强化学习及其应用

Reinforcement Learning and Its Applications

第一章 绪 论

Introduction

授课人：周晓飞
zhouxiaofei@iie.ac.cn

2023-6-12

教师团队介绍

主讲教师：　周晓飞，研究员

研究方向：机器学习、自然语言处理、多模态智能

zhouxiaofei@iie.ac.cn

助　　教：　董林华，博士研究生；

研究方向：知识图谱、图挖掘；

donglinhua@iie.ac.cn；

课程安排

课程名称	学时	学分	限选人数/选课人数	教师	开课地点
强化学习及其应用-1	20	1.0	220/318	周晓飞	教 1-002
强化学习及其应用-2	20	1.0	220/220	周晓飞	教 1-002

- 上课时间（6 月 12 日 – 6 月 16 日）

  1 班：4 学时，13:30-15:10, 15:20-17:00

  2 班：4 学时，18:10-19:50, 20:00-21:40

- 授课方式：课堂讲授
- 考核方式：读书报告

第一章 绪 论

1.1 概述

1.2 Markov 决策过程

1.3 强化学习

1.4 课程安排

1.5 小结

第一章 绪 论

1.1 概述

1.2 Markov 决策过程

1.3 强化学习

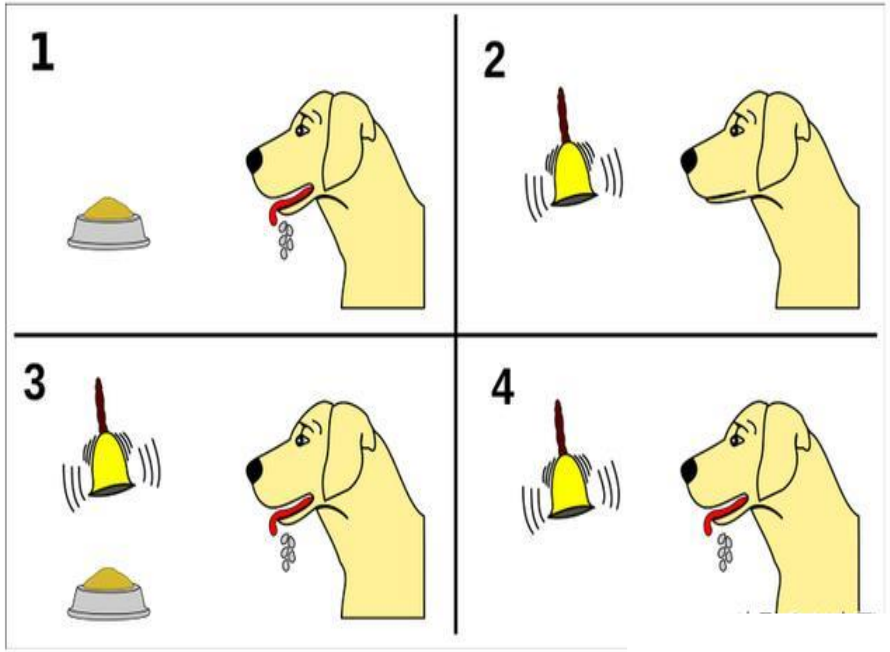
1.4 课程安排

1.5 小结

概述

动物条件反射

“强化”术语最早出自关于条件反射的描述中。条件反射是一种刺激关联强化的结果。



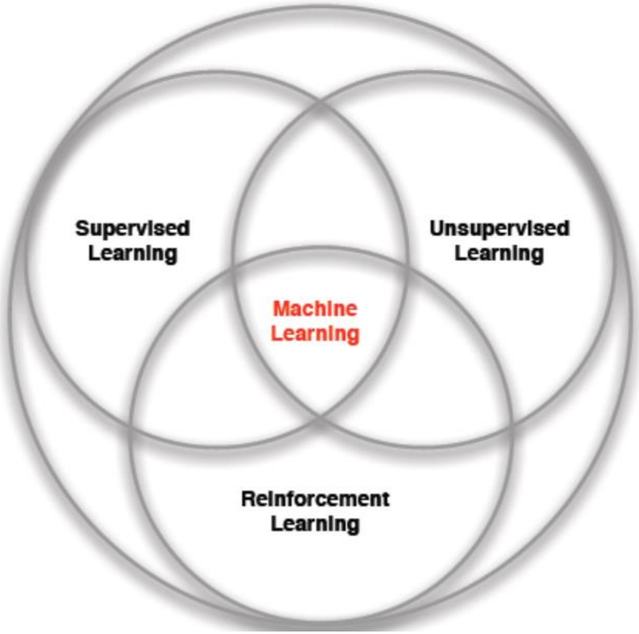
Reinforcement is the strengthening of a pattern of behavior as a result of an animal receiving a stimulus a reinforce in an appropriate temporal relationship with another stimulus or with a response。

强化就是一种行为模式的增强，它是由于动物受到一个激励（强化）与另一个激励在适当时候关联的结果。

概述

机器学习的一个研究分支

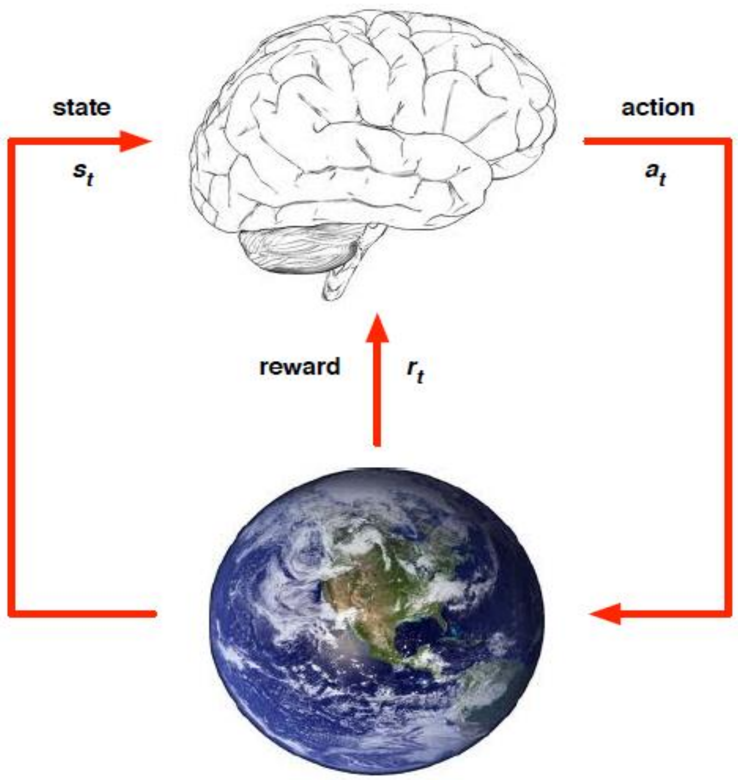
机器学习是研究如何在数据和以往经验的学习中自动改进计算机算法性能的科学



- 监督学习：有监督指导的学习类别、预测值的机器学习。
- 无监督学习：无监督指导的学习类簇、规律、特征的机器学习。
- 强化学习：通过与环境互动，获取环境反馈的样本；回报（作为监督），进行最优决策的机器学习。

概述

强化学习过程



Episodes:

S_1, a_1, R_2, S_2, a_2, R_3, S_3, a_3, R_4, … …

目标：最优策略 S→a 或 p(a/s)，获得最大回报

概述

应用范围：与环境进行交互的决策智能

--自动机器问答

--电商推荐系统

--视觉导航

--博弈

--游戏

--投资决策

。。。

概述

例子 1：回报最大化的决策

"Behind one door is tenure - behind the other is flipping burgers at McDonald's."

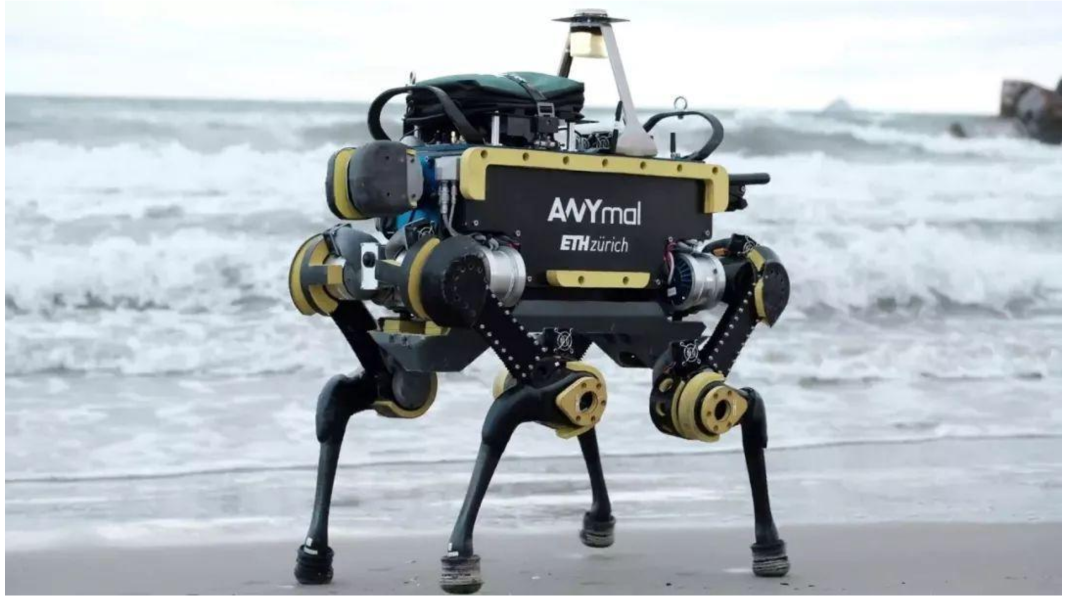
- There are two doors in front of you.
- You open the left door and get reward 0
  $V(left) = 0$
- You open the right door and get reward $+1$
  $V(right) = +1$
- You open the right door and get reward $+3$
  $V(right) = +2$
- You open the right door and get reward $+2$
  $V(right) = +2$

