中国科学技术大学
University of Science and Technology of China

《人工智能数学原理与算法》
第 7 章：强化学习

7.2 马尔可夫决策过程

吉建民
jianmin@ustc.edu.cn

目录

马尔可夫决策过程简介

- **马尔可夫决策过程（Markov Decision Process，MDP）**严格的描述了强化学习所面临的环境
  - 在 MDP 中，环境是完全可观测的：当前状态能够完整刻画系统的运行过程
  - 几乎所有强化学习问题都可以被表示为 MDP，例如：
    - 最优控制问题，可以看作处理连续状态和行动的 MDP 问题
    - 部分可观察（Partially Observable）问题，可以转换为面向信念状态（Belief State）的 MDP 问题
    - 赌博机（Bandit）问题，可以看作只有一个状态的 MDP 问题
    - 多智能体随机博弈（Stochastic Game）问题，可以近似为智能体与自然博弈的 MDP 问题

目录

随机过程

- **随机过程（Stochastic Process）**：随时间演化的一连串随机事件，每个时间点都对应一个可能出现多种取值的随机变量
  - 概率论的研究对象是静态的随机现象，而随机过程的研究对象是随时间演变的随机现象
    - 例如，天气随时间的变化、城市交通随时间的变化
  - 在随机过程中，随机现象在某时刻 t 的取值是一个随机变量，用 $\boldsymbol{S_t}$ 表示
  - 随机过程可以表示为：$\{S_t : t \in T\}$
  - 已知历史信息 $(S_1, \ldots, S_t)$ 时，下一个时刻状态为 S_{t+1} 的概率为

$$\mathrm{P}(S_{t+1} | S_1, \ldots, S_t)$$

马尔可夫性质

- 一个随机过程被称为具有**马尔可夫性质（Markov Property）**，当且仅当任意时刻的状态只取决于上一时刻的状态，用公式表示为

$$P(S_{t+1}|S_t) = \mathrm{P}(\mathrm{S}_{t+1}|\mathrm{S}_1, \ldots, \mathrm{S}_t)$$

  - 当前状态是未来的充分统计量，即下一个状态只取决于当前状态，而不会受到过去状态的影响
  - 马尔可夫性可以简化运算，只要当前状态可知，所有的历史信息都不再需要了，利用当前状态信息就可以决定未来

状态转移矩阵

- 对一个马尔科夫状态 $\boldsymbol{s}$ 及其后继状态 $\boldsymbol{s'}$，状态转移概率 (State Transition Probability) 定义为 $P_{ss'} = P(S_{t+1} = s' | S_t = s)$
- **状态转移矩阵 (State Transition Matrix)** $\mathcal{P}$ 则定义了从所有状态 $\boldsymbol{s}$ 到其所有后继状态 $\boldsymbol{s'}$ 的转移概率，即

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

其中矩阵的每一行元素之和都等于 1

马尔可夫过程

- 马尔可夫过程是具有马尔可夫性质的随机过程

定义

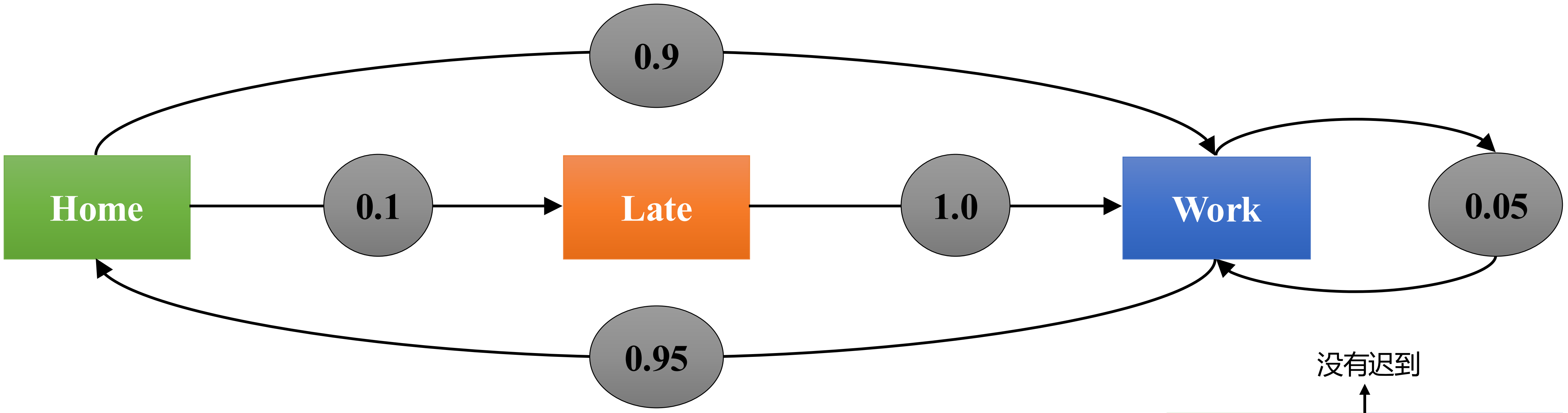
一个**马尔可夫过程（Markov Process）**或**马尔可夫链（Markov Chain）**

可以表示为一个二元组 $\langle \mathcal{S}, \mathcal{P} \rangle$，其中：

- $\mathcal{S}$ 是状态的（有限）集合
- $\mathcal{P}$ 是状态转移矩阵，$P_{ss'} = P(S_{t+1} = s' | S_t = s)$

