第二讲：马尔可夫决策过程

主讲人 陈达贵

清华大学自动化系

在读硕士

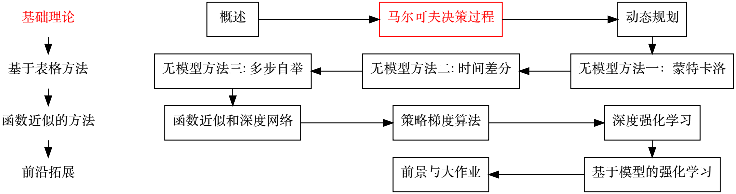
深蓝学院
shenlanxueyuan.com

强化学习理论与实践

2018-12-14

目录

章节目录


基础理论
基于表格方法
函数近似的方法
前沿拓展
概述
马尔可夫决策过程
动态规划
无模型方法一：蒙特卡洛
无模型方法二: 时间差分
无模型方法三: 多步自举
函数近似和深度网络
策略梯度算法
深度强化学习
基于模型的强化学习
前景与大作业

本章目录

马尔可夫决策过程简介

- 马尔可夫决策过程 (Markov Decision Processes, MDPs)是对强化学习问题的数学描述.
- 要求环境是全观测的. (原因后述)
- 几乎所有的 RL 问题都能用 MDPs 来描述
  - 最优控制问题可以描述成连续 MDPs
  - 部分观测环境可以转化成 POMDPs
  - 赌博机问题是只有一个状态的 MDPs

注: 虽然大部分 RL 问题都可以转化成 MDPs, 但是在我们这门课所描述的 MDPs 是在全观测的情况

数学规范

- 用大写字母表示随机变量：S, A, R 等
- 用小写字母表示某一个具体的值：s, a, r 等
- 用空心字母表示统计运算符：$\mathbb{E}, \mathbb{P}$ 等
- 用花体字母表示集合或函数：$\mathcal{S}, \mathcal{A}, \mathcal{P}$ 等

目录

马尔可夫性

只要知道现在，将来和过去条件独立

定义

如果在 t 时刻的状态 S_t 满足如下等式，那么这个状态被称为马尔可夫状态，或者说该状态满足马尔可夫性.

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

马尔可夫性

- 状态 S_t 包含了所有历史相关信息
- 或者说历史的所有状态的相关信息都在当前状态 S_t 上体现出来
- 一旦 S_t 知道了，那么 $S_1, S_2, \ldots, S_{t-1}$ 都可以被抛弃
- 数学上可以认为：状态是将来的充分统计量
- 因此，这里要求环境全观测
- 例子
  1. 下棋时，只用关心当前局面
  2. 打俄罗斯方块时，只用关心当前屏幕
- 有了马尔可夫状态之后
  - 定义状态转移矩阵
  - 忽略时间的影响

注：这里的相关指与问题相关，可能有一些问题无关的信息没有在 S_t 中

注：马尔可夫性和状态的定义息息相关

状态转移矩阵

定义

状态转移概率指从一个马尔可夫状态s 跳转到后继状态 (successor state)s' 的概率

$$\mathcal{P}_{ss'} = \mathbb{P}\left[S_{t+1} = s' | S_t = s\right]$$

所有的状态组成行，所有的后继状态组成列，我们得到状态转移矩阵

$$\mathcal{P} = \begin{bmatrix} \mathcal{P}_{11} & \dots & \mathcal{P}_{1n} \\ \vdots & \ddots & \vdots \\ \mathcal{P}_{n1} & \dots & \mathcal{P}_{nn} \end{bmatrix}$$

- n 表示状态的个数
- 由于 $\mathcal{P}$ 代表了整个状态转移的集合，所以用花体
- 每行元素相加等于 1

状态转移函数

我们也可以将状态转移概率写成函数的形式

$$\mathcal{P}(s'|s) = \mathbb{P}\left[S_{t+1} = s' | S_t = s\right]$$

- $\sum_{s'} \mathcal{P}(s'|s) = 1$
- 状态数量太多或者是无穷大（连续状态）时，更适合使用状态转移函数，此时 $\int_{s'} \mathcal{P}(s'|s) = 1$

马尔可夫过程

一个马尔可夫过程 (Markov process, MP) 是一个无记忆的随机过程，即一些马尔可夫状态的序列

定义

马尔可夫过程可以由一个二元组来定义 $\langle \mathcal{S}, \mathcal{P} \rangle$.