$\vdots$

概述

例子 2：博弈取胜的决策

例子 3：机器人



概述

强化学习的起源

现代的强化学习理论形成于 20 世纪 80 年代末

早期的强化学习的发展源于两个主要的独立分支：

- 动物行为模仿：试错学习（trial-and-error learning）（1850s-）
- 优化控制的值函数求解（1950s-）

另一个和两者相关的技术研究分支：

- 时序差分技术（1970s）

三个分支发展到 1980s 末，形成了现代强化学习理论

概述

动物行为模仿（1850s-)

试错学习的思想溯源

- 1850s ，Alexander Bain 讨论了摸索和实验的学习（groping and experiment）；
- 1894，英国心理学家 Conway Lloyd Morgan，通过摸索和实验进行动物行为观察；
- 1911，Edward Thorndike，提出效果定理，将试错作为一种学习原则。效果定理不断被延伸讨论，是许多行为研究中一个重要理论。

  “动物在同一情况下做出的几种反应中，那些伴随动物满意的反应，会与情况联系更紧密，因此当情况又出现时，它们更有可能再发生；那些伴随动物不适的反应，与这种情况的联系会减弱，因此当情况又出现时，它们就不太可能发生。满足或不适越大，相应的加强或减弱就越大。”

- 1927 年，词汇“强化”第一次描述动物学习，来自 Pavlov 关于“条件反射”的英文译著。
- 1938 年美国心理学家 R. S. Woodworth 正式提出该思想。

概述

动物行为模仿（1850s-)

试错学习的理论研究

- Thomas Ross (1933)：finding ways；
- W. Grey Walter（1951）：mechanical tortoise
- Shannon（1951, 1952）：maze-running mouse
- J. A. Deutsch （1954）：a maze-solving machine （some properties of model-based reinforcement learning）
- Marvin Minsky（1954）：SNARCs (computational models of reinforcement learning), some of which implemented trial-and-error.
- Farley and Clark (1954): described a digital simulation of a neural network learning machine that learned by trial and error. But their interests soon shifted from reinforcement learning to supervised learning (Clark and Farley, 1955).

概述

动物行为模仿（1850s-)

试错学习的理论研究

- 1948，图灵关于人工智能可能性的最早思考中，描述了一个快乐疼痛系统的设计，即在计算机中实现试错学习的想法。

  When a configuration is reached for which the action is undetermined, a random choice for the missing data is made and the appropriate entry is made in the description, tentatively, and is applied. When a pain stimulus occurs all tentative entries are cancelled, and when a pleasure stimulus occurs they are all made permanent. (Turing, 1948)

- 1960s and 1970s, 几乎很少有试错研究。

概述

优化控制的值函数求解

优化控制问题

--1957，Bellman 提出求解最优控制问题的马尔可夫决策过程（Markov Decision Process，MDP）的动态规划（Dynamic Programming）方法。该方法的求解采用了类似强化学习试错迭代求解机制，使得马尔可夫决策过程成为后来定义强化学习问题的最普遍形式。以致于后来的很多研究者都认为强化学习起源于 Bellman 的动态规划。

--1960 年，Howard 提出了求解马尔可夫决策过程的策略迭代方法。

--动态规划方法被扩展应用求解多种MDP问题（White,1985,1988,1993; Lovejoy,1991）。

概述

时序差分技术

时序差分求解方法

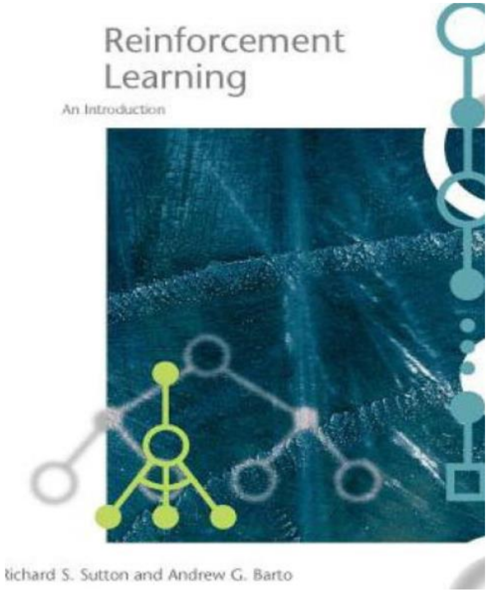
- 1950 年, Shannon 建议引入估值函数去改善试错学习
- 1959 年，Arthur Samuel 首先提出并执行了一个包括时序差分思想的学习方法
- 1961 年，Minsky 在 Shannon 建议的基础上提出了二级强化理论
- 1961-1972 年间，几乎没有关于时序差分的研究工作
- 1972 年，Klopf 完成了将试错学习与时序差分原理相结合的工作
- 1988 年，Sutton 整合提出了 TD（r）差分学习
- 1989 年，Chris Watkins 提出了 Q-Learning

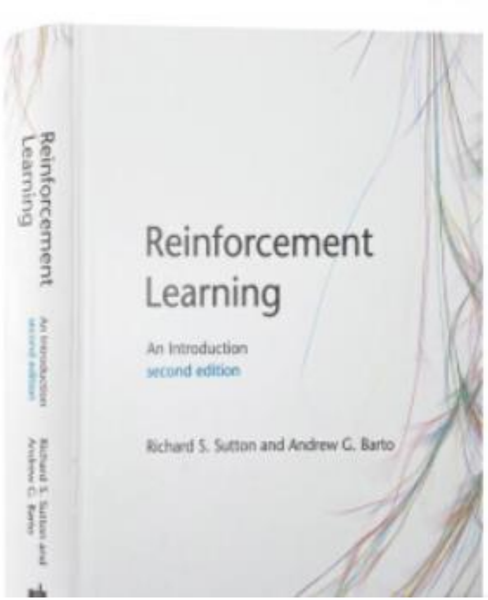
概述

现代强化学习理论

强化学习理论框架趋于成熟

- 《Reinforcement Learning: An Introduction》, Richard S. Sutton and Andrew G. Barto, 1998. 截止目前已拓展四版。



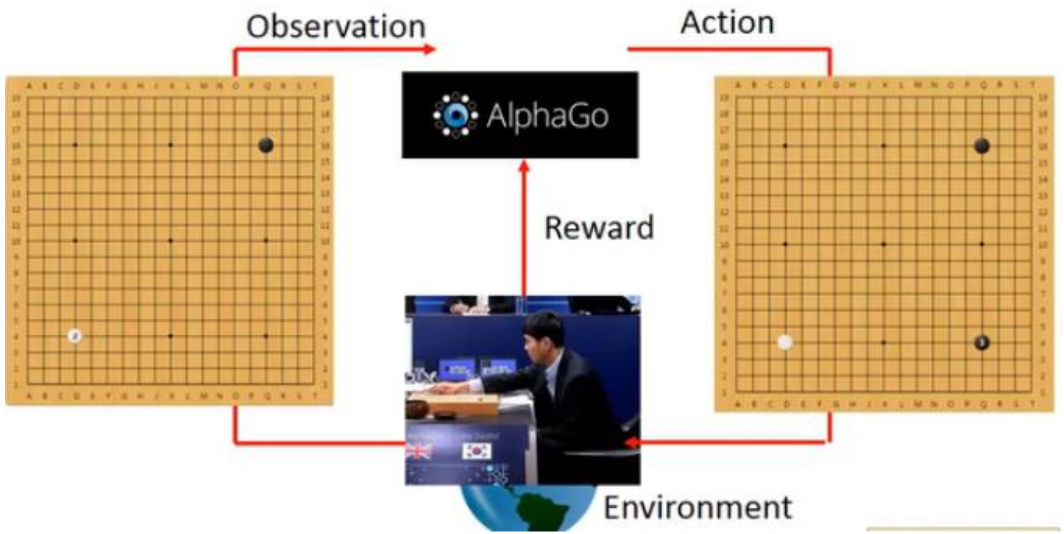


概述

现代强化学习理论

深度强化学习

- 2006 年以来，深度学习融入到强化学习体系中；
- 2016 年，AlphaGo； 2017 年，AlphaGo.



第一章 绪 论

1.1 概述

1.2 Markov 决策过程

1.3 强化学习

1.4 课程安排

1.5 小结

Markov 决策过程

Markov Process (MP)

- **MP 定义：**

A *Markov Process* (or *Markov Chain*) is a tuple $\langle \mathcal{S}, \mathcal{P} \rangle$

- $\mathcal{S}$ is a (finite) set of states
- $\mathcal{P}$ is a state transition probability matrix,
  $\mathcal{P}_{ss'} = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s\right]$

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

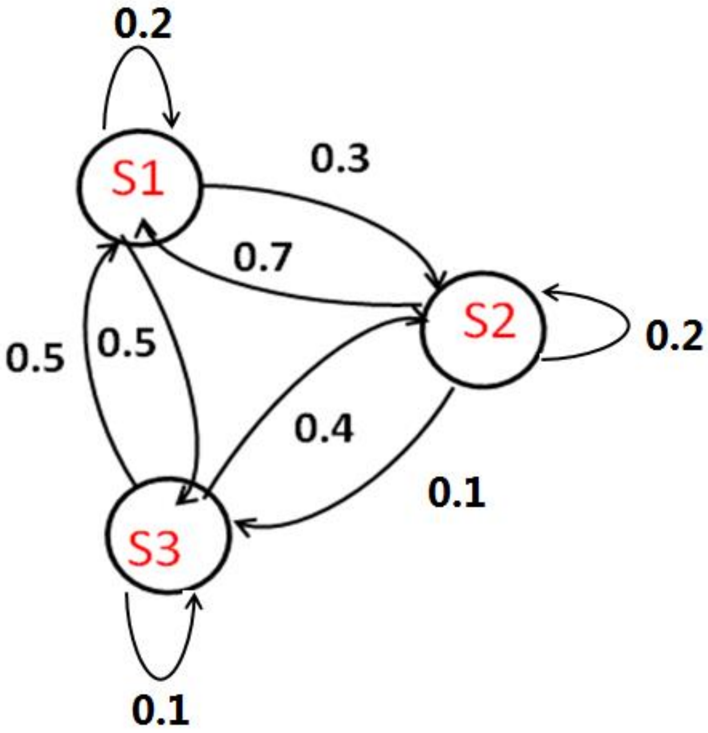
where each row of the matrix sums to 1

Markov Chain:

t +1 时刻状态的发生，只与 t 时刻有关

Markov 决策过程

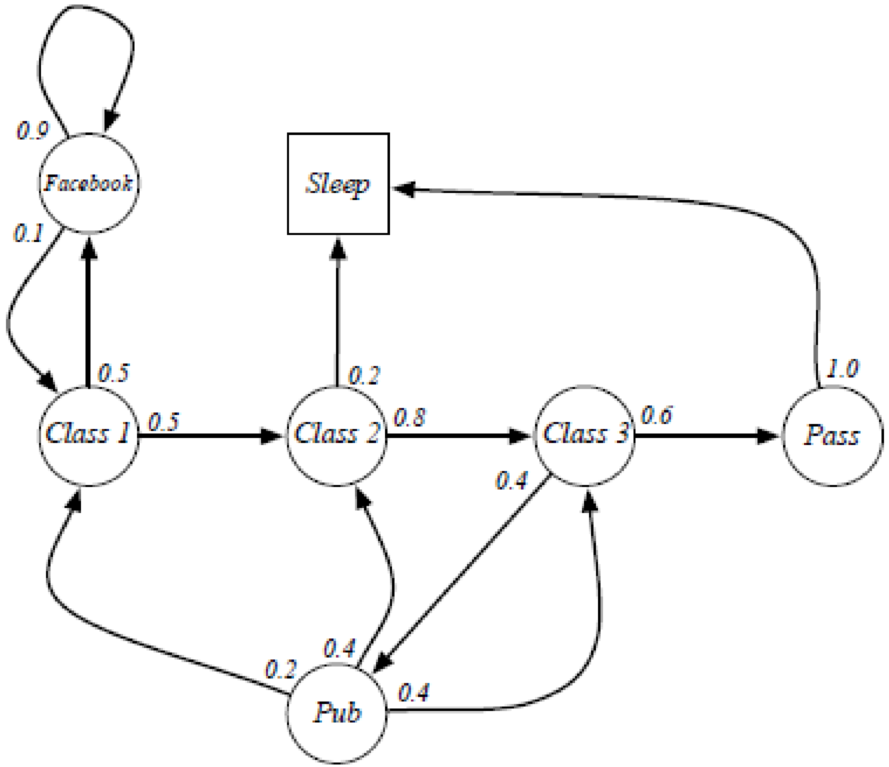
Markov Process (MP)



Markov 决策过程

Markov Process (MP)

- **例子：Student MP<S, P>**

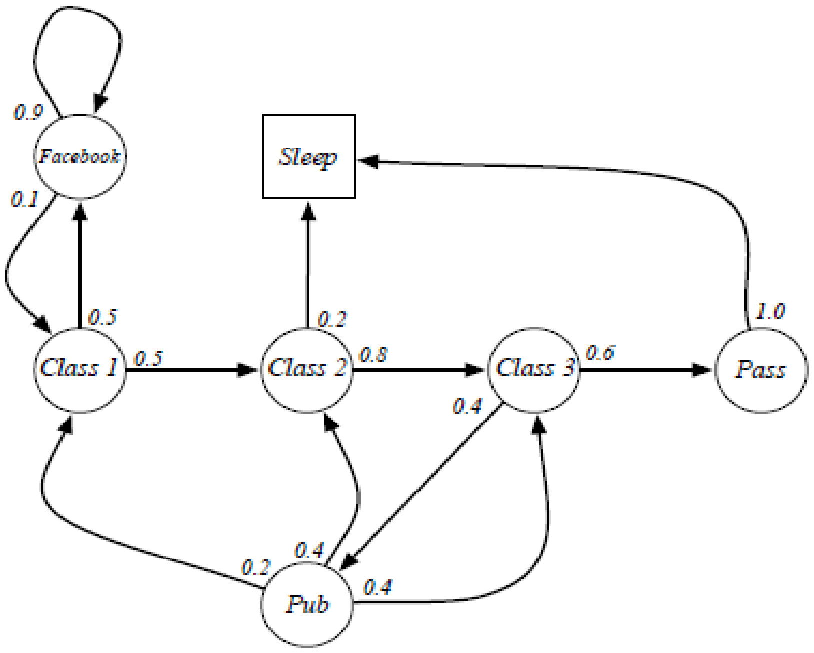


$$
\mathcal{P} = \begin{array}{cc}
 & \begin{array}{ccccccc} C1 & C2 & C3 & Pass & Pub & FB & Sleep \end{array} \\
\begin{array}{c} C1 \\ C2 \\ C3 \\ Pass \\ Pub \\ FB \\ Sleep \end{array} &
\begin{bmatrix}
 & 0.5 & & & & 0.5 & \\
 & & 0.8 & & & & 0.2 \\
 & & & 0.6 & 0.4 & & \\
 & & & & & & 1.0 \\
0.2 & 0.4 & 0.4 & & & & \\
0.1 & & & & & 0.9 & \\
 & & & & & & 1
\end{bmatrix}
\end{array}
$$

Markov 决策过程

Markov Process (MP)

- **Episodes ~ Student MP<S, P>**



Sample episodes for Student Markov Chain starting from $S_1 = C1$

$$S_1, S_2, ..., S_T$$

- C1 C2 C3 Pass Sleep
- C1 FB FB C1 C2 Sleep
- C1 C2 C3 Pub C2 C3 Pass Sleep
- C1 FB FB C1 C2 C3 Pub C1 FB FB FB C1 C2 C3 Pub C2 Sleep

Markov Reward Process (MRP)

- **MRP 定义：**

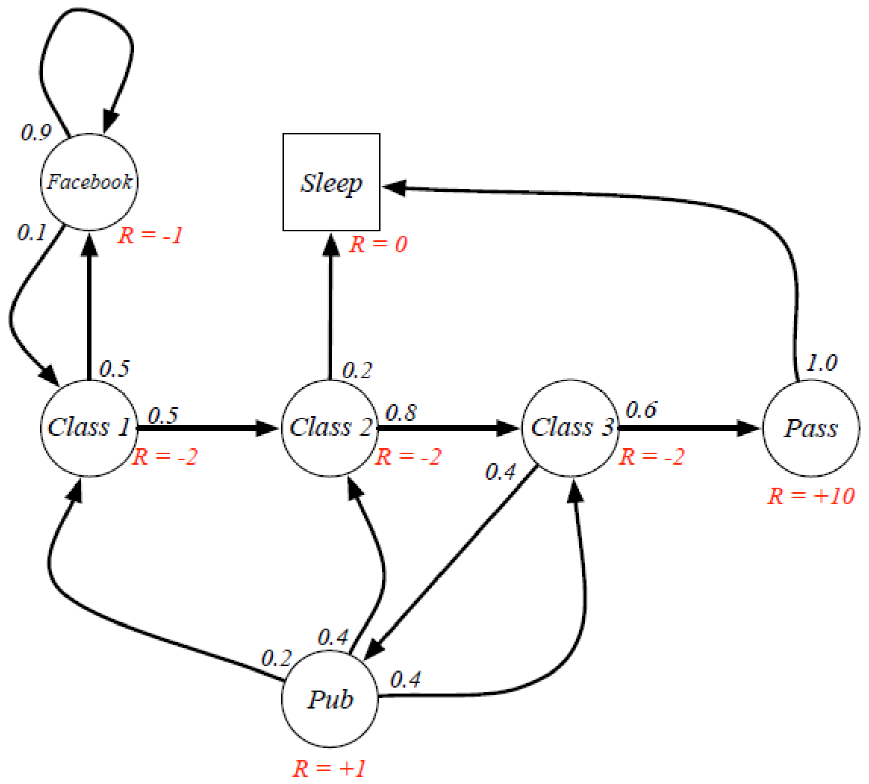
A *Markov Reward Process* is a tuple $\langle \mathcal{S}, \mathcal{P}, \mathcal{R}, \gamma \rangle$

- $\mathcal{S}$ is a finite set of states
- $\mathcal{P}$ is a state transition probability matrix,
  $\mathcal{P}_{ss'} = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s\right]$
- $\mathcal{R}$ is a reward function, $\mathcal{R}_s = \mathbb{E}\left[R_{t+1} \mid S_t = s\right]$
- γ is a discount factor, $\gamma \in [0, 1]$

Markov 决策过程

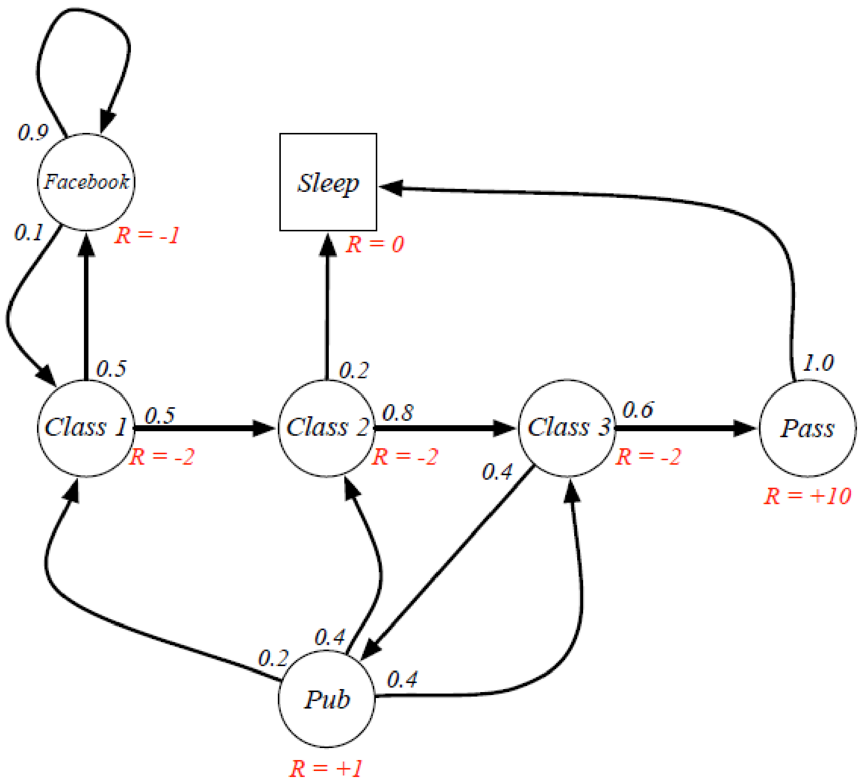
Markov Reward Process (MRP)

- **例子：Student MRP <S, P, R, γ >**



Markov Reward Process (MRP)

- **例子：Student MRP <S, P, R, γ>**



一个 Episode 的 Reward?