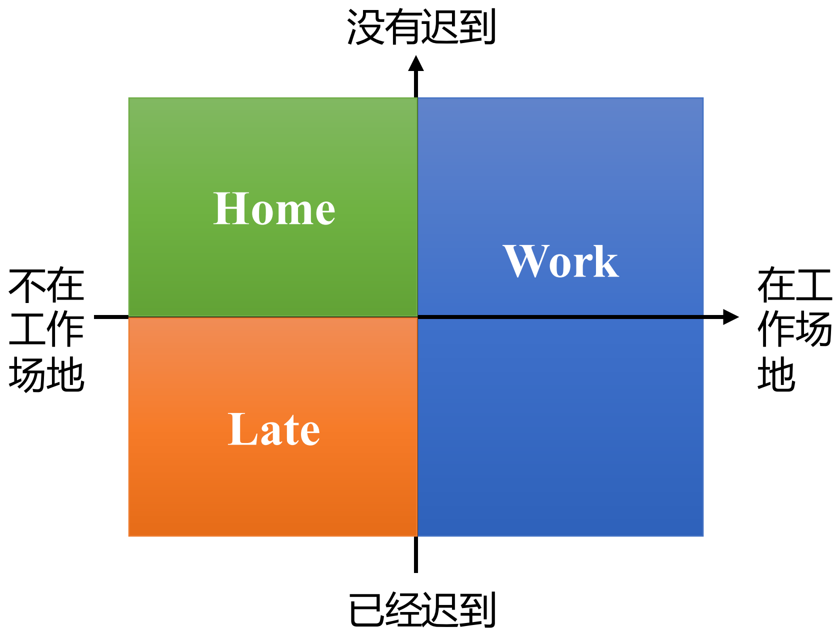
示例：通勤问题

- 通勤问题中三个状态，Home, Late, Work，构成马尔可夫链



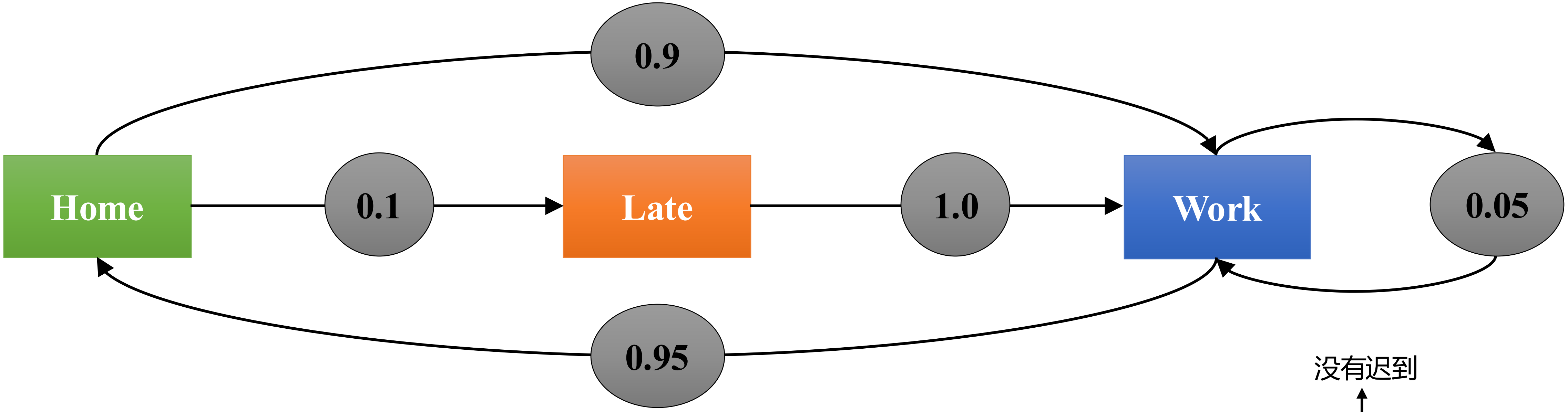
针对 (Home, Late, Work) 的状态转移矩阵为：

$$P_{ss'} = \begin{bmatrix} 0.0 & 0.1 & 0.9 \\ 0.0 & 0.0 & 1.0 \\ 0.95 & 0.0 & 0.05 \end{bmatrix}$$

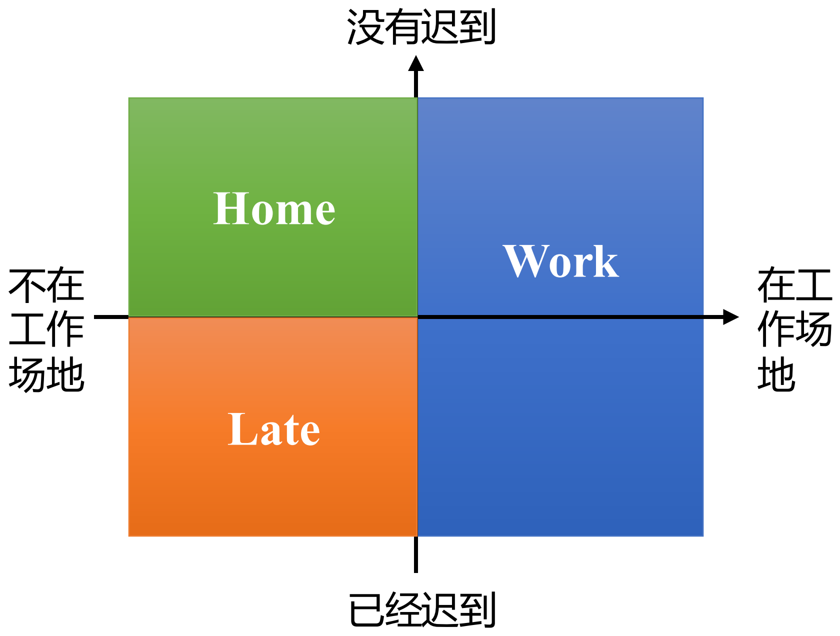


示例：通勤问题

- 通勤问题中三个状态，Home, Late, Work，构成马尔可夫链



- 初始状态为 Home 的马尔可夫链样本序列示例：
  - Home, Late, Work, Home, …
  - Home, Late, Work, Work, …
  - Home, Work, Work, Home, …
  - Home, Work, Work, Work, …
  - Home, Work, Home, Late, …
  - Home, Work, Home, Work, …



目录

马尔可夫奖励过程

- 马尔可夫奖励过程是在马尔可夫链基础上增加了"价值（奖励）"

定义

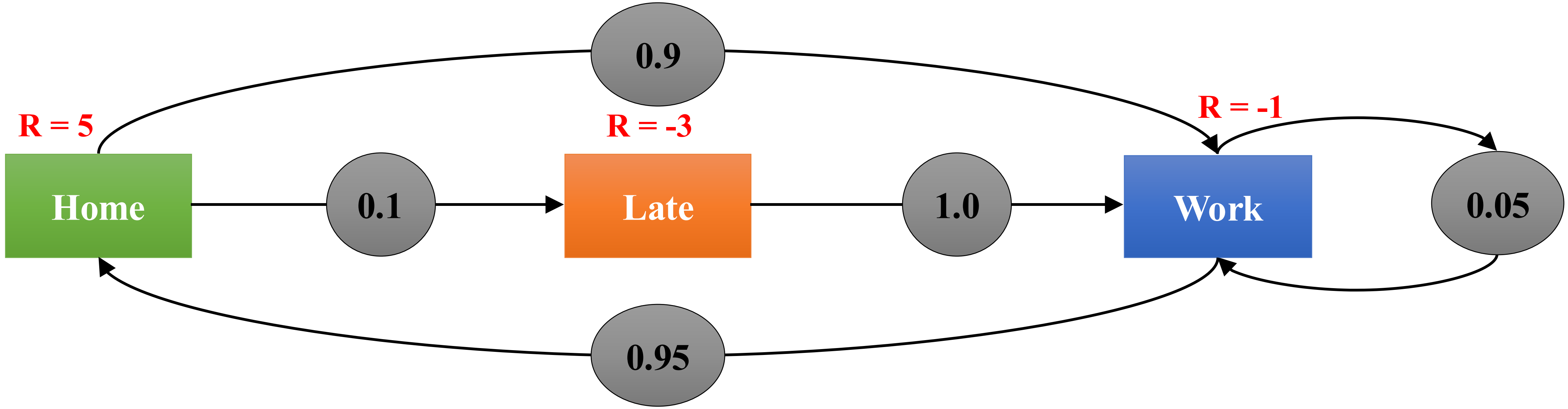
一个**马尔可夫奖励过程（Markov Reward Process, MRP）**可以表示为一个四元组 $\langle \mathcal{S}, \mathcal{P}, \mathcal{R}, \gamma \rangle$，其中：