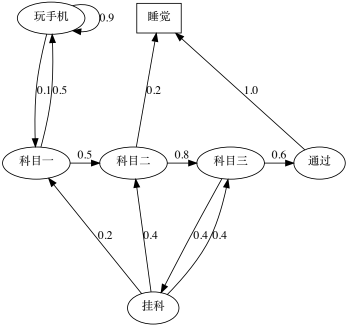
- $\mathcal{S}$ 代表了状态的集合
- $\mathcal{P}$ 描述了状态转移矩阵

$$\mathcal{P}_{ss'} = \mathbb{P}\left[S_{t+1} = s' | S_t = s\right]$$

注: 虽然我们有时候并不知道 $\mathcal{P}$ 的具体值，但是通常我们假设 $\mathcal{P}$ 存在且稳定的

注: 当 $\mathcal{P}$ 不稳定时，不稳定环境，在线学习，快速学习

马尔可夫过程的例子[1]



- 一个学生每天需要学习三个科目，然后通过测验
- 不过也有可能只学完两个科目之后直接睡觉
- 一旦挂科有可能需要重新学习某些科目
- 用椭圆表示普通状态，每一条线上的数字表示从一个状态跳转到另一个状态的概率
- 方块表示终止（terminal）状态
- 终止状态的定义有两种：
  - 时间终止
  - 状态终止

由于马尔可夫过程可以用图中的方块和线条组成，所以可以称马尔可夫过程为马尔可夫链 (MDPs chain)

[1] 例子改编自 David Silver 2015

片段

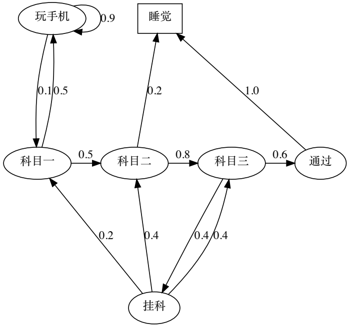
定义

强化学习中，从初始状态 S_1 到终止状态的序列过程，被称为一个片段 (episode)

$$S_1, S_2, \ldots, S_T$$

- 如果一个任务总以终止状态结束，那么这个任务被称为片段任务 (episodic task)
- 如果一个任务会没有终止状态，会被无限执行下去，这被称为连续性任务 (continuing task)

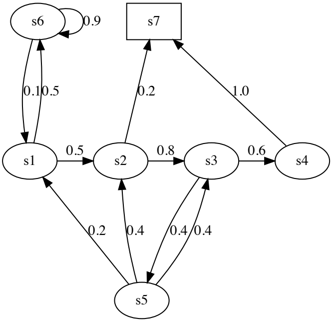
马尔可夫链的例子



假设初始状态是“科目一”，从这个马尔可夫链中，我们可能采样到如下的片段

- “科目一”，“科目二”，“睡觉”
- “科目一”，“科目二”，“科目三”，“通过”，“睡觉”
- “科目一”，“玩手机”，“玩手机”，...，“玩手机”，“科目一”，“科目二”，“睡觉”
- “科目一”，“科目二”，“科目三”，“挂科”，“科目二”，“科目三”，“挂科”，“科目一”，“科目二”，“科目三”，“挂科”，“科目三”，“通过”，“睡觉”

马尔可夫链的例子: 转移矩阵



分别用 $s_1 \sim s_7$ 代表 “科目一”, “科目二”, “科目三”, “通过”, “挂科”, “玩手机”, “睡觉”

$$\mathcal{P} = \begin{bmatrix} & 0.5 & & & & 0.5 & \\ & & 0.8 & & & & 0.2 \\ & & & 0.6 & 0.4 & & \\ & & & & & & 1.0 \\ 0.2 & 0.4 & 0.4 & & & & \\ 0.1 & & & & & 0.9 & \\ & & & & & & 1.0 \end{bmatrix}$$

目录

马尔可夫奖励过程

马尔可夫过程主要描述的是状态之间的转移关系，在这个转移关系上赋予不同的奖励值即得到了马尔可夫**奖励**过程

定义

马尔可夫奖励 (Markov Reward Process, MRP) 过程由一个四元组组成 $\langle \mathcal{S}, \mathcal{P}, \mathcal{R}, \gamma \rangle$

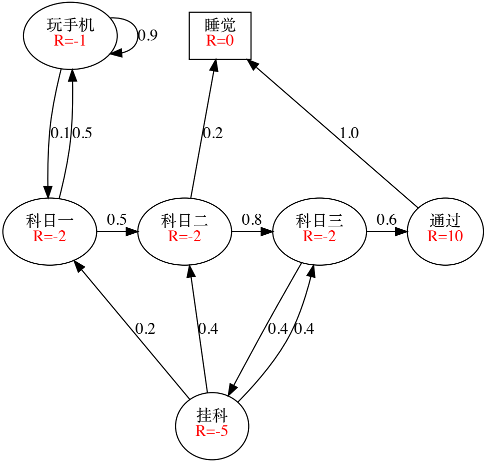
- $\mathcal{S}$ 代表了状态的集合
- $\mathcal{P}$ 描述了状态转移矩阵

$$\mathcal{P}_{ss'} = \mathbb{P}\left[S_{t+1} = s' | S_t = s\right]$$

- $\mathcal{R}$ 表示奖励函数，$\mathcal{R}(s)$ 描述了在状态 s 的奖励，$\mathcal{R}(s) = \mathbb{E}\left[R_{t+1} | S_t = s\right]$
- γ 表示衰减因子，$\gamma \in [0, 1]$