- C1 C2 C3 Pass Sleep
- C1 FB FB C1 C2 Sleep
- C1 C2 C3 Pub C2 C3 Pass Sleep
- C1 FB FB C1 C2 C3 Pub C1 FB FB FB C1 C2 C3 Pub C2 Sleep

Markov 决策过程

Markov Reward Process (MRP)

- **Return: discounted reward**

The *return* G_t is the total discounted reward from time-step t.

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

折扣因子：

1．区间［0， 1］意味着后续时刻的回报贡献越来越弱

2．如果 MP 过程无终态，那么G将无穷大

Markov Reward Process (MRP)

- **Return of Episode ~Student MRP <S, P, R, γ>**

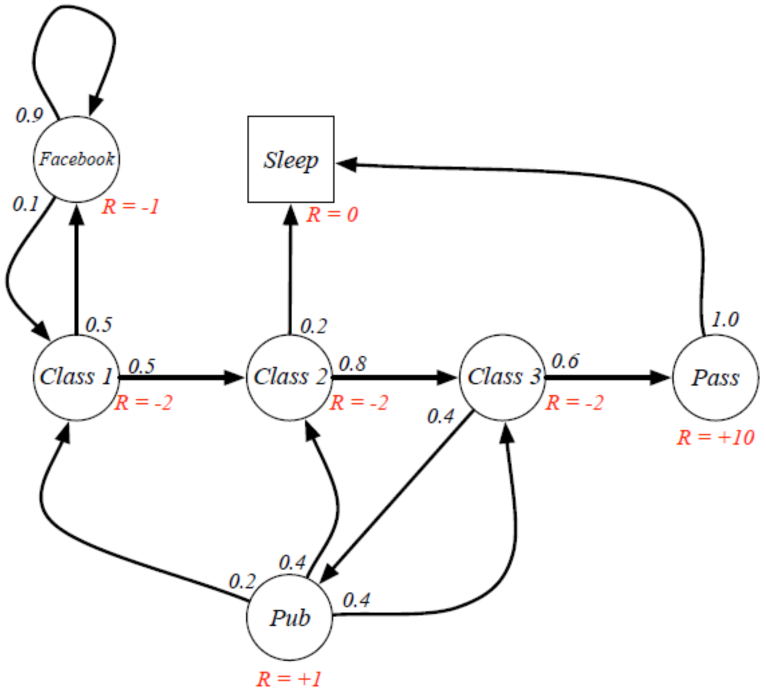
Sample returns for Student MRP:
Starting from $S_1 = C1$ with $\gamma = \frac{1}{2}$

$$G_1 = R_2 + \gamma R_3 + ... + \gamma^{T-2} R_T$$

C1 C2 C3 Pass Sleep	$v_1 = -2 - 2 * \frac{1}{2} - 2 * \frac{1}{4} + 10 * \frac{1}{8}$	$= -2.25$
C1 FB FB C1 C2 Sleep	$v_1 = -2 - 1 * \frac{1}{2} - 1 * \frac{1}{4} - 2 * \frac{1}{8} - 2 * \frac{1}{16}$	$= -3.125$
C1 C2 C3 Pub C2 C3 Pass Sleep	$v_1 = -2 - 2 * \frac{1}{2} - 2 * \frac{1}{4} + 1 * \frac{1}{8} - 2 * \frac{1}{16} \ldots$	$= -3.41$
C1 FB FB C1 C2 C3 Pub C1 ...	$v_1 = -2 - 1 * \frac{1}{2} - 1 * \frac{1}{4} - 2 * \frac{1}{8} - 2 * \frac{1}{16} \ldots$	$= -3.20$
FB FB FB C1 C2 C3 Pub C2 Sleep		

Markov Reward Process (MRP)

Value Function (Average Return)

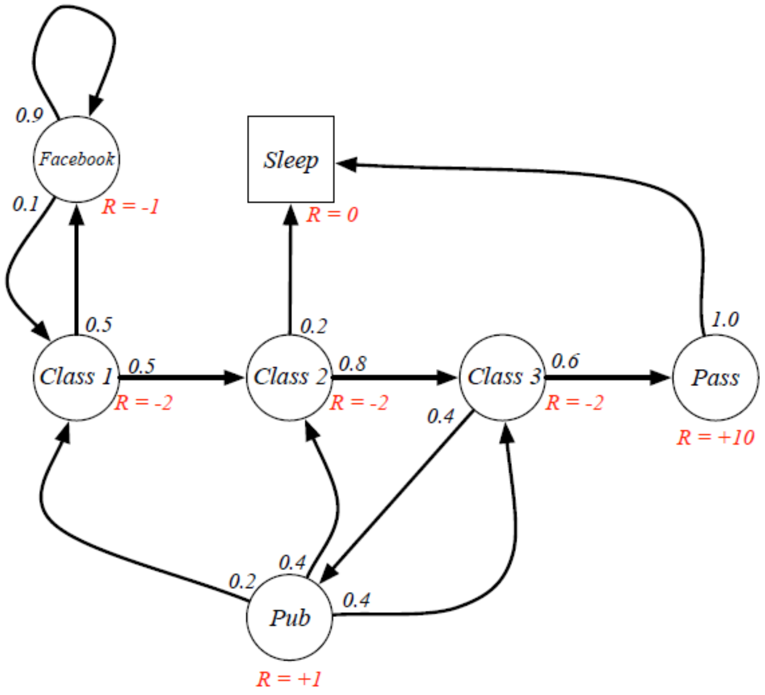


The *state value function* $v(s)$ of an MRP is the expected return starting from state s

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

Markov Reward Process (MRP)

Value Function (Average Return)



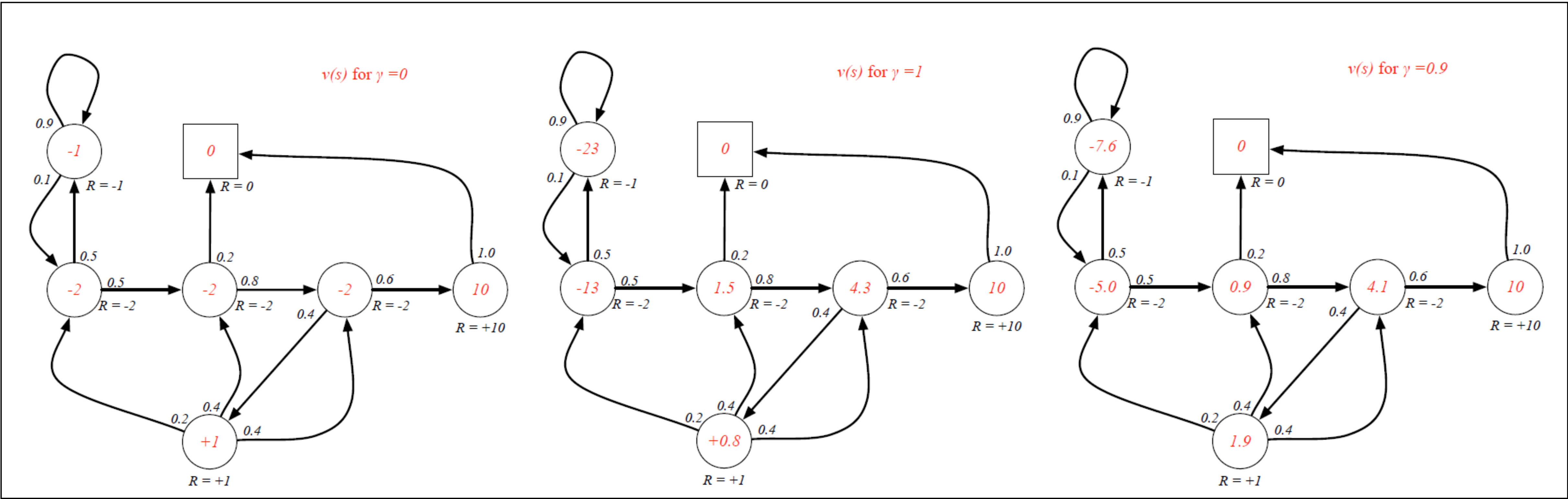
The *state value function* $v(s)$ of an MRP is the expected return starting from state s

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

当 MRP 确定，v(S)就应是确定的值

Markov Reward Process (MRP)

- **Value Function for Student MRP <S, P, R, γ >**



Markov 决策过程

Markov Reward Process (MRP)

该怎么求 v(S)?

Bellman Equation for MRP

一步迭代公式:

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

$$
v(s) = \mathcal{R}_s + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'} v(s')
$$

$$V(S) = \underset{t+1,t+2,\cdots\infty}{E}\left[G_t \mid S_t = S\right] = \sum_{t+1,\cdots\infty}\left(P(S_{t+1}\mid S_t=S)P(S_{t+2}\mid S_{t+1})\cdots\right)G_t$$

$$= \underset{t+1,\cdots\infty}{E}\left[R_{t+1} + rR_{t+2} + r^2 R_{t+3}\cdots \mid S_t = S\right]$$

$$= \underset{t+1,\cdots\infty}{E}\left[R_{t+1} \mid S_t = S\right] + \underset{t+1,\cdots\infty}{E}\left[rG(t+1) \mid S_t = S\right]$$

其中, $E_{t+1,\cdots\infty}\left[\cdot \mid S_t = S\right]$

$$= \sum_{t+1,\cdots\infty}\left(P(S_{t+1}\mid S_t=S)P(S_{t+2}\mid S_{t+1})P(S_{t+3}\mid S_{t+2})\cdots\right)$$

$V(S)$ $E_{t+1,\cdots\infty}\left[R_{t+1} \mid S_t = S\right]$

$$= \sum_{t+1,\infty}\left(\quad \right) R_{t+1},$$

$$= \sum_{t+1} P(S_{t+1} \mid S_t = S) R_{t+1}$$

R_{t+1} 只与 $P(S_{t+1}\mid S_t)$ 有关

t $t+1$ $t+2$

$$r \underset{t+1,\cdots\infty}{E}\left(G_{t+1} \mid S_t = S\right)$$

$$= r\sum_{t+1,\cdots\infty}\left(\quad \right) G_{t+1}$$

$$= r\sum_{t+1,\cdots\infty}\left(P(S_{t+1}\mid S_t=S)P(S_{t+2}\mid S_{t+1})\cdots\right) G_{t+1}$$

$$= r\sum_{t+1} P(S_{t+1}\mid S_t=S) \underbrace{\sum_{t+2,\cdots\infty}\left[P(S_{t+2}\mid S_{t+1})\cdots\right] G_{t+1}}$$

$$= r\sum_{t+1} P(S_{t+1}\mid S_t=S)\, V(S_{t+1})$$

$$V(S) = \sum_{t+1} P(S_{t+1}\mid S_t=S) R_{t+1} + r\sum_{t+1} P(S_{t+1}\mid S_t=S)\, V(S_{t+1})$$

$$V(S) = \underset{t+1}{E}\left[R_{t+1} + rV(S_{t+1}) \mid S_t = S\right]$$

Markov 决策过程

Markov Reward Process (MRP)

该怎么求 v(S)?