- $\mathcal{S}$ 是状态的（有限）集合
- $\mathcal{P}$ 是状态转移矩阵，$P_{ss'} = P(S_{t+1} = s' | S_t = s)$
- $\mathcal{R}$ 是奖励函数，$R_s = E[R_{t+1} | S_t = s]$
- γ 是折扣因子，$\gamma \in [0, 1]$

- 引入折扣因子是因为远期收益具有一定不确定性，有时我们更希望能够尽快获得一些奖励，所以我们需要对远期收益打一些折扣
  - 接近 1 的 γ 更关注长期的累计奖励，接近 0 的 γ 更考虑短期奖励

示例：通勤问题

- 通勤问题中三个状态，Home, Late, Work，构成马尔可夫链
- 通勤问题扩展为马尔可夫奖励过程
  - R(Home) = 5，R(Late) = -3，R(Work) = -1



回报

定义

在一个马尔可夫奖励过程（MRP）中，在时间步 *t* 开始计算的**回报（Return）** G_t 指的是从时间步 *t* 开始累积的折扣奖励总和：

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

其中，折扣因子 $\gamma \in [0,1]$，R_t 表示在时刻 *t* 获得的奖励。

在通勤问题中，当 $\gamma = 0.5$，一个马尔可夫链样本序列 Home, Late, Work, Home 的回报为：

$$G_1 = 5 + 0.5 \times (-3) + 0.5^2 \times (-1) + 0.5^3 \times 5 = 5 - 1.5 \ - 0.25 + 0.625 = 3.875$$

此序列发生的概率为（初始状态为 Home）：

$$P_1 = P(Home) \times P(Late|Home) \times P(Work|Late) \times P(Home|Work)$$
$$= 1 \times 0.1 \times 1 \times 0.95 = 0.095$$

价值函数

定义

在一个马尔可夫奖励过程（MRP）中，**状态价值函数（State Value Function）** $v(s)$ 定义为从状态 s 开始时的期望回报：

$$v(s) = E[G_t | S_t = s]$$

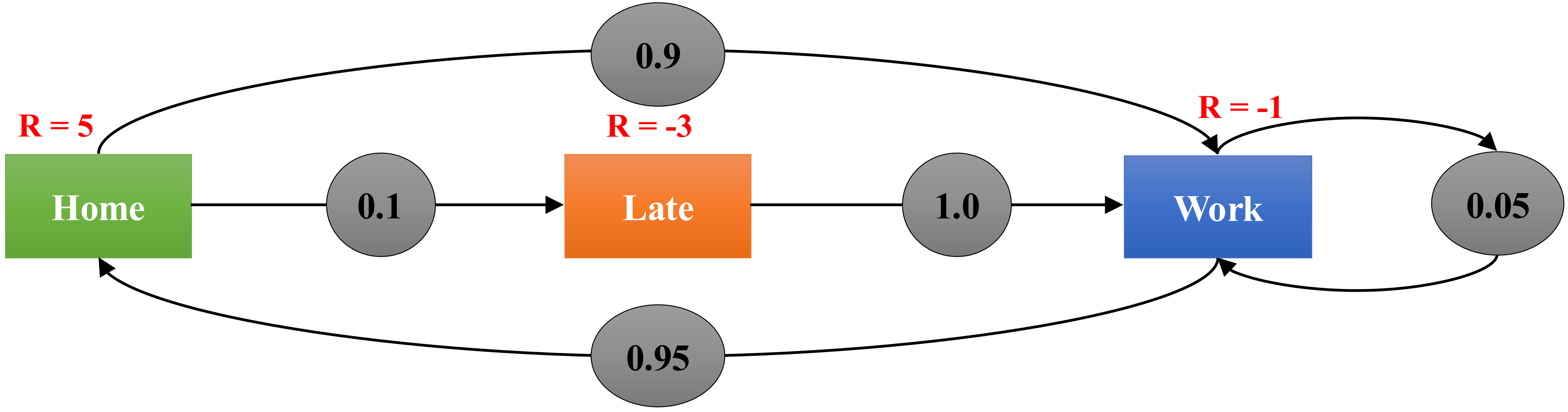
价值函数 $v(s)$ 表示状态 s 的长期价值，是从状态 s 开始的所有可能的马尔可夫链样本序列的回报**期望**，即，这些序列的回报乘以相应发生的概率，再进行累加。

若从状态 s 开始的所有可能的马尔可夫链样本序列的集合为 T，则

$$v(s) = E[G_t^\tau | \tau \in \mathrm{T}] = \sum_{\tau \in \mathrm{T}} G_t^\tau \times P(\tau)$$

示例：通勤问题

- 当 $\gamma = 0$，通勤问题对应的 MRP 中价值函数为
  - v(Home) = 5, v(Late) = -3, v(Work) = -1
- 当 $\gamma \neq 0$，如何计算价值函数?



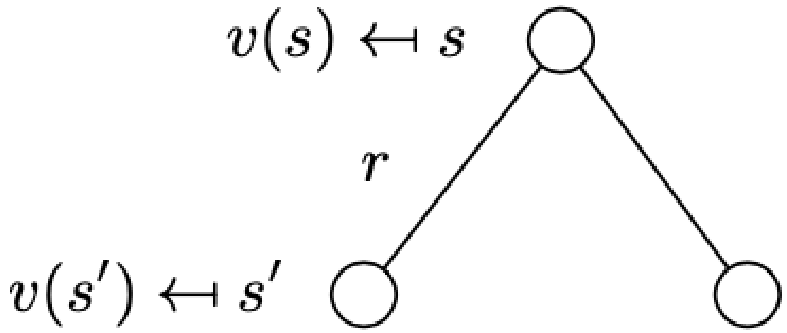
MRP 的贝尔曼方程（Bellman Equation）

- 价值函数可以分解为两部分：
  - 即时奖励 R_{t+1}
  - 后继状态折扣价值 $\gamma v(S_{t+1})$

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

MRP 的贝尔曼方程（Bellman Equation）

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



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

矩阵形式的贝尔曼方程

贝尔曼方程可以使用矩阵形式简洁地表示为

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

其中，v 是一个列向量，每个状态对应向量中的一个元素

$$\begin{bmatrix} v(S_1) \\ \vdots \\ v(S_n) \end{bmatrix} = \begin{bmatrix} R_{S_1} \\ \vdots \\ R_{S_n} \end{bmatrix} + \gamma \begin{bmatrix} P_{S_1 S_1} & \cdots & P_{S_1 S_n} \\ \vdots & & \vdots \\ P_{S_n S_1} & \cdots & P_{S_n S_n} \end{bmatrix} \begin{bmatrix} v(S_1) \\ \vdots \\ v(S_n) \end{bmatrix}$$