注: 需要注意 $\mathcal{R}$ 和 R 的区别

马尔可夫奖励过程例子



玩手机
R=-1
0.9
睡觉
R=0
0.1
0.5
0.2
1.0
科目一
R=-2
0.5
科目二
R=-2
0.8
科目三
R=-2
0.6
通过
R=10
0.2
0.4
0.4
0.4
挂科
R=-5

回报值

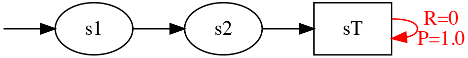
- 奖励值：对每一个状态的评价
- 回报值：对每一个片段的评价

定义

回报值 (return G_t) 是从时间 t 处开始的累计衰减奖励

- 对于片段性任务：
  $G_t = R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{T-t-1} R_T = \sum_{k=0}^{T-t-1} \gamma^k R_{t+k+1}$
- 对于连续性任务：$G_t = R_{t+1} + \gamma R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$

再聊片段



- 终止状态等价于自身转移概率为 1，奖励为 0 的的状态
- 因此我们能够将片段性任务和连续性任务统一表达

$$G_t = \sum_{k=0}^{T-t-1} \gamma^k R_{t+k+1}$$

这里当 $T = \infty$ 时，表示连续性任务，否则即为片段性任务

再聊衰减值

为什么我们要使用这样的指数衰减值？

- 直观感觉
  - 影响未来的因素不仅仅包含当前
  - 我们对未来的把握也是逐渐衰减的
  - 一般情况下，我们更关注短时间的反馈
- 数学便利
  - 一个参数就描述了整个衰减过程，只需要调节这一个参数 γ 即可以调节长时奖励和短时奖励的权衡 (trade-off)
  - 指数衰减形式又很容易进行数学分析
  - 指数衰减是对回报值的有界保证，避免了循环 MRP 和连续性 MRP 情况下回报值变成无穷大

注: 在一些特殊情况，也可以使用 $\gamma = 1$ 的回报值

值函数

- 为什么要值函数？
  - 回报值是一次片段的结果，存在很大的样本偏差
  - 回报值的角标是 t，值函数关注的是状态 s

定义

一个 MRP 的值函数如下定义

$$v(s) = \mathbb{E}\left[G_t | S_t = s\right]$$

- 这里的值函数针对的是状态 s，所以称为状态值函数，又称 V 函数
- G_t 是一个随机变量
- 这里使用小写的 v 函数，代表了真实存在的值函数

例子：值函数和回报值

针对初始状态 S_1 =“科目一”，且 $\gamma = \frac{1}{2}$ 计算不同片段的回报值

- “科目一”，“科目二”，“睡觉”

$$g_1 = -2 + 0.5 \times -2 = -3$$

- “科目一”，“科目二”，“科目三”，“通过”，“睡觉”

$$g_1 = -2 + 0.5 \times -2 + 0.5^2 \times -2 + 0.5^3 \times 10 = -2.25$$

- “科目一”，“玩手机”，“玩手机”，“玩手机”，“科目一”，“科目二”，“睡觉”

$$g_1 = -2 + 0.5 \times -1 + 0.5^2 \times -1 + 0.5^3 \times -1 + 0.5^4 \times -2 + 0.5^5 \times -2 = -3.0625$$

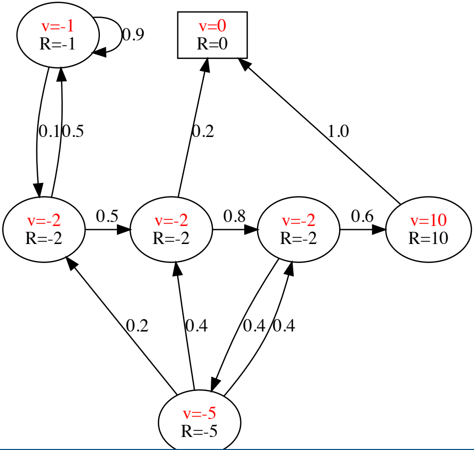
- “科目一”，“科目二”，“科目三”，“挂科”，“科目二”，“科目三”，“挂科”，“科目一”，“科目二”，“科目三”，“挂科”，“科目三”，“通过”，“睡觉”