- **Bellman Equation for MRP**

$$v(s) = \mathcal{R}_s + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'} v(s') \quad \Rightarrow$$

$$\begin{bmatrix} v(1) \\ \vdots \\ v(n) \end{bmatrix} = \begin{bmatrix} \mathcal{R}_1 \\ \vdots \\ \mathcal{R}_n \end{bmatrix} + \gamma \begin{bmatrix} \mathcal{P}_{11} & \dots & \mathcal{P}_{1n} \\ \vdots & & \\ \mathcal{P}_{11} & \dots & \mathcal{P}_{nn} \end{bmatrix} \begin{bmatrix} v(1) \\ \vdots \\ v(n) \end{bmatrix}$$

$$v = \mathcal{R} + \gamma \mathcal{P} v \quad \Rightarrow \quad v = (I - \gamma \mathcal{P})^{-1} \mathcal{R}$$

当 MRP（环境参数）已知时，理论上通过求解 Bellman 方程式可以计算 v(S)

Markov Decision Process (MDP)

MDP 定义：

A *Markov Decision Process* is a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$

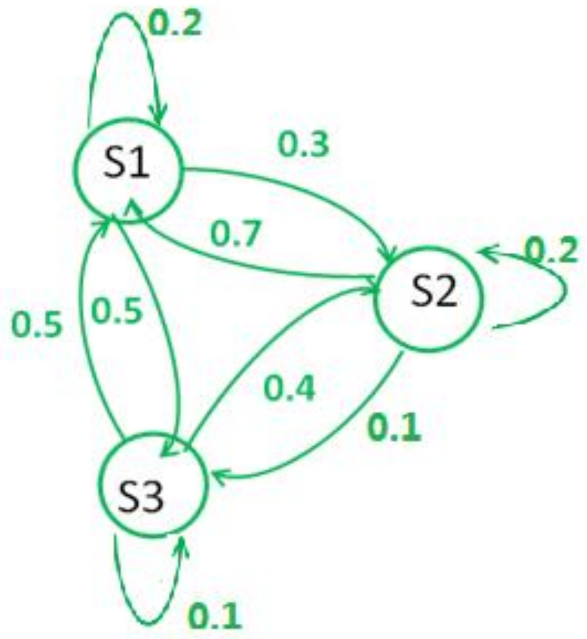
- $\mathcal{S}$ is a finite set of states
- $\mathcal{A}$ is a finite set of actions
- $\mathcal{P}$ is a state transition probability matrix, $\mathcal{P}_{ss'}^{a} = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s, A_t = a\right]$
- $\mathcal{R}$ is a reward function, $\mathcal{R}_s^a = \mathbb{E}\left[R_{t+1} \mid S_t = s, A_t = a\right]$
- γ is a discount factor $\gamma \in [0, 1]$.

形象的解释：MDP 是一个多层的 MRP，每一层对应一个行动 *a*.

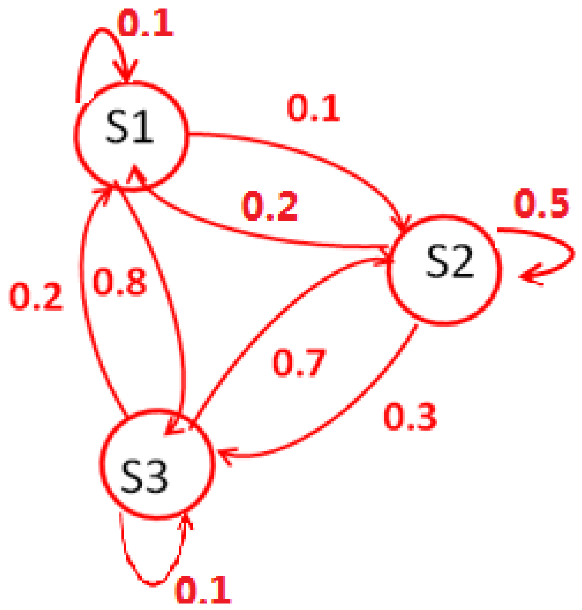
Markov 决策过程

Markov Decision Process (MDP)

a_1



a_2



Markov Decision Process (MDP)

- **如何 Episodes ~MDP?**

对于状态 S，需要知道如何选择行动 a？

- **Policy**

A *policy* π is a distribution over actions given states

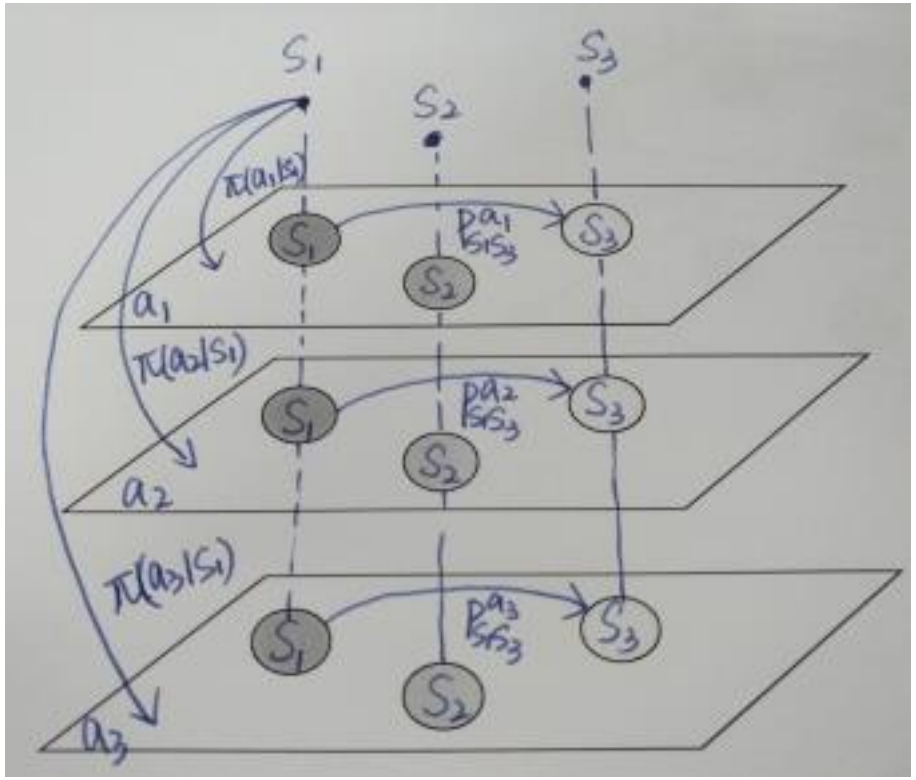
$$\pi(a|s) = \mathbb{P}\left[A_t = a \mid S_t = s\right]$$

i.e. Policies are *stationary* (time-independent),
$A_t \sim \pi(\cdot|S_t), \forall t > 0$

Markov 决策过程

Markov Decision Process (MDP)

Given an MDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ and a policy π



Episodes:

S_1, a_1, R_2, S_2, a_2, R_3, S_3, a_3, R_4, … …

Markov 决策过程

Markov Decision Process (MDP)

The state sequence $S_1, S_2, \ldots$

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

以图为例：

$$P_{s_1 s_3} = \pi(a_1|s_1) P^{a_1}_{s_1 s_3} + \pi(a_2|s_1) P^{a_2}_{s_1 s_3} + \pi(a_3|s_1) P^{a_3}_{s_1 s_3}$$

Markov 决策过程

Markov Decision Process (MDP)

The state sequence $S_1, S_2, \ldots$

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

以图为例:

$$P_{s_1 s_3} = \pi(a_1|s_1)P^{a_1}_{s_1 s_3} + \pi(a_2|s_1)P^{a_2}_{s_1 s_3} + \pi(a_3|s_1)P^{a_3}_{s_1 s_3}$$

$\sum_{s'} P_{s_1 s'} = ?$ 是否为1

Markov 决策过程

Markov Decision Process (MDP)

$$P_{s_1 s_3} = \pi(a_1|s_1) P^{a_1}_{s_1 s_3} + \pi(a_2|s_1) P^{a_2}_{s_1 s_3} + \pi(a_3|s_1) P^{a_3}_{s_1 s_3}$$

$$P_{s_1 s_2} = \pi(a_1|s_1) P^{a_1}_{s_1 s_2} + \pi(a_2|s_1) P^{a_2}_{s_1 s_2} + \pi(a_3|s_1) P^{a_3}_{s_1 s_2}$$

$$P_{s_1 s_1} = \pi(a_1|s_1) P^{a_1}_{s_1 s_1} + \pi(a_2|s_1) P^{a_2}_{s_1 s_1} + \pi(a_3|s_1) P^{a_3}_{s_1 s_1}$$

$$\sum \cdots = \pi(a_1|s_1)\cdot 1 + \pi(a_2|s_1)\cdot 1 + \pi(a_3|s_1)\cdot 1$$

$$= 1$$

Markov 决策过程

Markov Decision Process (MDP)

The state sequence $S_1, S_2, \ldots$ is a Markov process $\langle \mathcal{S}, \mathcal{P}^{\pi} \rangle$

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

MDP by a policy is a MP !