求解贝尔曼方程

贝尔曼方程是一个线性方程组，可直接求解

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

$$(I - \gamma \mathcal{P})\, v = \mathcal{R}$$

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

- 当状态数为 n 时，直接求解的计算复杂度为 $O(n^3)$，因此只适用于较小规模的 MRP
- 对于规模较大的 MRP，则通常采用迭代方法，例如：
  - 动态规划（Dynamic Programming）
  - 蒙特卡洛评估（Monte-Carlo Evaluation）
  - 时序差分学习（Temporal-Difference Learning）

示例：通勤问题

- 当 $\gamma = 0.5$，通勤问题对应的 MRP 中价值函数计算

$$\begin{bmatrix} v(\text{Home}) \\ v(\text{Late}) \\ v(\text{Work}) \end{bmatrix} = \begin{bmatrix} 5 \\ -3 \\ -1 \end{bmatrix} + 0.5 \begin{bmatrix} 0.0 & 0.1 & 0.9 \\ 0.0 & 0.0 & 1.0 \\ 0.95 & 0.0 & 0.05 \end{bmatrix} \begin{bmatrix} v(\text{Home}) \\ v(\text{Late}) \\ v(\text{Work}) \end{bmatrix}$$

$$\begin{bmatrix} v(\text{Home}) \\ v(\text{Late}) \\ v(\text{Work}) \end{bmatrix} = \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - 0.5 \begin{bmatrix} 0.0 & 0.1 & 0.9 \\ 0.0 & 0.0 & 1.0 \\ 0.95 & 0.0 & 0.05 \end{bmatrix} \right)^{-1} \begin{bmatrix} 5 \\ -3 \\ -1 \end{bmatrix}$$

$$\begin{bmatrix} v(\text{Home}) \\ v(\text{Late}) \\ v(\text{Work}) \end{bmatrix} = \begin{bmatrix} \frac{6806}{1199} \\ -\frac{2554}{1199} \\ \frac{2086}{1199} \end{bmatrix} \approx \begin{bmatrix} 5.6772 \\ -2.1301 \\ 1.7397 \end{bmatrix}$$

示例：通勤问题

- 当 $\gamma = 0.5$，通勤问题对应的 MRP 中价值函数计算
- 对计算结果进行验证

$$\begin{bmatrix} v(\text{Home}) \\ v(\text{Late}) \\ v(\text{Work}) \end{bmatrix} \approx \begin{bmatrix} 5 \\ -3 \\ -1 \end{bmatrix} + 0.5 \begin{bmatrix} 0.0 & 0.1 & 0.9 \\ 0.0 & 0.0 & 1.0 \\ 0.95 & 0.0 & 0.05 \end{bmatrix} \begin{bmatrix} 5.6772 \\ -2.1301 \\ 1.7397 \end{bmatrix}$$

$$\approx \begin{bmatrix} 5.67636 \\ -2.13015 \\ 1.74016 \end{bmatrix}$$

$$\begin{bmatrix} v(\text{Home}) \\ v(\text{Late}) \\ v(\text{Work}) \end{bmatrix} = \begin{bmatrix} 5 \\ -3 \\ -1 \end{bmatrix} + 0.5 \begin{bmatrix} 0.0 & 0.1 & 0.9 \\ 0.0 & 0.0 & 1.0 \\ 0.95 & 0.0 & 0.05 \end{bmatrix} \begin{bmatrix} \frac{6806}{1199} \\ -\frac{2554}{1199} \\ \frac{2086}{1199} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{6806}{1199} \\ -\frac{2554}{1199} \\ \frac{2086}{1199} \end{bmatrix}$$

目录

马尔可夫决策过程

- 马尔可夫决策过程是在马尔可夫奖励过程基础上增加了动作决策

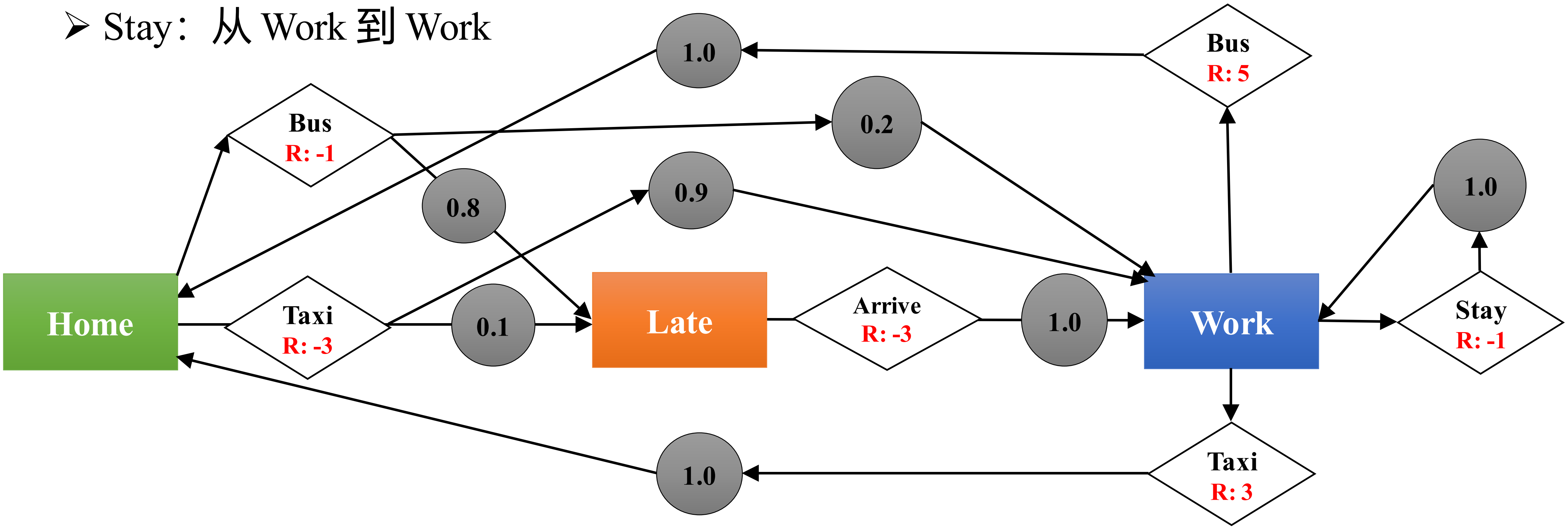
定义

一个**马尔可夫决策过程（Markov Decision Process, MDP）**可以表示为一个五元组 $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$，其中：

- $\mathcal{S}$ 是状态的（有限）集合
- $\mathcal{A}$ 是动作的（有限）集合
- $\mathcal{P}$ 是状态转移矩阵，$P_{ss'}^{a} = P(S_{t+1} = s' | S_t = s, A_t = a)$
- $\mathcal{R}$ 是奖励函数，$R_s^a = E[R_{t+1} | S_t = s, A_t = a]$
- γ 是折扣因子，$\gamma \in [0, 1]$

