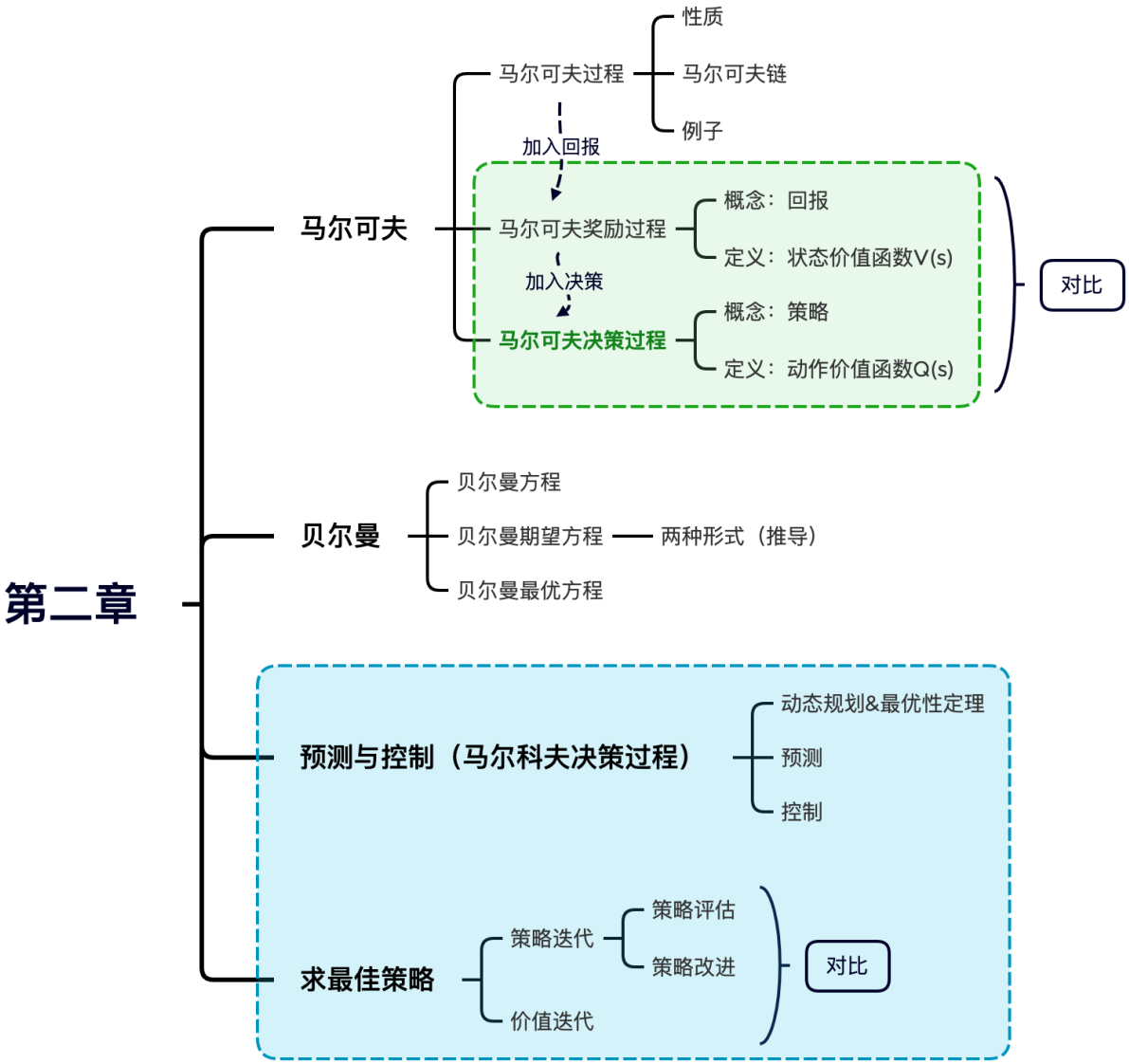
《EasyRL》学习笔记 · 强化学习中的马尔可夫

by：Jackeeee_M

1 引言

最近正在学习强化学习的相关知识，主要以《**EasyRL**强化学习教程》为主要参考，希望将每一章的关键知识点整理成笔记记录下来，既方便自己之后温故而知新，也希望能给同样在学习的大家一个参考。