Markov Decision Process (MDP)

The state and reward sequence $S_1, R_2, S_2, \ldots$ is a Markov reward process $\langle \mathcal{S}, \mathcal{P}^{\pi}, \mathcal{R}^{\pi}, \gamma \rangle$

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

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

MDP by a policy is also a MRP !

Have a break !

Markov 决策过程

Markov Decision Process (MDP)

Value Function

The *state-value function* $v_\pi(s)$ of an MDP is the expected return starting from state s, and then following policy π

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

The *action-value function* $q_\pi(s, a)$ is the expected return starting from state s, taking action a, and then following policy π

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

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

v(s)是由 q(s,a)和策略（行动）决定。

Markov Decision Process (MDP)

- **Bellman Expectation Equation**
  - **State Value Function （V 值方程）**
  - **Action-Value Function （Q 值方程）**

Markov 决策过程

Bellman Expectation Equation for MDP

State Value Function

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

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

Bellman Equation for MRP:

$$V(S) = \mathop{E}_{t+1}\left[R_{t+1} + \gamma V(S_{t+1}) \mid S_t = s\right]$$

$$= \sum_{S_{t+1}} P(S_{t+1} \mid S_t = s)\left[R_{t+1} + \gamma V(S_{t+1})\right]$$

$\boxed{S_{t+1} = s'}$

$$= \sum_{s'} P_{ss'}\left[R_{t+1} + \gamma V(s')\right]$$

MDP is a MRP:

$$\begin{cases} P_{ss'}^{\pi} = \sum_{a} \pi(a|s) P_{ss'}^{a} \\ V_\pi(s) = \sum_{s'} P_{ss'}^{\pi}\left[R_{t+1} + r V_\pi(s')\right] \end{cases}$$

$$V_\pi(s) = \sum_{s'}\sum_{a} \pi(a|s) P_{ss'}^{a}\left[R_{t+1} + r V_\pi(s')\right]$$

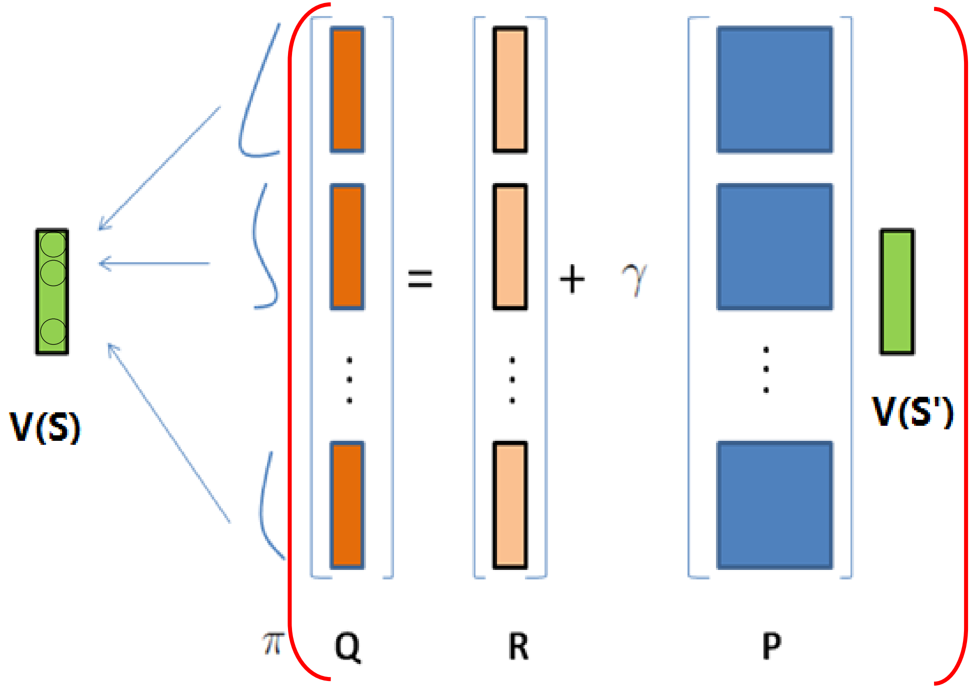
$$= \sum_{a} \pi(a|s) \sum_{s'} P_{ss'}^{a}\left[R_{t+1} + r V_\pi(s')\right]$$

$$= \sum_{a} \pi(a|s)\left(R_s^a + \gamma \sum_{s'} P_{ss'}^{a} V_\pi(s')\right)$$