示例：通勤问题

- 通勤问题在状态 Home, Late, Work 的基础上，增加四个行动
  - Bus：乘坐巴士，从 Home 到 Late 或 Work，或者从 Work 到 Home
  - Taxi：乘坐出租车，从 Home 到 Late 或 Work，或者从 Work 到 Home
  - Arrive：从 Late 到 Work
  - Stay：从 Work 到 Work



示例：通勤问题

□ 通勤问题的 MDP 描述

- ➢ 状态集合：Home, Late, Work
- ➢ 动作集合：Bus, Taxi, Arrive, Stay
- ➢ 状态转移矩阵：

$$P_{ss'}^{\mathsf{Stay}} = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

$$P_{ss'}^{\mathsf{Bus}} = \begin{bmatrix} 0.0 & 0.8 & 0.2 \\ 0.0 & 1.0 & 0.0 \\ 1.0 & 0.0 & 0.0 \end{bmatrix} \quad P_{ss'}^{\mathsf{Taxi}} = \begin{bmatrix} 0.0 & 0.1 & 0.9 \\ 0.0 & 1.0 & 0.0 \\ 1.0 & 0.0 & 0.0 \end{bmatrix} \quad P_{ss'}^{\mathsf{Arrive}} = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

- ➢ 奖励函数：

$$R_s^{\mathsf{Bus}} = \begin{bmatrix} -1 \\ -\infty \\ 5 \end{bmatrix} \quad R_s^{\mathsf{Taxi}} = \begin{bmatrix} -3 \\ -\infty \\ 3 \end{bmatrix} \quad R_s^{\mathsf{Arrive}} = \begin{bmatrix} -\infty \\ -3 \\ -\infty \end{bmatrix} \quad R_s^{\mathsf{Stay}} = \begin{bmatrix} -\infty \\ -\infty \\ -1 \end{bmatrix}$$

策略

定义

策略（Policy） π 是在给定状态 s 时，采取各动作 a 的概率分布：

$$\pi(a|s) = P[A_t = a|S_t = s]$$

- 策略完全决定了智能体的行为
- 在 MDP 中，策略仅依赖于当前状态（与历史无关）
  - 换言之，策略是平稳的（与时间无关）：对于所有 $t > 0$，有

$$A_t \sim \pi(\cdot|S_t)$$

MDP 与 MP 和 MRP 的关系

给定一个 MDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ 及策略 $\boldsymbol{\pi}$:

- 状态序列 $S_1, S_2, \ldots$ 构成马尔可夫过程（Markov Process）$\langle S, P^{\pi} \rangle$
- 状态-奖励序列 $S_1, R_2, S_2, \ldots$ 构成马尔可夫奖励过程（MRP）

$$\langle \mathcal{S}, \mathcal{P}^{\pi}, \mathcal{R}^{\pi}, \gamma \rangle$$

其中:

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

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

价值函数

定义

在给定策略 $\boldsymbol{\pi}$ 的 MDP 中，**状态价值函数（State Value Function）** $v_\pi(s)$ 表示从状态 $\boldsymbol{s}$ 开始，随后按照策略 $\boldsymbol{\pi}$ 行动时的期望回报：

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

定义

在给定策略 $\boldsymbol{\pi}$ 的 MDP 中，**动作价值函数（Action Value Function）** $q_\pi(s,a)$ 表示从状态 $\boldsymbol{s}$ 开始，先执行行动作 $\boldsymbol{a}$，然后按照策略 $\boldsymbol{\pi}$ 行动时的期望回报：

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

贝尔曼期望方程（Bellman Expectation Equation）

- 对于状态价值函数，有以下分解：

$$v_\pi(s) = E_\pi[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s]$$

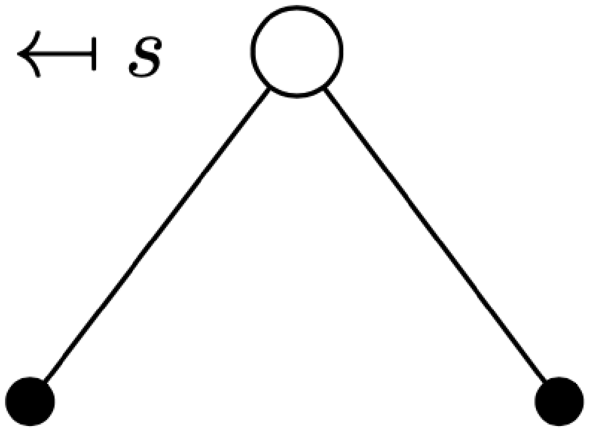
- 对于动作价值函数，同样可以分解：

$$q_\pi(s, a) = E_\pi[R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1}) \mid S_t = s, A_t = a]$$