$$g_1 = -2 + 0.5 \times -2 + 0.5^2 \times -2 + \cdots = -4.42138671875$$

虽然都是从相同的初始状态开始，但是不同的片段有不同的回报值，而值函数是它们的期望值。

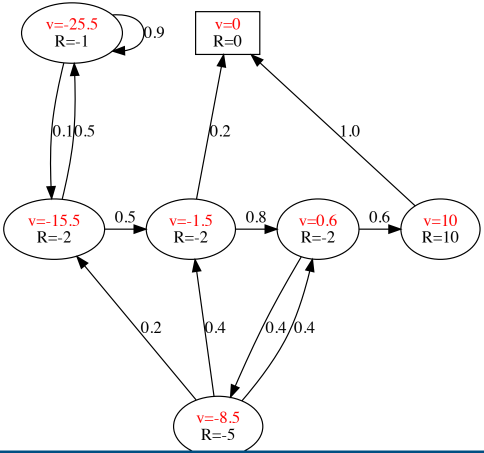
针对例子的 V 函数值

当$\gamma = 0$时



针对例子的 V 函数值

当$\gamma = 1$时



MRPs 中的贝尔曼方程

值函数的表达式可以分解成两部分

- 瞬时奖励 R_{t+1}
- 后继状态 S_{t+1} 的值函数乘上一个衰减系数

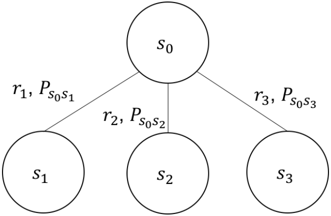
$$
\begin{aligned}
v(s) &= \mathbb{E}\left[G_t | S_t = s\right] \\
&= \mathbb{E}\left[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \ldots | S_t = s\right] \\
&= \mathbb{E}\left[R_{t+1} + \gamma\left(R_{t+2} + \gamma R_{t+3} + \ldots\right) | S_t = s\right] \\
&= \mathbb{E}\left[R_{t+1} + \gamma G_{t+1} | S_t = s\right] \\
&= \mathbb{E}\left[R_{t+1} + \gamma v(S_{t+1}) | S_t = s\right]
\end{aligned}
$$

- 体现了 $v(s)$ 与 $v(S_{t+1})$ 之间的迭代关系
- 这是强化学习中的理论核心之一
- 注意 s 小写，S_{t+1} 大写

MRPs 中的贝尔曼方程

如果我们已知转移矩阵 $\mathcal{P}$，那么

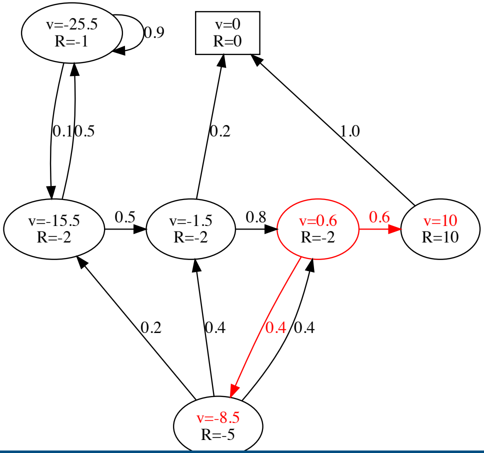
$$
\begin{aligned}
v(s) &= \mathbb{E}\left[R_{t+1} + \gamma v(S_{t+1}) | S_t = s\right] \\
&= \mathbb{E}\left[R_{t+1} | S_t = s\right] + \gamma \mathbb{E}\left[v(S_{t+1}) | S_t = s\right] \\
&= \mathcal{R}(s) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'} v(s')
\end{aligned}
$$



图中我们可以用 $v(s_1), v(s_2), v(s_3)$ 去计算 $v(s_0)$

针对例子的贝尔曼方程

当 $\gamma = 1$ 时 $0.6 = -2 + 0.6 \times 10 + 0.4 \times -8.5$



贝尔曼方程的矩阵形式

使用矩阵-向量的形式表达贝尔曼方程，即

$$v = \mathcal{R} + \gamma \mathcal{P} v$$

假设状态集合为 $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$，那么

$$\begin{bmatrix} v(s_1) \\ \vdots \\ v(s_n) \end{bmatrix} = \begin{bmatrix} \mathcal{R}(s_1) \\ \vdots \\ \mathcal{R}(s_n) \end{bmatrix} + \gamma \begin{bmatrix} \mathcal{P}_{s_1 s_1} & \cdots & \mathcal{P}_{s_1 s_n} \\ \vdots & \ddots & \vdots \\ \mathcal{P}_{s_n s_1} & \cdots & \mathcal{P}_{s_n s_n} \end{bmatrix} \begin{bmatrix} v(s_1) \\ \vdots \\ v(s_n) \end{bmatrix}$$

