函数和最优策略执行迭代更新

1. 策略迭代
2. 价值迭代

策略迭代

- 对于一个有限状态和动作空间的MDP

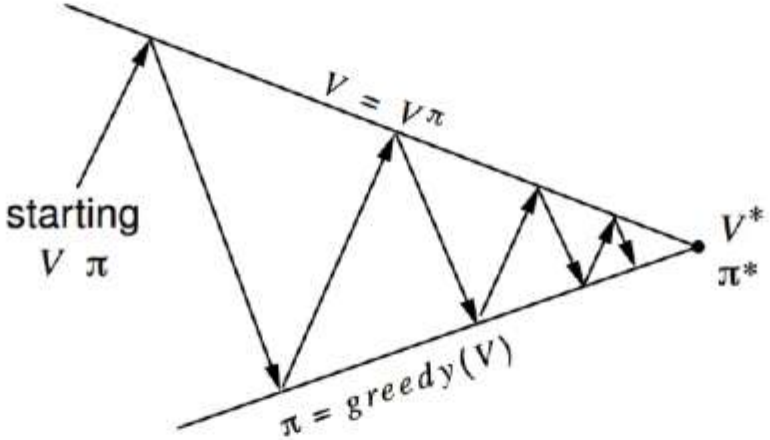
$$|S| < \infty, |A| < \infty$$

- 策略迭代过程（基于V价值函数）

1. 随机初始化策略π
2. 重复以下过程直到收敛 {
   a) 让 $V := V^\pi$
   b) 对每个状态，更新

$$\pi(s) = arg \max_{a \in A} r(s,a) + \gamma \sum_{s' \in S} P_{sa}(s')V(s')$$

}



更新价值函数会很耗时

策略迭代

- 对于一个有限状态和动作空间的MDP

$$|S| < \infty, |A| < \infty$$

- 策略迭代过程（基于Q价值函数）

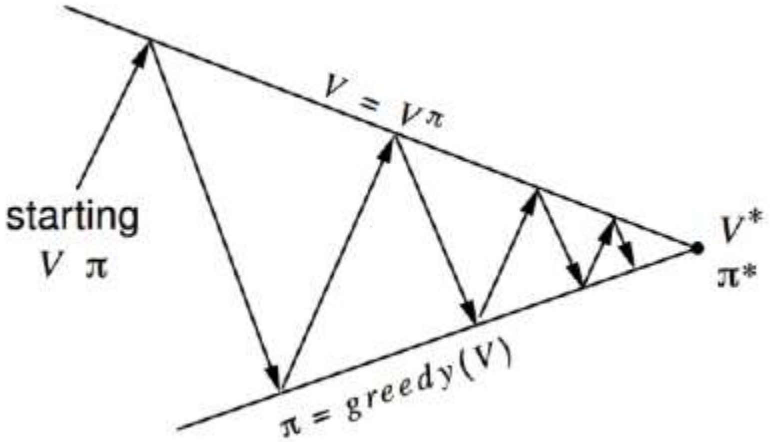
1. 随机初始化策略π
2. 重复以下过程直到收敛 {
   a) 让 $Q := Q^{\pi}$
   b) 对每个状态，更新