v_π 的贝尔曼期望方程

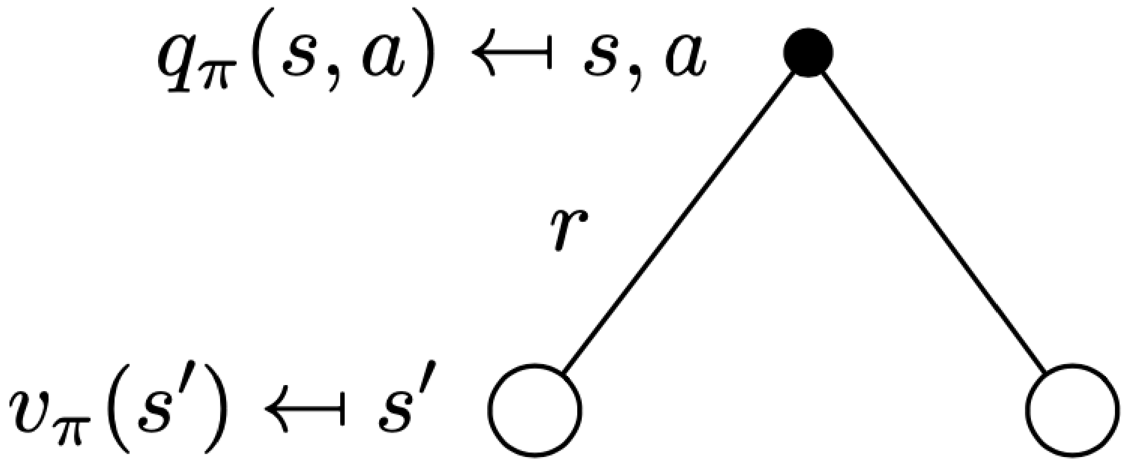
$v_\pi(s) \mapsfrom s$

$q_\pi(s, a) \mapsfrom a$



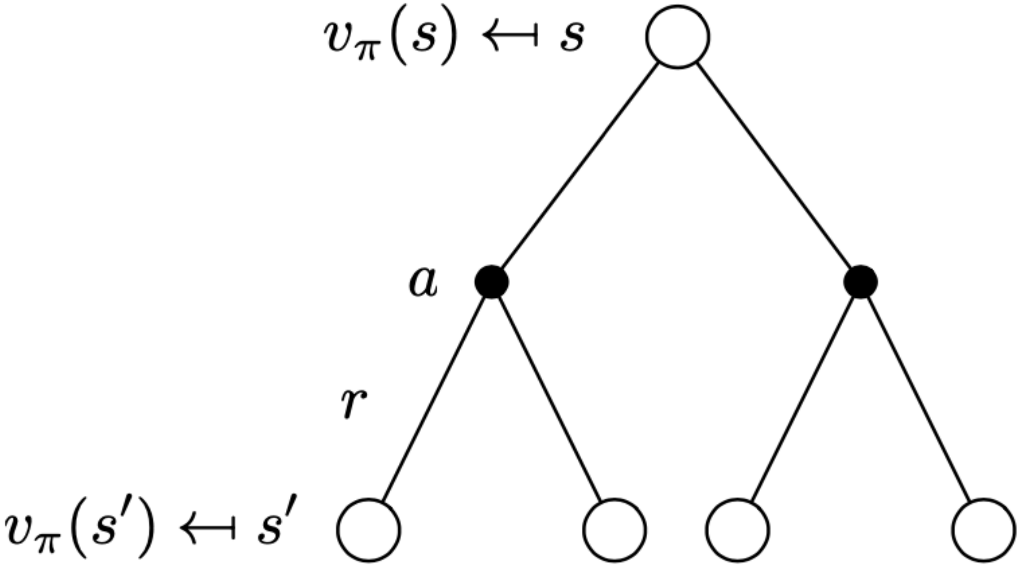
$$v_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) q_\pi(s, a)$$

q_π 的贝尔曼期望方程



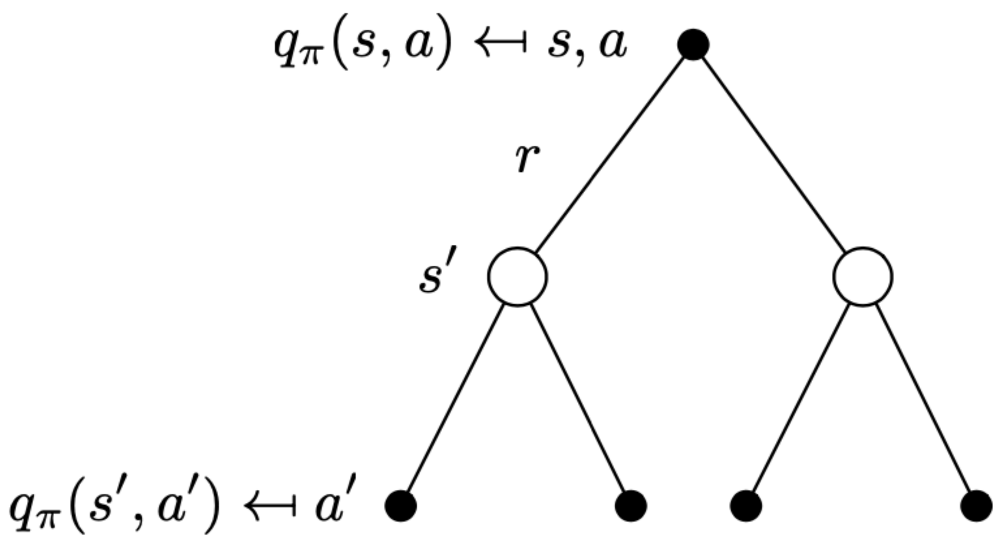
$$q_\pi(s,a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_\pi(s')$$

v_π 的贝尔曼期望方程



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

q_π 的贝尔曼期望方程



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

矩阵形式的贝尔曼期望方程

借鉴 MDP 与 MRP 之间的关系，贝尔曼期望方程可以使用矩阵形式简洁地表示为：

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

直接结果为：

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

$$P_{ss'}^\pi = \sum_{a \in \mathcal{A}} \pi(a|s) P_{ss'}^a \qquad R_s^\pi = \sum_{a \in \mathcal{A}} \pi(a|s) R_s^a$$

最优价值函数

定义

最优状态价值函数（Optimal State Value Function） $\boldsymbol{v}_*(\boldsymbol{s})$ 是对所有策略的状态价值函数的最大值：

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

最优动作价值函数（Optimal Action Value Function） $\boldsymbol{q}_*(\boldsymbol{s}, \boldsymbol{a})$ 是对所有策略的动作价值函数的最大值：

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

- 最优价值函数刻画在该 MDP 中能够获得的最佳可能表现
- 当我们知道了最优价值函数时，就意味着该 MDP 已被“解决”

最优策略

- 在策略之间定义偏序关系：

$$\pi \geq \pi' \text{ 当且仅当 } v_\pi(s) \geq v_{\pi'}(s), \text{对任意状态 } s$$

定理

对于任意马尔可夫决策过程（MDP）：

- 存在一个**最优策略（Optimal Policy）**，它不劣于任何其他策略，即

$$\pi_* \geq \pi \text{ 对所有策略 } \pi \text{ 都成立}$$

- 所有最优策略都能实现最优状态价值函数，即

$$\text{对所有状态 } \boldsymbol{s}，\ v_{\pi_*}(s) = v_*(s)$$

- 所有最优策略都能实现最优动作价值函数，即

$$\text{对所有状态 } \boldsymbol{s} \text{ 和动作 } \boldsymbol{a}，\ q_{\pi_*}(s,a) = q_*(s,a)$$