笔记从第二章《马尔可夫》开始，笔记对于本章知识点的顺序略做了调整，如下图所示。

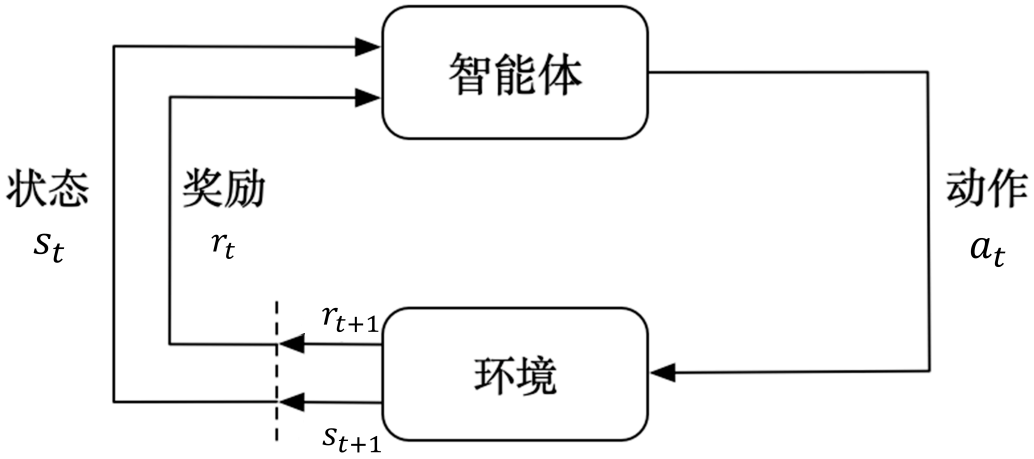


Tips：文中的插图以及许多描述均摘自《EasyRL强化学习教程》，仅做整理与个人的理解。

2 马尔可夫

强化学习（**reinforcement learning**，**RL**）讨论的问题是智能体（**agent**）怎么在复杂、不确定的环境（**environment**）里面去最大化它能获得的奖励。为了实现这样一个过程，我们首先则需要对这一过程进行建模。

其中，智能体与环境交互的过程可以简化为下图，而这一交互过程，大多都可以建模为马尔可夫决策过程（**Markov Decision Process, MDP**），而马尔科夫决策过程则是基于其简化版本马尔可夫过程（**Markov process**，**MP**）以及马尔可夫奖励过程（**Markov reward process**，**MRP**）。



2.1 马尔可夫过程

2.1.1 马尔可夫性质

首先，为了理解马尔可夫过程，我们首先需要对其性质进行了解。

在随机过程中，马尔可夫性质（**Markov property**）是指**一个随机过程在给定现在状态及所有过去状态情况下，其未来状态的条件概率分布仅依赖于当前状态**。

马尔可夫性质也可以描述为**给定当前状态时，将来的状态与过去状态是条件独立的**。如果某一个过程满足马尔可夫性质，那么未来的转移与过去的是独立的，它只取决于现在。