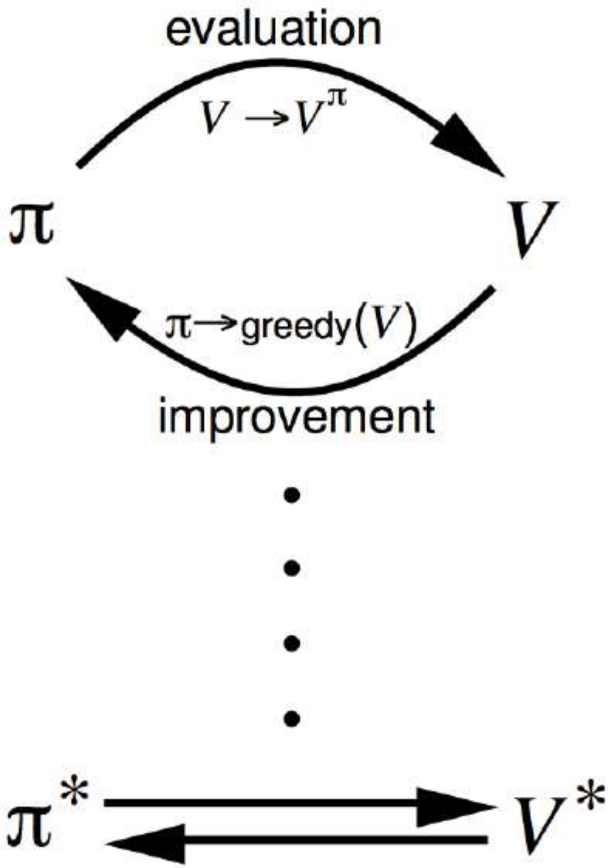
$$\pi(s) = arg\max_{a \in A} Q(s,a)$$

}

更新价值函数会很耗时

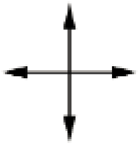
策略迭代



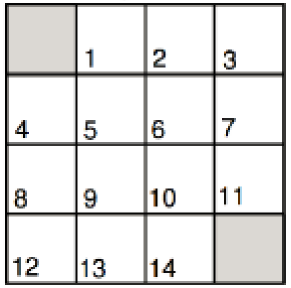


- 策略评估
  - 估计V^{π}
  - 迭代的评估策略
- 策略改进
  - 生成 $\pi' \geq \pi$
  - 贪心策略改进

举例：策略评估



动作



- 非折扣MDP（$\gamma = 1$）
- 非终止状态：1, 2, ...,14
- 两个终止状态（灰色方格）
- 如果动作指向所有方格以外，则这一步不动
- 奖励均为-1，直到到达终止状态
- 智能体的策略为均匀随机策略

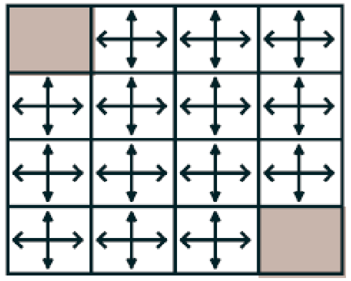
举例：策略评估

随机策略的 V_k

V_k对应的贪心策略

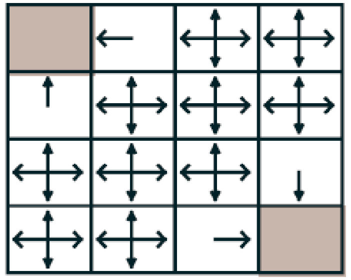
K=0

0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0



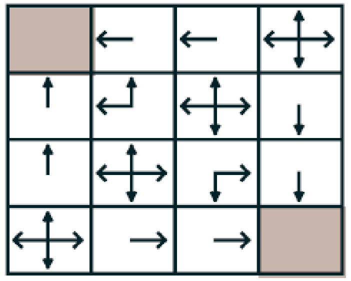
K=1

0.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	0.0



K=2

0.0	-1.7	-2.0	-2.0
-1.7	-2.0	-2.0	-2.0
-2.0	-2.0	-2.0	-1.7
-2.0	-2.0	-1.7	0.0



举例：策略评估

随机策略的 V_k

V_k对应的贪心策略

K=3

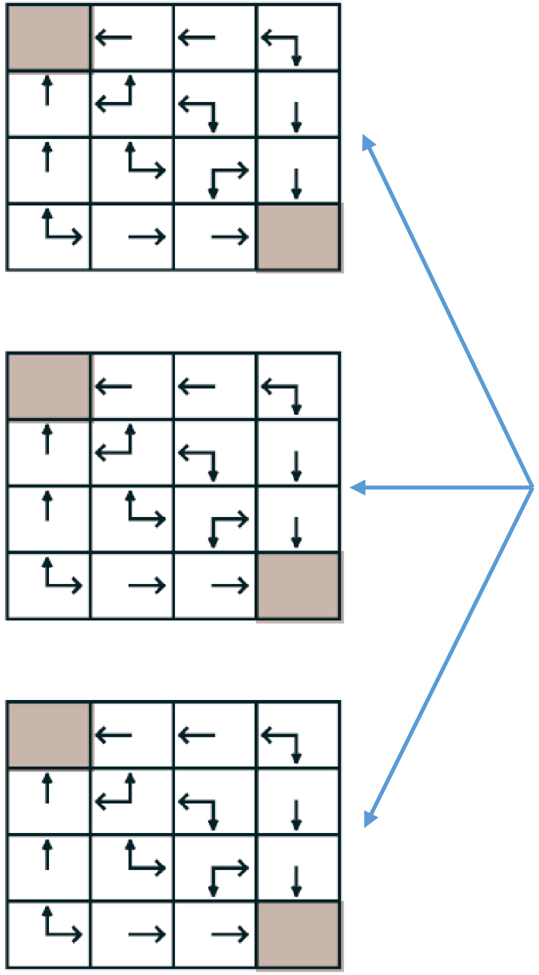
0.0	-2.4	-2.9	-3.0
-2.4	-2.9	-3.0	-2.9
-2.9	-3.0	-2.9	-2.4
-3.0	-2.9	-2.4	0.0