解贝尔曼方程

贝尔曼方程本质上是一个线性方程，可以直接解

$$
\begin{aligned}
v &= \mathcal{R} + \gamma \mathcal{P} v \\
(I - \gamma \mathcal{P}) v &= \mathcal{R} \\
v &= (I - \gamma \mathcal{P})^{-1} \mathcal{R}
\end{aligned}
$$

- 计算复杂度 $O(n^3)$
- 要求已知状态转移矩阵 $\mathcal{P}$
- 直接求解的方式仅限于小的 MRPs

目录

与 MP 和 MRP 的区别

- MP 和 MRP 中，我们都是作为观察者，去观察其中的状态转移现象，去计算回报值
- 对于一个 RL 问题，我们更希望去改变状态转移的流程，去最大化回报值
- 通过在 MRP 中引入决策即得到了马尔可夫决策过程（Markov Decision Processes, MDPs）

马尔可夫决策过程

定义

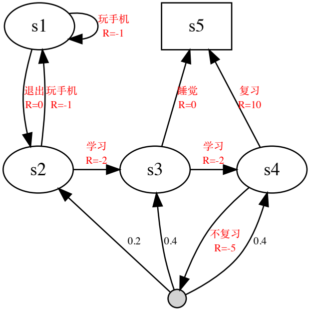
一个马尔可夫决策过程 (MDPs) 由一个五元组构成 $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$

- $\mathcal{S}$ 代表了状态的集合
- $\mathcal{A}$ 代表了动作的集合
- $\mathcal{P}$ 描述了状态转移矩阵

$$\mathcal{P}_{ss'}^{a} = \mathbb{P}\left[S_{t+1} = s' | S_t = s, A_t = a\right]$$

- $\mathcal{R}$ 表示奖励函数，$\mathcal{R}(s, a)$ 描述了在状态 s做动作 a的奖励，$\mathcal{R}(s, a) = \mathbb{E}\left[R_{t+1} | S_t = s, A_t = a\right]$
- γ 表示衰减因子，$\gamma \in [0, 1]$

例子



- 针对状态的奖励变成了针对 $\langle s, a \rangle$ 的奖励
- 能通过动作进行控制的状态转移由原来的状态转移概率替换成了动作
- MDP 只关注哪些可以做出决策的动作
- 被动的状态转移关系被压缩成一个状态
  - “被动”指无论做任何动作，状态都会发生跳转。这样的状态不属于 MDPs 中考虑的状态
  - 原图中，由于“通过”后一定去“睡觉”因此进行了压缩
  - 原图中，“挂科”后的跳转不受控制

注: 图中除了在“科目三”的状态上执行“不复习”动作外，其他的所有状态跳转都是确定性的，我们通过在不同的状态上执行不同的动作，即可实现不同的状态跳转

策略

我们将之前 MRPs 中的状态转移矩阵分成了两个部分

- 能被智能体控制的策略, Policy
- MDPs 中的转移矩阵$\mathcal{P}$ (不受智能体控制，认为是环境的一部分)

定义

在 MDPs 中，一个策略 (Policy)π是在给定状态下的动作的概率分布

$$\pi(a|s) = \mathbb{P}\left[A_t = a | S_t = s\right]$$

$$\pi(a|s) = \begin{array}{c|cccc} & s_1 & s_2 & s_3 & s_4 \\ \hline a_1 & 0 & 0.1 & 0.5 & 1 \\ a_2 & 0.2 & 0.1 & 0.5 & 0 \\ a_3 & 0.7 & 0.1 & 0 & 0 \\ a_4 & 0.1 & 0.7 & 0 & 0 \end{array}$$

策略的特点

- 策略是对智能体行为的全部描述
- MDPs 中的策略是基于马尔可夫状态的（而不是基于历史）
- 策略是时间稳定的，只与 s 有关，与时间 t 无关
- 策略是 RL 问题的终极目标
- 如果策略的概率分布输出都是独热的 (one-hot)[2]的，那么称为确定性策略，否则即为随机策略

$$\pi(a|s) = \begin{array}{c|cccc} & s_1 & s_2 & s_3 & s_4 \\ a_1 & 0 & 0 & 1 & 1 \\ a_2 & 0 & 0 & 0 & 0 \\ a_3 & 1 & 0 & 0 & 0 \\ a_4 & 0 & 1 & 0 & 0 \end{array} \Longleftrightarrow \begin{array}{cc} s_1 & a_3 \\ s_2 & a_4 \\ s_3 & a_1 \\ s_4 & a_1 \end{array}$$