马尔可夫性质是所有马尔可夫过程的基础。

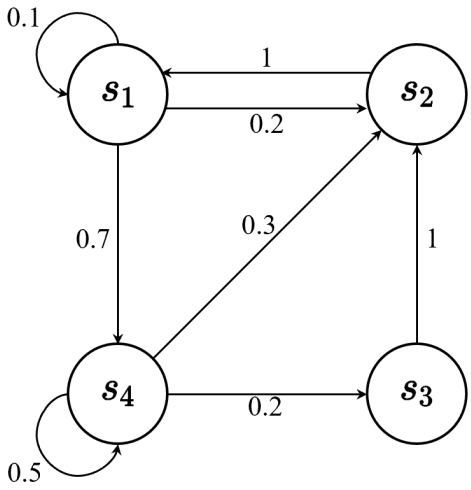
2.1.2 马尔可夫链

马尔可夫过程是一组具有马尔可夫性质的随机变量序列$s_1, \ldots, s_t$，其中下一个时刻的状态s_{t+1}只取决于当前状态s_t。我们设状态的历史为$h_t = \{s_1, s_2, \ldots, s_t\}$，其包含了**所有的**之前状态，因此马尔可夫过程的**状态转移**满足条件：

$$p(s_{t+1}|s_t) = p(s_{t+1}|h_t) \tag{1}$$

即**未来与过去无关**。

而**离散的马尔科夫过程便可称之为马尔可夫链**，其状态是有限的，如下图所示。



2.1.3 马尔可夫过程例子

至此，应当对马尔可夫过程有了基本的了解，其中最重要的便是**马尔可夫性质**，之后的马尔可夫奖励过程与马尔可夫决策过程均基于此基础上拓展，而马尔可夫也是强化学习中最为基础的知识。相关的例子**Bilibili**上有许多，可自行搜索学习。

2.2 马尔可夫奖励过程

马尔可夫奖励过程（**Markov reward process, MRP**）是马尔可夫链加上奖励函数。在马尔可夫奖励过程中，**只是多了奖励函数（reward function）**。奖励函数**R** 是一个期望，表示当我们到达某一个状态的时候，可以获得多大的奖励，这里另外定义了折扣因子γ。

$$R(s_t = s) = \mathbb{E}[r_t | s_t = s] \tag{2}$$

如果状态数是有限的，那么奖励函数**R** 可以是一个向量。

基于奖励函数**R**，我们可以进一步定义**回报函数G**与**状态价值函数V**。

2.2.1 马尔可夫奖励过程中的回报函数

回报（**return**）可以定义为奖励的逐步叠加，假设时刻**t**后的奖励序列为$r_{t+1}, r_{t+2}, r_{t+3}, \ldots$，且则假设我们的的时间步**t**最多可以到**T**，那么回报为

$$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \gamma^3 r_{t+4} + \cdots + \gamma^{T-t-1} r_T \tag{3}$$

其中γ即折扣因子（$\gamma < 1$），因此越往后的奖励打的折扣越多。这说明我们对当前的奖励更加重视，而对未来的奖励却不那么重视。通过调节折扣因子γ，我们就可以调节我们对未来奖励的重视程度。

2.2.2 马尔可夫奖励过程中的状态价值函数

对于马尔可夫奖励过程，**状态价值函数V被定义成回报函数G的期望**，即

$$\begin{aligned} V^t(s) =& \mathbb{E}[G_t | s_t = s] \\ =& \mathbb{E}[r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \gamma^3 r_{t+4} + \cdots + \gamma^{T-t-1} r_T | s = s_t] \end{aligned} \tag{4}$$