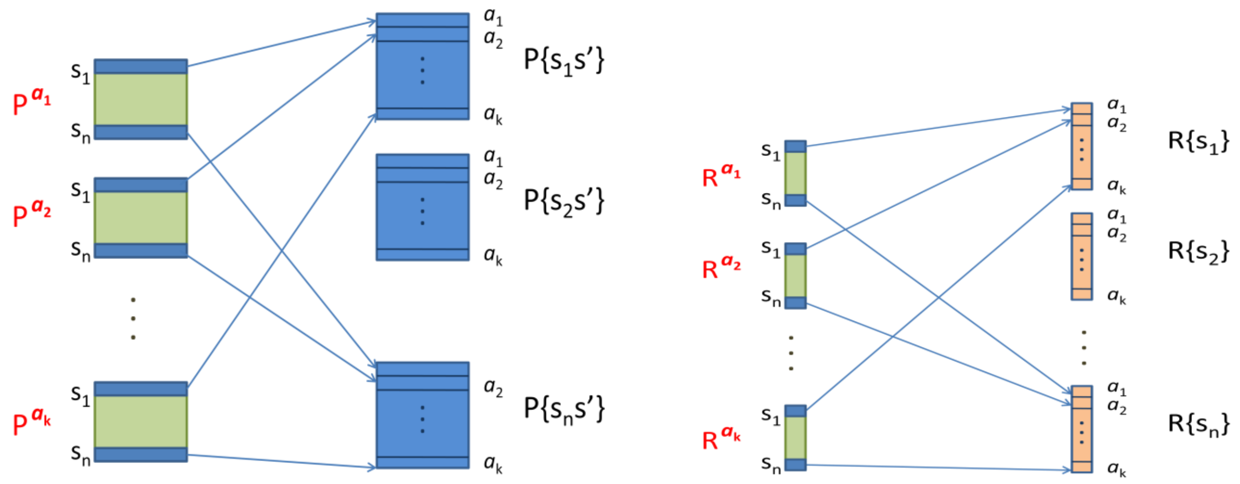
Bellman Expectation Equation for MDP

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

每一行 block 运算对应一个状态



其中，

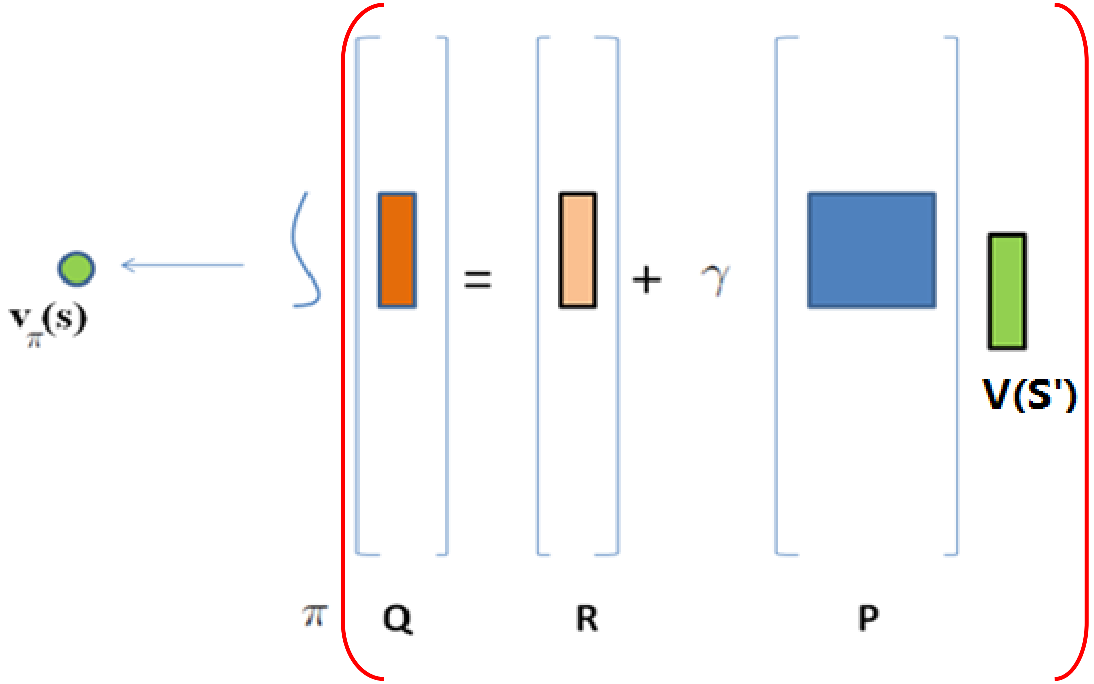


Markov 决策过程

Bellman Expectation Equation for MDP

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

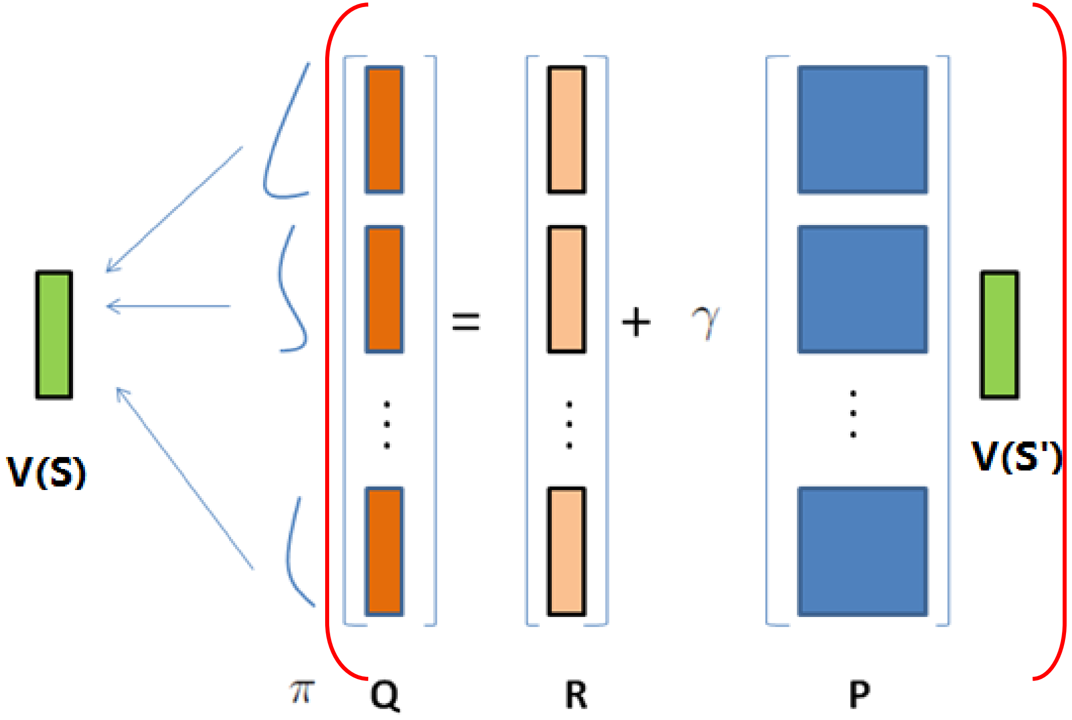
Each s in S



Bellman Expectation Equation for MDP

All s in S

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



Bellman Expectation Equation for MDP

MDP by Policy is A MRP

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

$$\Rightarrow \quad \mathcal{P}_{s,s'}^\pi = \sum_{a\in\mathcal{A}} \pi(a|s)\mathcal{P}_{ss'}^a$$

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

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

（ MRP Process ）

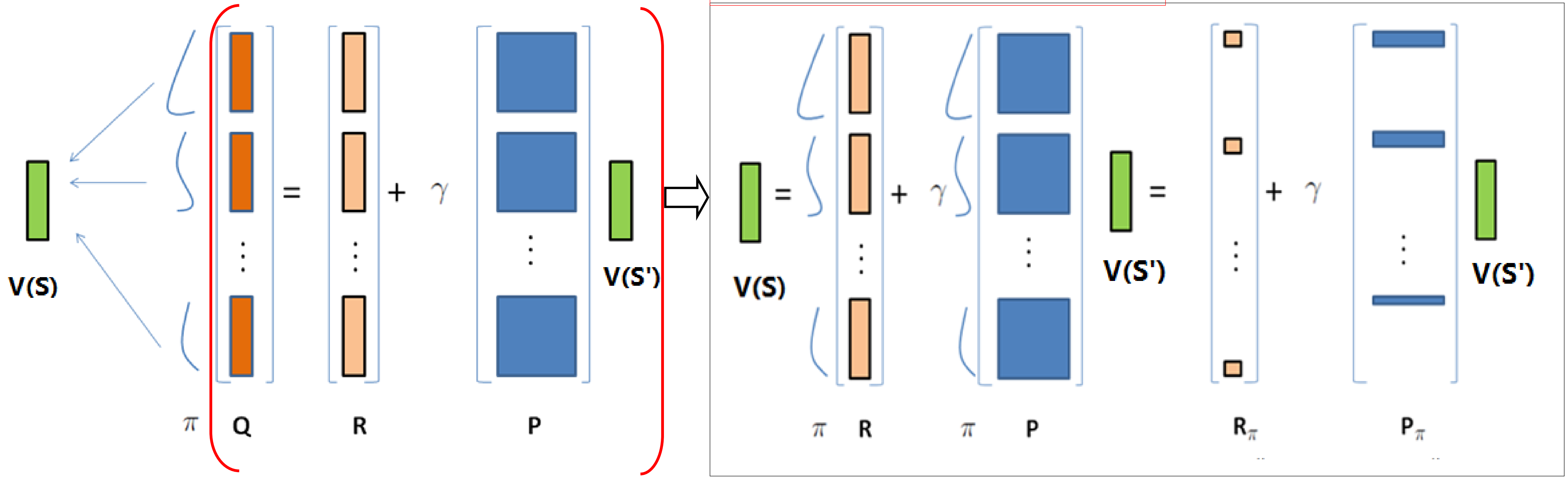
Markov 决策过程

Bellman Expectation Equation for MDP

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

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

（ MRP Process ）

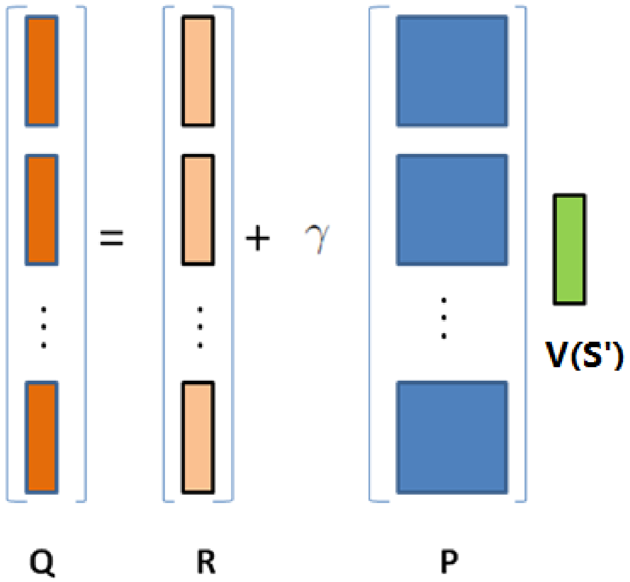


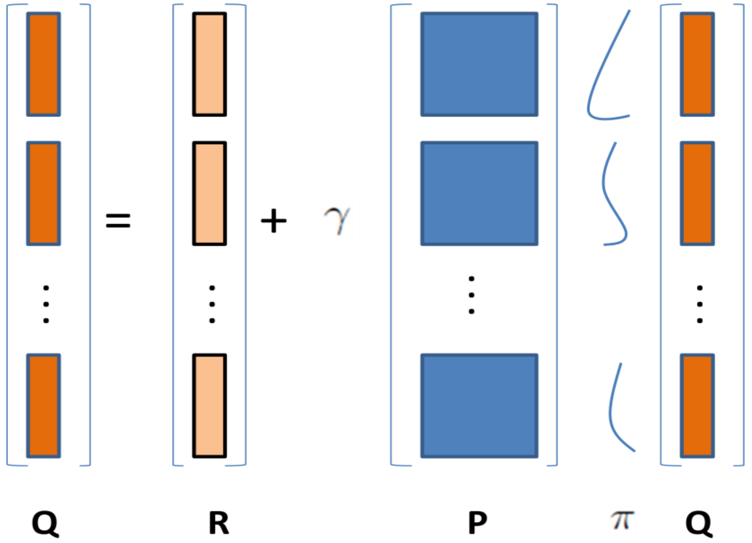
Markov 决策过程

Bellman Expectation Equation for MDP

- **Action-Value Function**

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





Bellman Expectation Equation for MDP

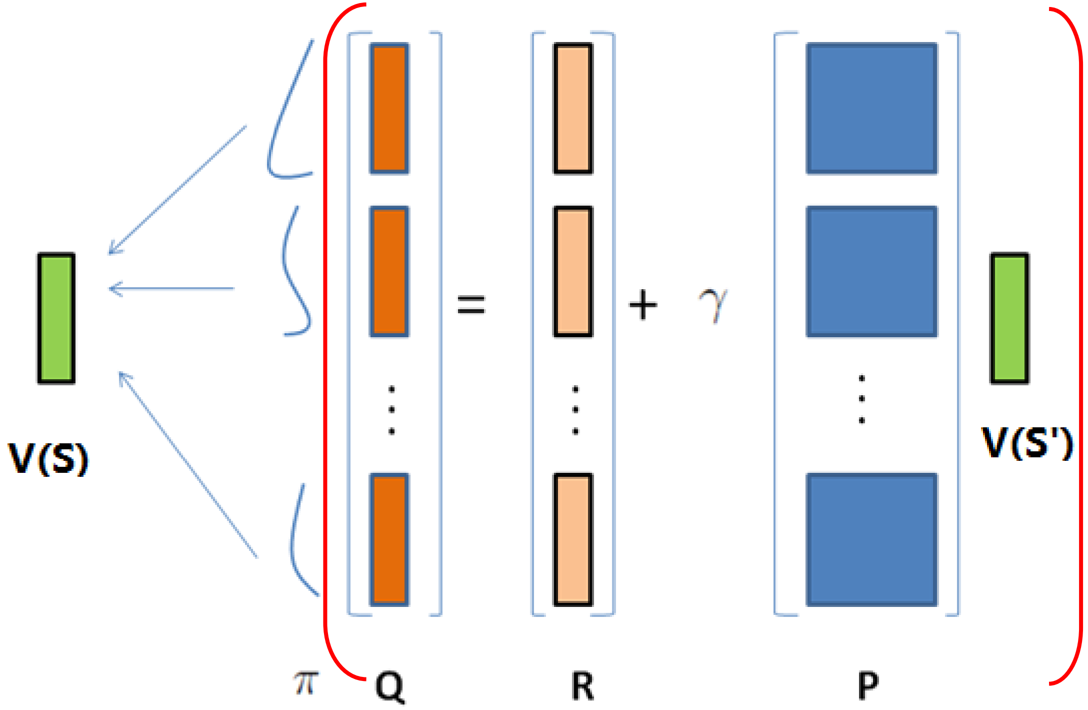
- **Bellman Expectation Equations**

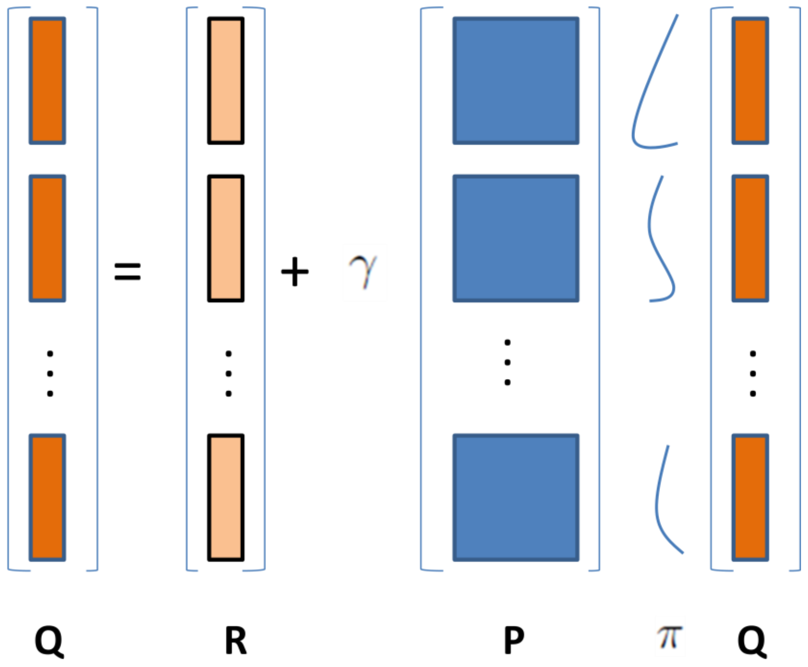
$$v_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) \left(\mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_\pi(s') \right)$$

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

Markov 决策过程

公式的矩阵图形：





Markov 决策过程

Have a break !