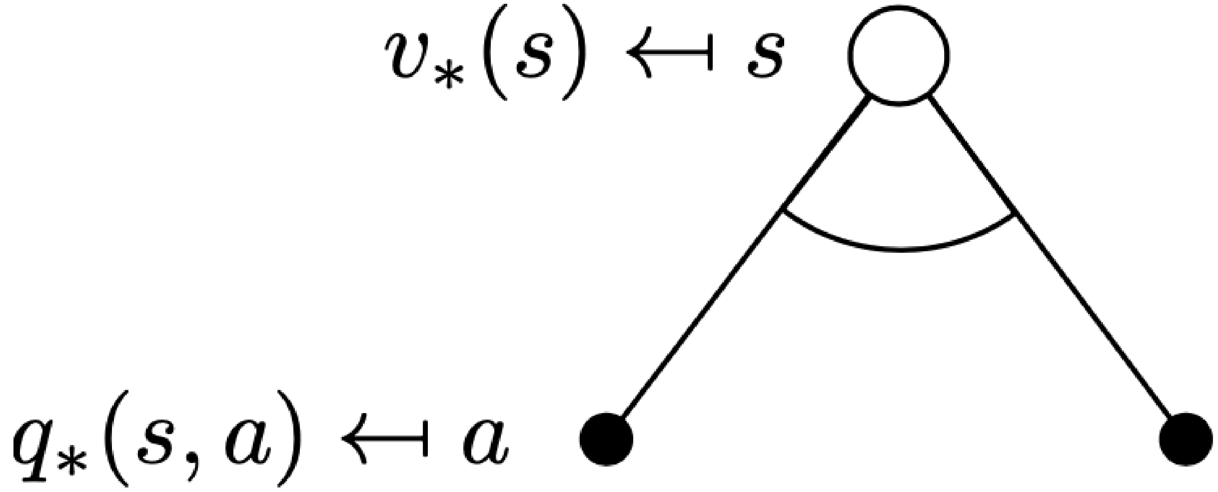
寻找最优策略

- 可以通过对 $q_*(s,a)$ 取最大值来构造最优策略：

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

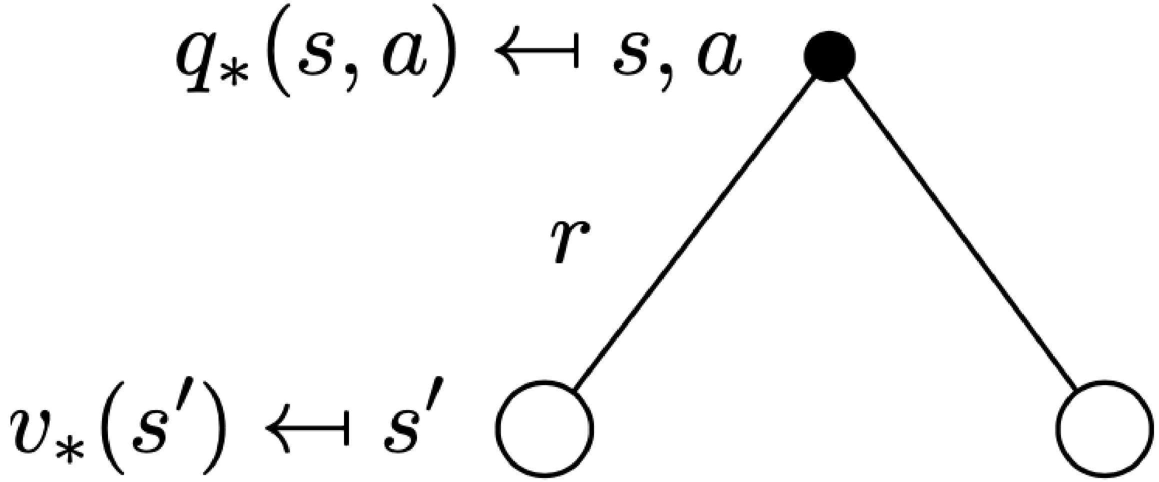
- 对于任意 MDP，总存在一个确定性的最优策略
- 一旦我们知道了 $q_*(s,a)$ ，便可立刻得到该最优策略

v_* 的贝尔曼最优方程（Bellman Optimality Equation）



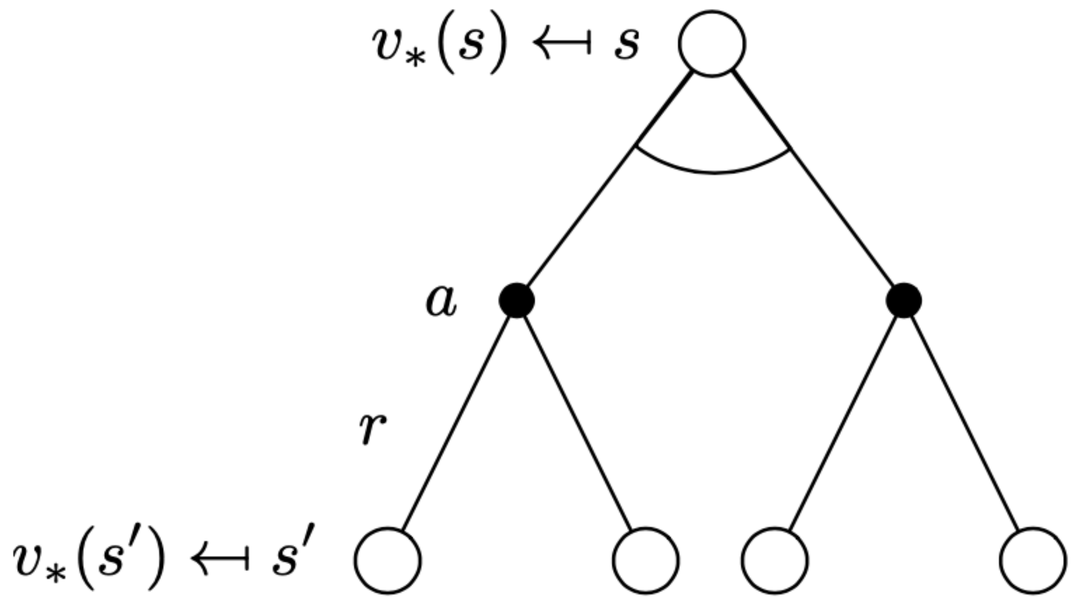
$$v_*(s) = \max_a q_*(s,a)$$

q_* 的贝尔曼最优方程



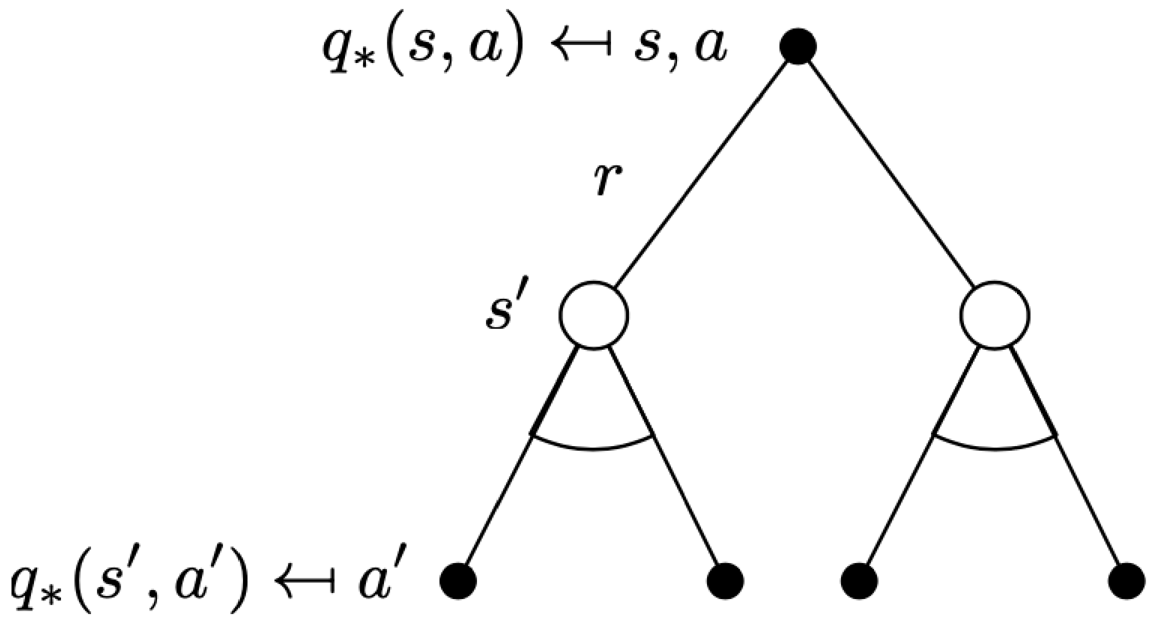
$$q_*(s,a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_*(s')$$

v_* 的贝尔曼最优方程



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

q_* 的贝尔曼最优方程



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

贝尔曼期望方程和贝尔曼最优方程

$$V_\pi(s) = \sum_{a \in A} \pi(a \mid s)\left(R(s, a) + \gamma \sum_{s' \in S} P(s' \mid s, a) V_\pi(s')\right),$$

$$V^*(s) = \max_{a \in A}\left(R(s, a) + \gamma \sum_{s' \in S} P(s' \mid s, a) V^*(s')\right),$$

$$Q_\pi(s, a) = R(s, a) + \gamma \sum_{s' \in S} P(s' \mid s, a) \sum_{a' \in A} \pi(a' \mid s') Q_\pi(s', a'),$$

$$Q^*(s, a) = R(s, a) + \gamma \sum_{s' \in S} P(s' \mid s, a) \max_{a' \in A} Q^*(s', a').$$