2.3 马尔可夫决策过程

之前我们提到，马尔可夫奖励过程是马尔可夫链加上奖励函数，那么现在我们可以引出马尔可夫决策过程了。

马尔科夫决策过程就是马尔可夫奖励过程加上决策。

未来的状态不仅依赖于当前的状态，也依赖于在当前状态下智能体采取的动作，因此马尔可夫决策过程的**状态转移**满足条件：

$$p(s_{t+1}|s_t, a_t) = p(s_{t+1}|h_t, a_t) \tag{5}$$

对应的，奖励函数也需要做出相应的改变：

$$R(s_t = s, a_t = a) = \mathbb{E}[r_t|s_t = s, a_t = a] \tag{6}$$

2.3.1 马尔可夫决策过程中的策略函数

我们知道，在马尔可夫决策过程中，下一时刻的状态是由当前时刻的状态与动作共同决定的，那么我们**该如何选取当前时刻的动作**呢？

此时，我们就需要引入**策略函数**π，如下

$$\pi(a|s) = p(a_t = a|s_t = s) \tag{7}$$

策略的输出为概率，即在当前状态s下选取动作a的概率是多少。

2.3.2 马尔可夫决策过程中的价值函数

参考之前马尔可夫奖励过程中所给定的价值函数，对于马尔可夫决策过程，我们也可以定义相应的价值函数为

$$V_\pi(s) = \mathbb{E}_\pi[G_t|s_t = s] \tag{8}$$

其中，期望基于我们采取的策略。**当策略决定后，我们通过对策略进行采样来得到一个期望，计算出它的价值函数。**这里我们另外引入了一个**Q函数（Q-function）**。Q函数也被称为**动作价值函数（action-valuefunction）**。

动作价值函数**Q**函数定义的是在某一个状态采取某一个动作，它有可能得到的回报的一个期望，即

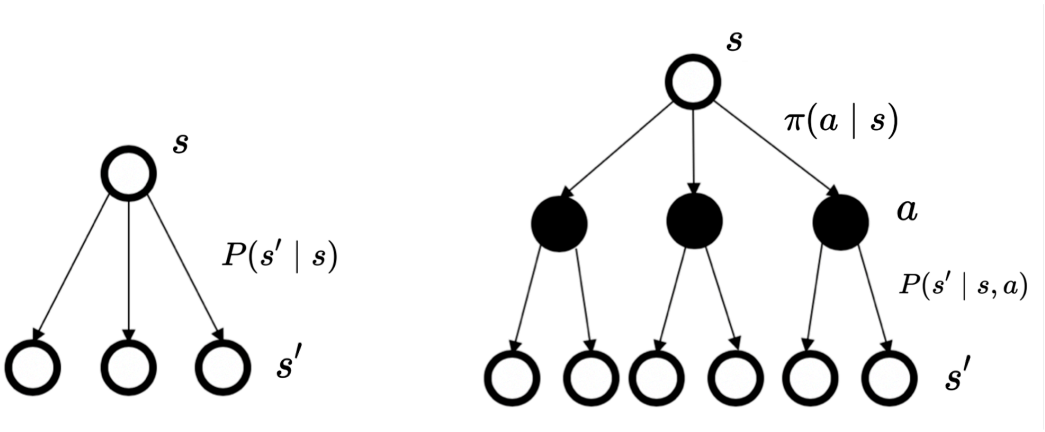
$$Q_\pi(s, a) = \mathbb{E}_\pi[G_t|s_t = s, a_t = a] \tag{9}$$

此处的期望也是基于策略函数$\pi(a|s)$的，而策略函数的输出所代表的恰恰是概率，因此我们可以基于策略函数的分布对动作价值函数**Q**进行加权和，从而得到价值函数：

$$V_\pi(s) = \sum_{a\in A} \pi(a|s)Q_\pi(s, a) \tag{10}$$

至此，我们也得到了在马尔可夫决策过程中，**V**函数与**Q**函数的转化关系式。

2.4 马尔可夫决策过程和马尔可夫奖励过程的区别



上图左侧为马尔可夫奖励过程，右侧为马尔可夫决策过程。

（1）对于马尔可夫奖励过程，从当前状态s开始，其可以直接根据概率$p(s'|s)$进行状态转移，从而切换至状态s'。

（2）对于马尔可夫决策过程，显而易见的，在状态s与状态s'之间多了一层动作a，我们需要先根据策略函数$\pi(a,s)$决定应当采取的动作a，之后再根据概率$p(s'|s,a)$进行状态转移，从而切换至状态s'。

3 贝尔曼

在强化学习中，马尔可夫是离不开的核心，在上文的学习中，我们也对马尔可夫奖励过程和马尔科夫决策过程有了简单的了解。相应的，我们也定义了回报函数、策略函数以及**价值函数**等...

3.1 贝尔曼方程

首先，我们以马尔可夫奖励过程为例，我们很容易有疑问，**该如何求解其价值函数**呢?