第一章 绪 论

1.1 概述

1.2 Markov 决策过程

1.3 强化学习

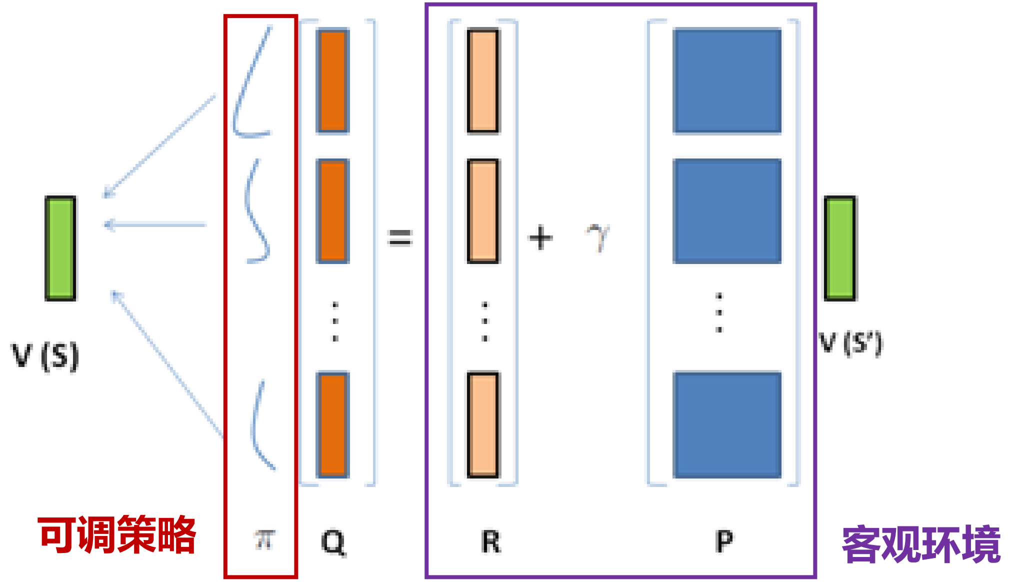
1.4 课程安排

1.5 小结

问题描述

- 怎样的选择 *a* | S_t，可以使得 Average Return （Value Function ）最大?

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



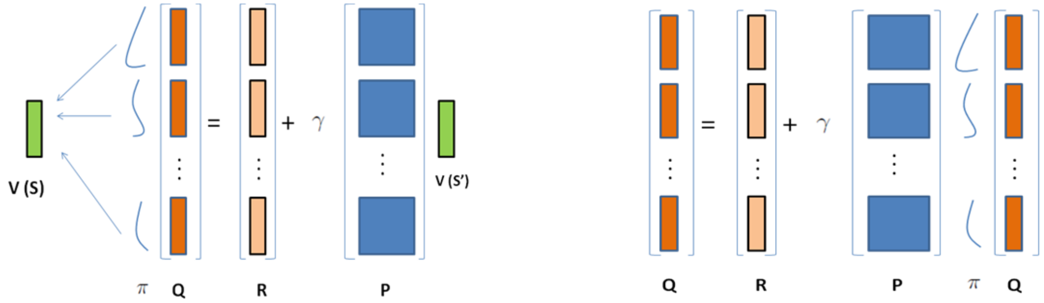
Optimal Value Function

The *optimal state-value function* $v_*(s)$ is the maximum value function over all policies

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

The *optimal action-value function* $q_*(s, a)$ is the maximum action-value function over all policies

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



Optimal Policy

- **策略的优越性评价：**

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

Theorem

For any Markov Decision Process

- *There exists an optimal policy* π_* *that is better than or equal to all other policies,* $\pi_* \geq \pi, \forall \pi$
- *All optimal policies achieve the optimal value function,* $v_{\pi_*}(s) = v_*(s)$
- *All optimal policies achieve the optimal action-value function,* $q_{\pi_*}(s,a) = q_*(s,a)$

Optimal Policy

Deterministic Optimal Policy

An optimal policy can be found by maximising over $q_*(s, a)$,

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

Bellman Optimality Equation

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

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

- **Bellman Optimality Equation for v***

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

- **Bellman Optimality Equation for Q***

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

Solving Bellman Optimality Equation

- Bellman Optimality Equation is non-linear
- No closed form solution (in general)
- Many iterative solution methods
  - Value Iteration
  - Policy Iteration
  - Q-learning
  - Sarsa

第一章 绪 论

1.1 Markov 决策过程

1.2 强化学习

1.3 课程安排

1.4 小结

课程安排

课程内容

1 绪论：强化学习问题

2 动态规划方法 求解Bellman方程，动态规划方法：值估计和策略控制

3 代价值估计

4 策略控制 随机方法

5 值函数逼近

6 策略梯度方法 函数拟合方法

7 模型方法

8 蒙特卡洛树搜索 环境模拟和探测

9 强化学习应用案例： AlphaGo， AlphaGo Zero， NLP 任务，视觉导航 等

1. 第一章 绪 论

1.1 Markov 决策过程

1.2 强化学习

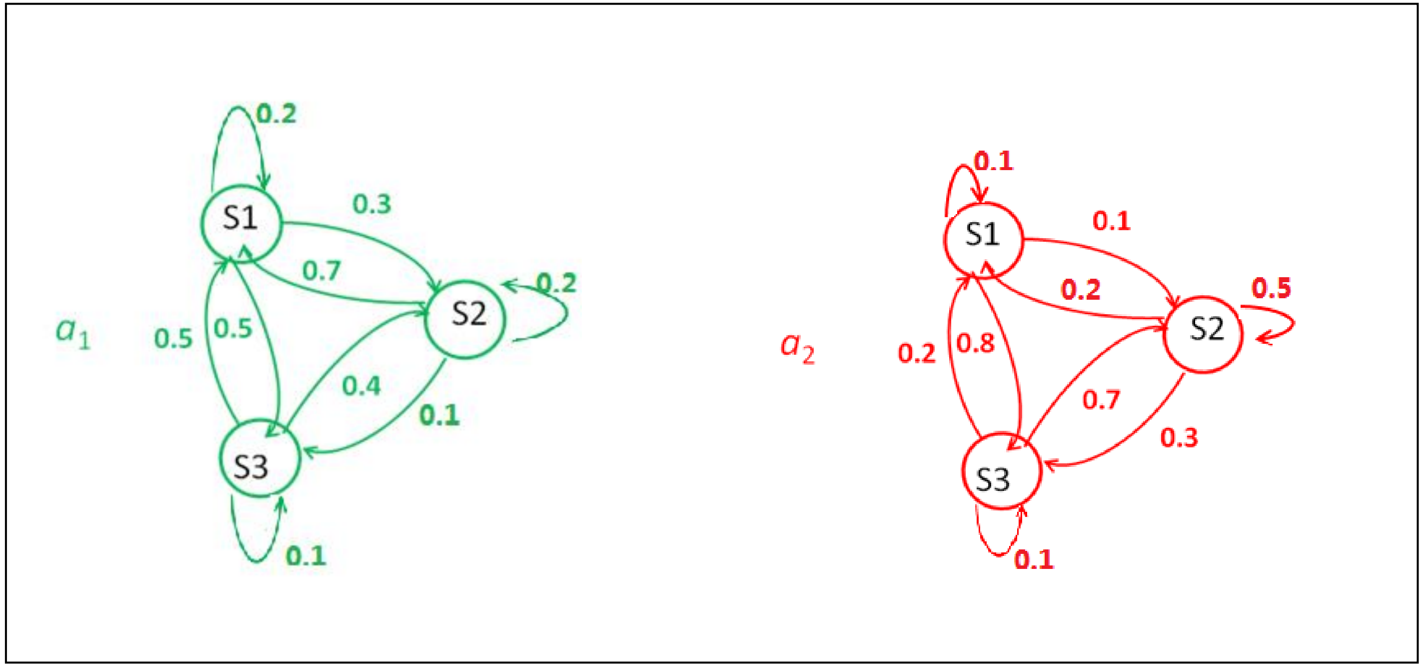
1.3 课程安排

1.4 小结

小结

1. MDP

多个行动 a 的 MRP



小结

2. **Average Return（Value Function)**

MRP：

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

$$
v(s) = \mathcal{R}_s + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'} v(s')
$$

MDP：

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

小结

3. **强化学习问题**

怎样的选择 $a | S_t$，可以使得 Average Return（Value Function)最大?

An optimal policy can be found by maximising over $q_*(s, a)$,

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

小结

4. 强化学习本质是求解 Bellman Optimality Equation

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

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

Bellman Optimality Equation for v*

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

Bellman Optimality Equation for Q*

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

本讲参考文献

1. Richard S. Sutton and Andrew G. Barto. Reinforcement Learning: An Introduction. (Second edition)
2. David Silver , Slides@ 《Reinforcement Learning:An Introduction》,2016.