求解贝尔曼最优方程

- 贝尔曼最优方程是非线性的
- （一般）不存在封闭形式解
- 常见的迭代求解方法：
  - 值迭代（Value Iteration）
  - 策略迭代（Policy Iteration）
  - Q 学习（Q-Learning）
  - Sarsa

目录

部分可观察马尔可夫决策过程

- 当状态不可完全观察时，通过引入观察和观察函数来扩展 MDP

定义

一个**部分可观察马尔可夫决策过程（Partially Observable Markov Decision Process, POMDP）**可以表示为一个七元组 $\langle \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{P}, \mathcal{R}, \mathcal{Z}, \gamma \rangle$，其中：

- $\mathcal{S}$ 是状态的（有限）集合
- $\mathcal{A}$ 是动作的（有限）集合
- $\mathcal{O}$ 是观察的（有限）集合
- $\mathcal{P}$ 是状态转移矩阵，$P_{ss'}^a = P(S_{t+1} = s' | S_t = s, A_t = a)$
- $\mathcal{R}$ 是奖励函数，$R_s^a = E[R_{t+1} | S_t = s, A_t = a]$
- $\mathcal{Z}$ 是观察函数，$Z_{s'o}^a = P(O_{t+1} = o \mid S_{t+1} = s', A_t = a)$
- γ 是折扣因子，$\gamma \in [0, 1]$

信念状态

定义

历史（History） H_t 是动作、观测和奖励构成的序列：

$$H_t = A_0, O_1, R_1, \dots, A_{t-1}, O_t, R_t$$

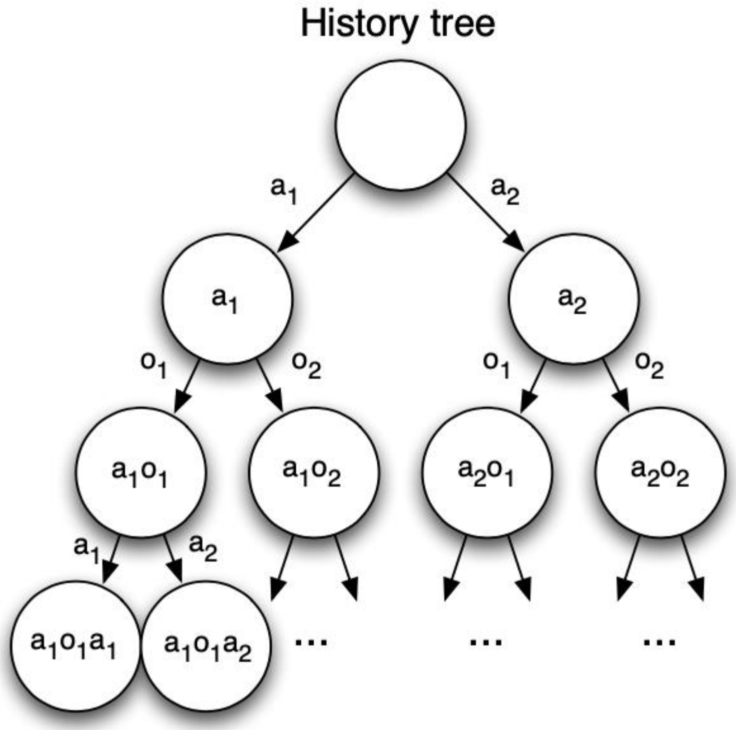
定义

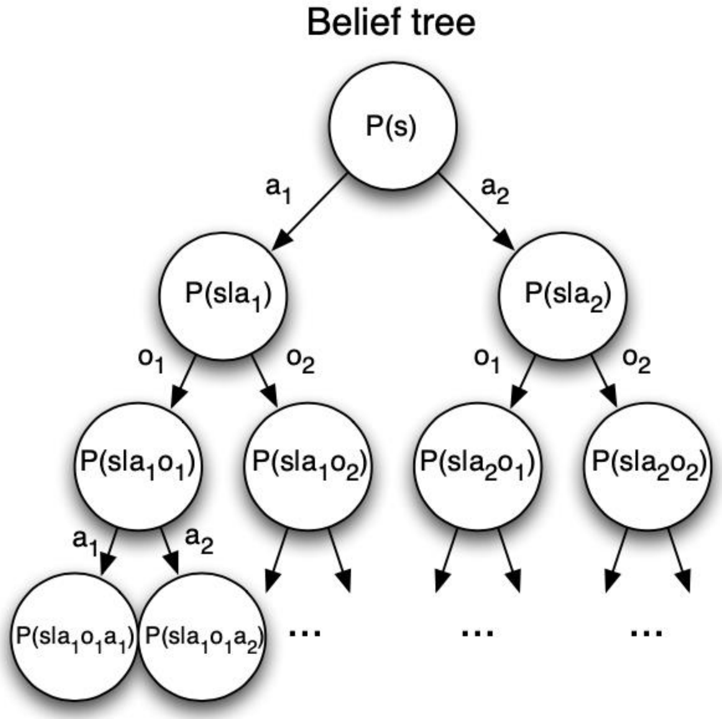
信念状态（Belief State） $b(h)$ 是在给定历史 h 的条件下，对所有可能状态的概率分布：

$$b(h) = \left(P[S_t = s^1 \mid H_t = h], \dots, P[S_t = s^n \mid H_t = h] \right)$$

POMDP 的简化

- 历史 H_t 满足马尔可夫性质，POMDP 可以简化为一个（无限的）历史树
  - 给定完整的历史序列 H_t，未来状态只依赖于该历史
- 信念状态 $b(H_t)$ 满足马尔可夫性质，POMDP 可以简化为一个（无限的）信念状态树
  - 给定当前的信念分布 $b_t = b(H_t)$，未来只依赖于该分布，而与过往历史无关





课后作业

1. 若 MDP 例子“通勤问题”的折扣因子 γ=0.5，请计算如下策略 π 的状态价值函数 $v_\pi(s)$：

$$\pi(Taxi \mid Home) = 1,\ \pi(Arrive \mid Late) = 1,\ \pi(Bus \mid Work) = 1。$$

谢谢！