回顾之前的定义，状态价值函数$\mathbf{V}$被定义成回报函数$\mathbf{G}$的期望

$$V^t(s)=\mathbb{E}[G_t|s_t=s] \tag{11}$$

可见，价值函数是与后续状态的回报紧密相连的，而从中我们便可以推出**贝尔曼方程**，推导如下

$$\begin{aligned}V(s)=&\mathbb{E}[G_t|s_t=s]\\=&\mathbb{E}[r_{t+1}+\gamma r_{t+2}+\gamma^2 r_{t+3}+\gamma^3 r_{t+4}+\cdots|s_t=s]\\=&\mathbb{E}[r_{t+1}|s_t=s]+\gamma\mathbb{E}[r_{t+2}+\gamma^2 r_{t+3}+\gamma^3 r_{t+4}+\cdots|s_t=s]\\=&R(s)+\gamma\mathbb{E}[G_{t+1}|s_t=s]\\=&R(s)+\gamma\mathbb{E}[V(s_{t+1})|s_t=s]\\=&R(s)+\gamma\sum_{s'\in S}p(s'|s)V(s')\end{aligned} \tag{12}$$

其中，对于倒数第二个等号，个人题提供一个不严谨的理解：**期望的期望等于期望**，严谨的证明可参考书中，仿照全期望公式进行了推导。

贝尔曼方程就是当前状态与未来状态的迭代关系，表示当前状态的价值函数可以通过下个状态的价值函数来计算。

贝尔曼方程被广泛应用于动态规划方法中，我们通过对价值函数反复的自举（**bootstrapping**）迭代，直到价值函数收敛，即可得到最终的价值函数值。

计算马尔可夫奖励过程价值的动态规划算法如下图所示

1: 对于所有状态 $s \in S, V'(s) \leftarrow 0, V(s) \leftarrow \infty$
2: 当$||V - V'|| > \epsilon$执行
3: $V \leftarrow V'$
4: 对于所有状态 $s \in S, V'(s) = R(s) + \gamma \sum_{s' \in S} P(s'|s)V(s')$
5: 结束循环
6: 返回$V'(s)$,对于所有状态$s \in S$

3.2 贝尔曼期望方程

讲完了马尔可夫奖励过程中的贝尔曼方程，我们可以开始讲马尔可夫决策过程了。同样，我们可以对马尔可夫奖励过程中的价值函数进行分解。

对状态价值函数$V_\pi(s)$与动作价值函数$Q_\pi(s)$分别进行分解，如下

$$\begin{aligned} V_\pi(s) =& \mathbb{E}_\pi[G_t|s_t = s] \\ =& \mathbb{E}_\pi[r_{t+1} + \gamma V_\pi(s_{t+1})|s_t = s] \end{aligned} \tag{13}$$

$$\begin{aligned} Q_\pi(s,a) =& \mathbb{E}_\pi[G_t|s_t = s, a_t = a] \\ =& \mathbb{E}_\pi[r_{t+1} + \gamma Q_\pi(s_{t+1}, a_{t+1})|s_t = s, a_t = a] \end{aligned} \tag{14}$$

上述二式，分别为马尔可夫决策过程中$V_\pi(s)$与$Q_\pi(s)$的**贝尔曼期望方程**，其与马尔可夫奖励过程的贝尔曼方程类似。

对贝尔曼期望方程进行进一步的化简，可得

$$V_\pi(s) = \sum_{a \in A} \pi(a|s) Q_\pi(s,a) \tag{15}$$

$$Q_\pi(s,a) = R(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V_\pi(s') \tag{16}$$

我们将式（**16**）代入式（**15**）中，可得

$$V_\pi(s) = \sum_{a \in A} \pi(a|s)(R(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V_\pi(s')) \tag{17}$$

将式（**15**）代入式（**16**），可得

$$Q_\pi(s,a) = R(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) \sum_{a \in A} \pi(a'|s') Q_\pi(s',a') \tag{18}$$