K=10

0.0	-6.1	-8.4	-9.0
-6.1	-7.7	-8.4	-8.4
-8.4	-8.4	-7.7	-6.1
-9.0	-8.4	-6.1	0.0

K=∞

0.0	-14.	-20.	-22.
-14.	-18.	-20.	-20.
-20.	-20.	-18.	-14.
-22.	-20.	-14.	0.0



$V := V^{\pi}$

最优策略

如何加速策略迭代：价值迭代！

- 对于一个有限状态和动作空间的MDP

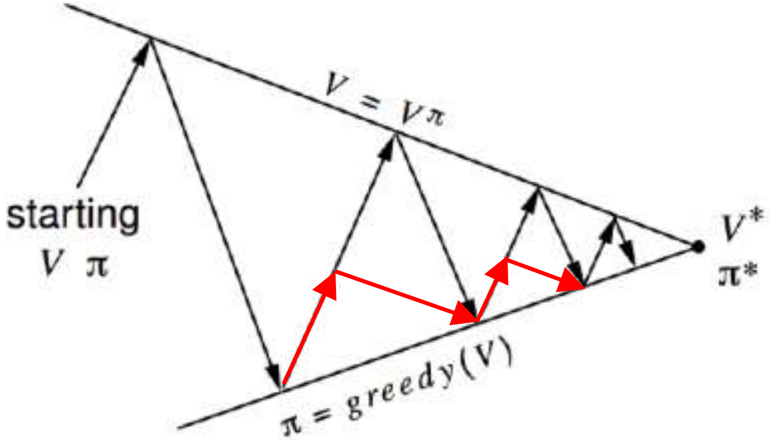
$$|S| < \infty, |A| < \infty$$

- 价值迭代过程（基于V价值函数）

1. 随机初始化策略π
2. 重复以下过程直到收敛 {
   a) 让 $V \coloneqq V^{\pi}$
   b) 对每个状态，更新

$$\pi(s) = arg\max_{a \in A} r(s,a) + \gamma \sum_{s' \in S} P_{sa}(s')V(s')$$

}



更新价值函数会很耗时，但前面的例子中，迭代计算 V 价值函数为收敛时，其导出的策略已经是最优

价值迭代

- 对于一个有限状态和动作空间的MDP

$$|S| < \infty, |A| < \infty$$

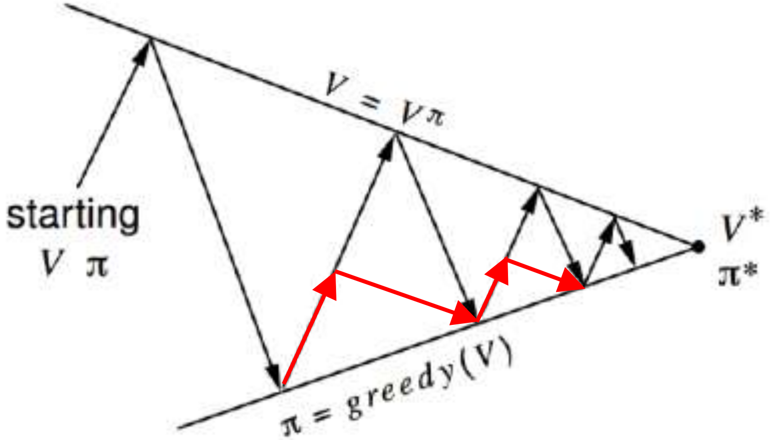
- 价值迭代过程（基于V价值函数）

1. 对每个状态s，初始化 $V(s) = 0$
2. 重复以下过程直到收敛 {

   对每个状态，更新

$$V(s) = R(s) + \max_{a \in A} \gamma \sum_{s' \in S} P_{sa}(s') V(s')$$

}



计算过程中没有明确的策略

收敛性证明：https://towardsdatascience.com/mathematical-analysis-of-reinforcement-learning-bellman-equation-ac9f0954e19f