[2] one-hot 指一个向量只有一个元素为 1，其他均为 0

MDPs 和 MRPs 之间的关系

对于一个 MDP 问题 $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$，如果给定了策略 π

- MDP 将会退化成 MRP，$\langle \mathcal{S}, \mathcal{P}^\pi, \mathcal{R}^\pi, \gamma \rangle$
- 此时

$$\mathcal{P}^\pi_{ss'} = \sum_{a \in \mathcal{A}} \pi(a|s) \mathcal{P}^a_{ss'}$$

$$\mathcal{R}^\pi_s = \sum_{a \in \mathcal{A}} \pi(a|s) \mathcal{R}^a_s$$

注：注意由于策略 π 代表 1，一个动作，2，一个动作分布，所以在有时，π 和 a 会存在一定的混淆。

MDPs 中的值函数

在 MDPs 问题中，由于动作的引入，值函数分为了两种：1，状态值函数（V 函数）2，状态动作值函数 (Q 函数)

定义

MDPs 中的状态值函数是从状态 s 开始，使用策略 π 得到的期望回报值

$$v_\pi(s) = \mathbb{E}_\pi\left[G_t | S_t = s\right]$$

定义

MDPs 中的状态动作值函数是从状态 s 开始，执行动作 a，然后使用策略 π 得到的期望回报值

$$q_\pi(s, a) = \mathbb{E}_\pi\left[G_t | S_t = s, A_t = a\right]$$

注:MDPs 中，任何不说明策略 π 的情况下，讨论值函数都是在耍流氓!

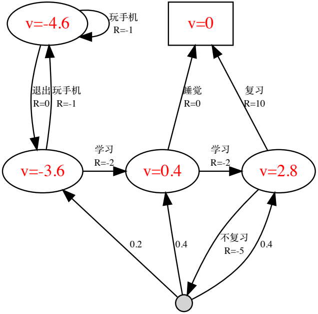
贝尔曼期望方程

和 MRP 相似，MDPs 中的值函数也能分解成瞬时奖励和后继状态的值函数两部分

$$v_\pi(s) = \mathbb{E}_\pi\left[R_{t+1} + \gamma v_\pi(S_{t+1}) | S_t = s\right]$$

$$q_\pi(s, a) = \mathbb{E}_\pi\left[R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1}) | S_t = s, A_t = a\right]$$

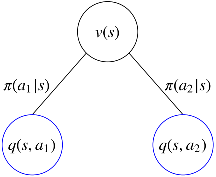
例子



- 当 $\pi(a|s) = 0.5\forall a, s$ 且 $\gamma = 1$ 时，$v_\pi(s)$ 如图
- 将图中的圆圈状态从上到下从左到右分别记为 s_1, s_2, s_3, s_4，将终止状态记为 s_5，则当 $\pi(a|s) = 0.5\forall a, s$ 时，状态转移矩阵为

$$
\mathcal{P}^\pi = \begin{bmatrix} 0.5 & 0.5 & & & \\ 0.5 & & 0.5 & & \\ & & & 0.5 & 0.5 \\ & 0.1 & 0.2 & 0.2 & 0.5 \\ & & & & 1 \end{bmatrix}
$$

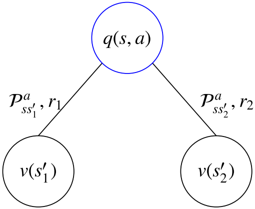
V 函数与 Q 函数之间的相互转化



$$v_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) q_\pi(s, a)$$

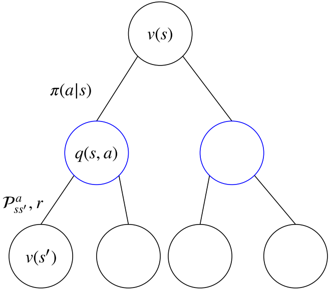
注: 本质上是全概率公式

V 函数与 Q 函数之间的相互转化 2



$$q_\pi(s,a) = \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}^a_{ss'} v_\pi(s')$$

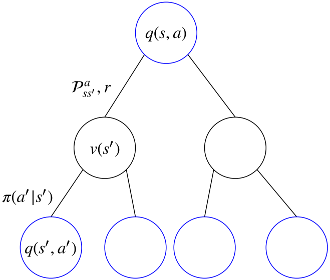
贝尔曼期望方程-V 函数



$$v_\pi(s) = \sum_{a\in\mathcal{A}} \pi(a|s) q_\pi(s,a) = \sum_{a\in\mathcal{A}} \pi(a|s) \left(\mathcal{R}(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathcal{P}^a_{ss'} v_\pi(s') \right)$$

实际上等价于 $v_\pi(s) = \mathbb{E}_\pi\left[R_{t+1} + \gamma v_\pi(S_{t+1}) | S_t = s\right]$，**为什么**？

贝尔曼期望方程-Q 函数



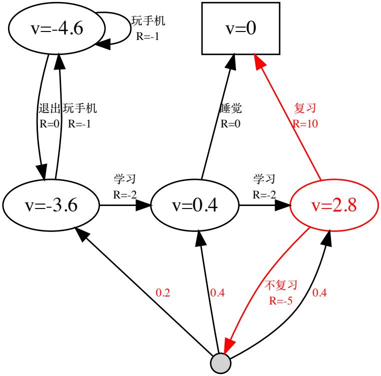
$$q_\pi(s,a) = \mathcal{R}(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathcal{P}^a_{ss'} \sum_{a'\in\mathcal{A}} \pi(a'|s') q_\pi(s',a')$$