即得到贝尔曼期望方程的第二种形式，为式（**17**）与式（**18**）。

至此我们得到了贝尔曼期望方程的两种形式：

形式一：

$$\begin{cases} V_\pi(s) =& \mathbb{E}_\pi[r_{t+1} + \gamma V_\pi(s_{t+1})|s_t = s] \\ Q_\pi(s,a) =& \mathbb{E}_\pi[r_{t+1} + \gamma Q_\pi(s_{t+1}, a_{t+1})|s_t = s, a_t = a] \end{cases} \tag{19}$$

形式二：

$$\begin{cases} V_\pi(s) =& \sum_{a \in A} \pi(a|s) \left(R(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V_\pi(s') \right) \\ Q_\pi(s,a) =& R(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) \sum_{a \in A} \pi(a'|s') Q_\pi(s',a') \end{cases} \tag{20}$$

3.3 贝尔曼最优方程

此处为了连贯性，先给出贝尔曼最优方程的定义，之后在**策略迭代**中会重新提及。

$$V_\pi(s) = \max_{a \in A} Q_\pi(s,a) \tag{21}$$

4 预测与控制

预测（**prediction**）和控制（**control**）是马尔可夫决策过程里面的核心问题。

预测（评估一个给定的策略）的输入是马尔可夫决策过程$< S, A, P, R, \gamma >$和策略π，输出是价值函数V_π。预测是指给定一个马尔可夫决策过程以及一个策略π ，计算它的价值函数，也就是计算每个状态的价值。

控制（搜索最佳策略）的输入是马尔可夫决策过程$< S, A, P, R, \gamma >$，输出是最佳价值函数（**optimalvalue function**）V^*和最佳策略（**optimal policy**）π^*。控制就是我们去寻找一个最佳的策略，然后同时输出它的最佳价值函数以及最佳策略。

4.1 动态规划

在聊到动态规划之前，我们首先需要了解什么是**最优性原理**:

多阶段决策过程的最优决策序列具有这样的性质：不论初始状态和初始决策如何，对于前面决策所造成的某一状态而言，其后各阶段的决策序列必须构成最优策略。

而动态规划（**dynamic programming**，**DP**）适合解决满足最优子结构（**optimal substructure**）和重叠子问题（**overlapping subproblem**）两个性质的问题，其通常用于求解具有某种最优性质的问题。

马尔可夫决策过程是满足动态规划要求的，因此我们可以使用动态规划完成**预测问题**与**控制问题**的求解。在此之前，我们已经推导了贝尔曼方程，我们可以把它分解成递归的结构后，进行迭代求解。

4.2 马尔可夫决策过程中的策略评估

策略评估就是给定马尔可夫决策过程和策略，评估我们可以获得多少价值，即对于当前策略，我们可以得到多大的价值。

我们可以将式（**20**）中**V**函数的贝尔曼方程改写如下:

$$V_\pi^{t+1}(s) = \sum_{a \in A} \pi(a|s) \left(R(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V_\pi^t(s') \right) \tag{22}$$

式（**21**）是指我们可以把**贝尔曼期望方程**视为**动态规划的迭代**。当我们得到上一时刻的V_t 的时候，就可以通过递推的关系推出下一时刻的值。反复迭代，最后的V值就是从V_1、V_2 到最后收敛之后的值V_π。而V_π 就是当前给定的策略π对应的价值函数。