最优价值函数

- 对状态s来说的最优价值函数是所有策略中可获得的最大可能折扣奖励的和

$$V^*(s) = \max_{\pi} V^{\pi}(s)$$

- 最优价值函数的Bellman等式

$$V^*(s) = R(s) + \max_{a\in A} \gamma \sum_{s'\in S} P_{sa}(s') V^*(s')$$

- 最优策略

$$\pi^*(s) = \arg\max_{a\in A} \sum_{s'\in S} P_{sa}(s') V^*(s')$$

- 对状态s和策略π

$$V^*(s) = V^{\pi^*}(\mathrm{s}) \geq V^{\pi}(s)$$

最优价值函数

价值迭代

1. 对每个状态s，初始化 $V(s) = 0$
2. 重复以下过程直到收敛 {

   对每个状态，更新

$$V(s) = R(s) + \max_{a \in A} \gamma \sum_{s' \in S} P_{sa}(s') V(s')$$

}

策略迭代

1. 随机初始化策略 π
2. 重复以下过程直到收敛 {
   a) 让 $V := V^{\pi}$
   b) 对每个状态，更新

$$\pi(s) = arg \max_{a \in A} \sum_{s' \in S} P_{sa}(s') V(s')$$

}

备注：

1. 价值迭代是贪心更新法
2. 策略迭代中，用Bellman等式更新价值函数代价很大
3. 对于空间较小的MDP，策略迭代通常很快收敛
4. 对于空间较大的MDP，价值迭代更实用（效率更高）
5. 如果没有状态转移循环，最好使用价值迭代

总结

强化学习
问题建模

强化学习
优化求解

随机过程

马尔可夫过程
$P(s_{t+1}|s_t)$

马尔可夫奖励过程
$P_s^{s'}, r, \gamma$

马尔可夫决策过程
$\mathcal{S}, \mathcal{A}, P_{s,a}^{s'}, r, \gamma$

学习目标

$$\max_\pi \mathbb{E}[r(S_0,A_0)+\gamma r(S_1,A_1)+\gamma^2 r(S_2,A_2)+\cdots|s_0,\pi]$$
$$=\sum_{s,a}\rho^\pi(s,a)r(s,a)=\mathbb{E}_\pi[r(s,a)]$$

引申

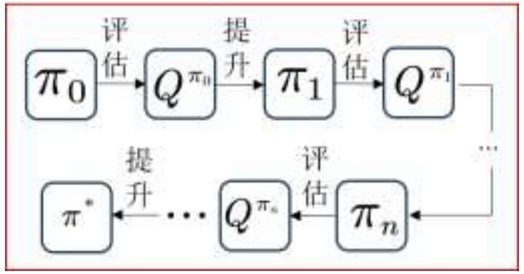
（策略）评价标准

$$V^\pi(s)=\mathbb{E}_{a\sim\pi(s)}\left[r(s,a)+\gamma\sum_{s'\in\mathcal{S}}P_{s,a}(s')V^\pi(s')\right]$$
$$Q^\pi(s,a)=r(s,a)+\gamma\sum_{s'\in\mathcal{S}}P_{s,a}(s')\sum_{a'\in\mathcal{A}}\pi(a'|s')Q^\pi(s',a')$$

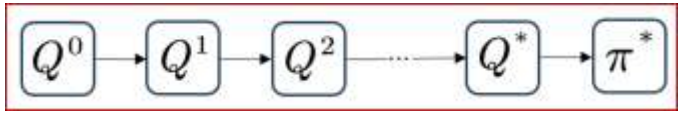
策略提升定理

$$Q^\pi(s,\pi'(s))\geq V^\pi(s), \quad \forall s\in\mathcal{S}$$
$$V^{\pi'}(s)\geq V^\pi(s), \quad \forall s\in\mathcal{S}$$

策略迭代



价值迭代



Q & A