实际上等价于 $q_\pi(s,a) = \mathbb{E}_\pi\left[R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1}) | S_t = s, A_t = a\right]$，**为什么？**

贝尔曼期望方程例子

当 $\gamma = 1, \pi(a|s) = 0.5$ 时

$$2.8 = 0.5 \times (10 + 0) + 0.5 \times (-5 + 0.2 \times -3.6 + 0.4 \times 0.4 + 0.4 \times 2.8)$$



贝尔曼期望方程的矩阵形式

MDPs 下的贝尔曼期望方程和 MRP 的形式相同。

$$v_\pi = \mathcal{R}^\pi + \gamma \mathcal{P}^\pi v_\pi$$

同样地，可以直接求解

$$v_\pi = (I - \gamma \mathcal{P}^\pi)^{-1} \mathcal{R}^\pi$$

求解的要求：

- 已知 $\pi(a|s)$
- 已知 $\mathcal{P}_{ss'}^a$

最优值函数

之前值函数，以及贝尔曼期望方程针对的都是给定策略 π 的情况，是一个评价的问题。
现在我们来考虑强化学习中的优化问题，即找出最好的策略。

定义

最优值函数指的是在所有策略中的值函数最大值，其中包括最优 V 函数和最优 Q 函数

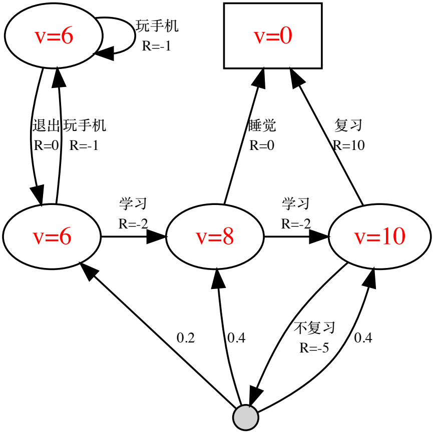
$$v_*(s) = \max_\pi v_\pi(s)$$

$$q_*(s, a) = \max_\pi q_\pi(s, a)$$

- 最优值函数指的是一个 MDP 中所能达到的最佳性能
- 如果我们找到最优值函数即相当于这个 MDP 已经解决了

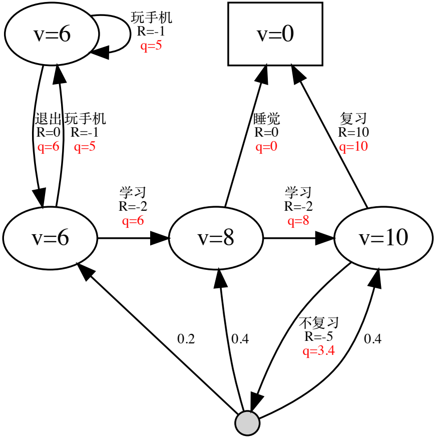
最优 V 函数

当 $\gamma = 1$ 时的最优 V 函数 $v_*(s)$



最优 Q 函数

当 $\gamma = 1$ 时的最优 Q 函数 $q_*(s, a)$



最优策略

为了比较不同策略的好坏，我们首先应该定义策略的比较关系

$$\pi \geq \pi' \quad \text{if} \quad v_{\pi}(s) \geq v_{\pi'}(s), \forall s$$

定理

对于任何 MDPs 问题

- 总存在一个策略 π_* 要好于或等于其他所有的策略， $\pi_* \geq \pi, \forall \pi$
- 所有的最优策略都能够实现最优的 V 函数 $v_{\pi_*}(s) = v_*(s)$
- 所有的最优策略都能够实现最优的 Q 函数 $q_{\pi_*}(s, a) = q_*(s, a)$

注: 具体证明参考 Total Expected Discounted Reward MDPs: Existence of Optimal Policies

怎么得到最优策略?