由于策略函数$\pi(a|s)$已经给定，故马尔可夫决策过程退化为马尔可夫奖励过程，式（**21**）也可以相应的写为马尔可夫奖励过程的表达形式，如下

$$V_{t+1}(s) = r_\pi(s) + \gamma P_\pi(s'|s) V_t(s') \tag{23}$$

4.3 马尔可夫决策过程中的过程控制

过程控制就是只给定马尔可夫决策过程，去寻找**最佳策略**$\pi^*(s)$，从而得到**最佳价值函数**$V^*(s)$，其中最佳价值函数的定义为

$$V^*(s) = \max_{\pi} V_{\pi}(s) \tag{24}$$

最佳价值函数是指，我们搜索一种策略π让每个状态的价值最大。V^*就是到达每一个状态，它的值的最大化情况。在这种最大化情况中，我们得到的策略就是最佳策略，即

$$\pi^*(s) = \arg\max_{\pi} V_{\pi}(s) \tag{25}$$

寻找最佳策略的过程就是马尔可夫决策过程的控制过程。马尔可夫决策过程控制就是去寻找一个最佳策略使我们得到一个最大的价值函数值。

5 求最优策略

对于一个事先定好的马尔可夫决策过程，当智能体采取最佳策略的时候，最佳策略一般都是确定的，而且是稳定的（它不会随着时间的变化而变化）。但最佳策略不一定是唯一的，多种动作可能会取得相同的价值。

我们可以通过**策略迭代**和**价值迭代**来解决马尔可夫决策过程的控制问题。

5.1 策略迭代

首先，我们需要先进行初始化，初始化一个**状态价值函数**V和**策略**π。

策略迭代主要分为两步，分别是：**策略评估**与**策略改进**。

（**1**）策略评估

我们先保证当下策略不变，然后估计它的价值，即给定当前的策略函数来估计状态价值函数，公式同式（**22**），即**V**函数的贝尔曼方程

$$V_{\pi}^{t+1}(s) = \sum_{a \in A} \pi(a|s) \left(R(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V_{\pi}^{t}(s') \right) \tag{26}$$

（**2**）策略改进

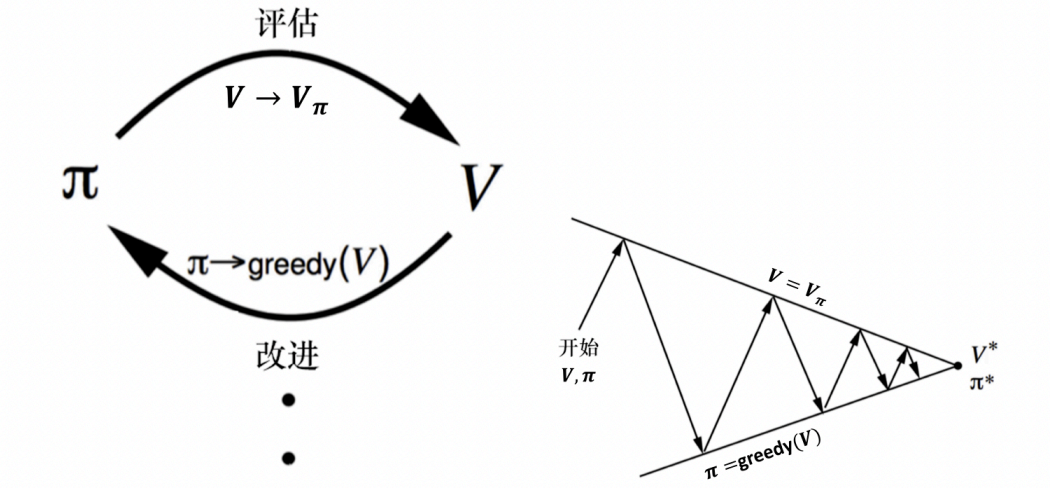
得到状态价值函数后，我们可以进一步推算出它的**Q**函数，公式同式（**16**），即贝尔曼期望方程的化简形式。

$$Q_{\pi_i}(s,a) = R(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V_{\pi_i}(s') \tag{27}$$

得到**Q**函数后，我们直接对**Q**函数进行最大化，通过在**Q**函数做一个贪心搜索来进一步改进策略（即我们希望通过所得策略可以获得最大的回报），从而得到新的策略$\pi_{i+1}(s)$。

$$\pi_{i+1}(s) = \arg\max_{a} Q_{\pi_i}(s,a) \tag{28}$$

上述两步**交替迭代**进行，过程如下图所示



我们首先以初始化的$\mathbf{V}$与π作为初始状态，对其进行策略评估，从而获得新的状态价值函数$\mathbf{V}$；再根据所得新的$\mathbf{V}$函数计算其对应的$\mathbf{Q}$函数；最终对$\mathbf{Q}$函数取最大，从而得到新的策略π。至此我们完成了一个回合的策略迭代，得到了新的状态价值函数与新的策略，将二者作为新的初始状态继续迭代，最终逐渐收敛，此时得到的便是**最佳状态价值函数**V^*和最佳策略π^*。

5.1.1 贝尔曼最优方程

由上文分析可知，我们选取策略的方法是对$\mathbf{Q}$值不停做贪心操作（**argmax**），我们就会得到更好的或者不变的策略，而不会使价值函数变差。所以当改进停止后，我们就会得到一个最佳策略。当改进停止后，我们取让$\mathbf{Q}$函数值最大化的动作，$\mathbf{Q}$函数就会直接变成价值函数，即

$$Q_\pi(s,\pi'(s)) = \max_{a\in A} Q_\pi(s,a) = Q_\pi(s,\pi(s)) = V_\pi(s) \tag{29}$$

这也正是**2.3**节中所提到的贝尔曼最优方程

$$V_\pi(s) = \max_{a\in A} Q_\pi(s,a) \tag{30}$$

贝尔曼最优方程表明：**最佳策略下的一个状态的价值必须等于在这个状态下采取最好动作得到的回报的期望。**

当马尔可夫决策过程满足贝尔曼最优方程的时候，整个马尔可夫决策过程已经达到最佳的状态。只有当整个状态已经收敛后，我们得到最佳价值函数后，贝尔曼最优方程才会满足。满足贝尔曼最优方程后，我们可以采用最大化操作，即

$$V^*(s) = \max_{a} Q^*(s,a) \tag{31}$$

当我们取让$\mathbf{Q}$函数值最大化的动作对应的值就是当前状态的最佳的价值函数的值。

5.2 价值迭代

之前，我们已经介绍了**贝尔曼最优方程**，而在价值迭代中，我们将贝尔曼最优方程作为一个更新规则来进行迭代，即

$$V(s) \leftarrow \max_{a\in A}\left(R(s,a) + \gamma \sum_{s'\in S} p(s'|s,a)V(s')\right) \tag{32}$$

只有当整个马尔可夫决策过程已经达到最佳的状态时，式（**31**）才满足，因此我们通过对贝尔曼最优方程不断迭代，最终价值函数便会收敛至最佳V^*。

价值迭代算法的过程如下：

（**1**）初始化：令$k=1$，对于所有状态s，$V_0(s)=0$。

（**2**）对于$k=1:H$（H是让$V(s)$收敛所需的迭代次数）

（**a**）对于所有状态s

$$Q_{k+1}(s,a)=R(s,a)+\gamma\sum_{s'\in S}p(s'|s,a)V_k(s') \tag{33}$$

$$V_{k+1}(s)=\max_a Q_{k+1}(s,a) \tag{34}$$

（**b**）$k\leftarrow k+1$。

（**3**）在迭代后提取**最优策略**:

$$\pi(s)=\arg\max_a\left[R(s,a)+\gamma\sum_{s'\in S}p(s'|s,a)V_{H+1}(s')\right] \tag{35}$$

5.3 策略迭代与价值迭代的不同

策略迭代与**价值迭代**都可以解马尔可夫决策过程的控制问题，并以获得最佳策略π^*为目的。

策略迭代主要分两步。首先进行策略评估，即对当前已经搜索到的策略函数进行估值。得到估值后，我们进行策略改进，即把**Q**函数算出来，进行进一步改进。不断**重复这两步**，直到策略收敛。

价值迭代直接使用贝尔曼最优方程进行迭代，从而寻找最佳价值函数。**找到最佳价值函数后，我们再提取最佳策略。**

因此我们可以把价值迭代理解为只进行一次策略改进的策略迭代算法。