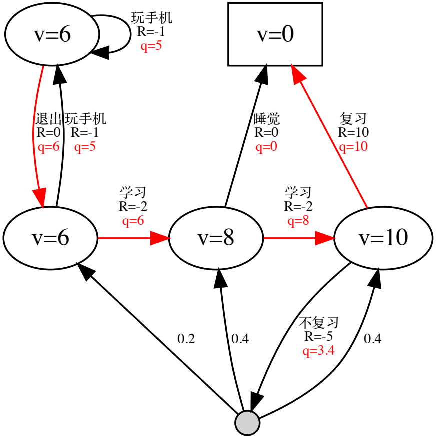
当我们已知了最优 Q 函数后，我们能够马上求出最优策略，只要根据 $q_*(s,a)$ 选择相应的动作即可。

$$\pi_*(a|s) = \begin{cases} 1 & \text{if } a = \arg\max_{a \in \mathcal{A}} q_*(s,a) \\ 0 & \text{otherwise} \end{cases}$$

- 可以看出对于任何 MDPs 问题，总存在一个确定性的最优策略
- 如果已知最优 V 函数，能不能找到最优策略呢？

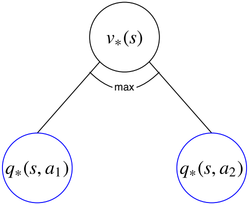
最优策略例子

当 $\gamma = 1$ 时的最优策略 $\pi_*(a|s)$



v_* 与 q_* 的相互转化

- 之前我们已经探讨了 $v_\pi(s)$ 和 $q_\pi(s,a)$ 之间的关系——贝尔曼期望方程
- 同样地，$v_*(s)$ 和 $q_*(s,a)$ 也存在递归的关系——贝尔曼**最优**方程

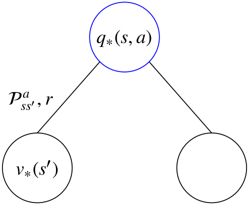


$$v_*(s) = \max_a q_*(s,a)$$

和贝尔曼期望方程的关系

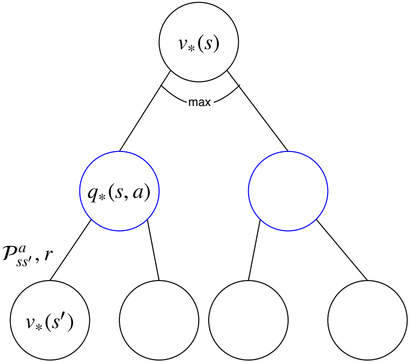
$$v_*(s) = v_{\pi_*}(s) = \sum_{a\in\mathcal{A}} \pi_*(a|s) q_{\pi_*}(s,a) = \max_a q_{\pi_*}(s,a) = \max_a q_*(s,a)$$

v_* 与 q_* 的相互转化 2



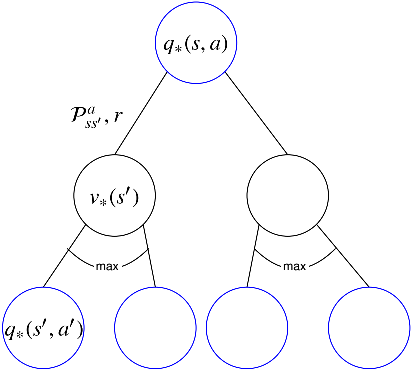
$$q_*(s, a) = \mathcal{R}(s, a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^{a} v_*(s')$$

贝尔曼最优方程——V 函数



$$v_*(s) = \max_a \left[\mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_*(s') \right]$$

贝尔曼最优方程——Q 函数



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

和贝尔曼期望方程的关系

- 贝尔曼最优方程本质上就是利用了 π_* 的特点，将求期望的算子转化成了 $\max_a$
- 在贝尔曼期望方程中，π 是已知的。而在贝尔曼最优方程中，π_* 是未知的
- 解贝尔曼期望方程的过程即对应了评价，解贝尔曼最优方程的过程即对应了优化

解贝尔曼最优方程

- 贝尔曼最优方程不是线性的
- 一般很难有闭式的解
- 可以使用迭代优化的方法去解
  - 值迭代
  - 策略迭代
  - Q 学习
  - SARSA
  - ...

目录

MDPs 的拓展

- 无穷或连续 MDPs
- 部分可观测 MDPs（Partially observable MDPs, POMDPs）
- 无衰减 MDPs

无穷或连续 MDPs

- 动作空间或状态空间无限可数
- 动作空间或状态空间无限不可数 (连续)
- 时间连续

POMDPs

- 此时观测不等于状态 $O_t \neq S_t$
- POMDPs 由七元组构成 $< \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{P}, \mathcal{R}, \mathcal{Z}, \gamma >$
- $\mathcal{Z}$ 是观测函数

$$\mathcal{Z}_{s'o}^{a} = \mathbb{P}\left[O_{t+1} = o | S_{t+1} = s', A_t = a\right]$$

- 观测不满足马尔可夫性，因此也不满足贝尔曼方程
- 状态未知，隐马尔可夫过程
- 有时对于 POMDPs 来说，最优的策略是随机性的

无衰减 MDPs

- 用于各态历经马尔可夫决策过程
  - 各态历经性：平稳随机过程的一种特性
- 存在独立于状态的平均奖赏 ρ^{π}
- 求值函数时，需要减去该平均奖赏，否则有可能奖赏爆炸

致谢

谢谢！