第13章：强化学习

胡成玉 |  **中国地质大学（武汉）**

机器学习与数据挖掘团队

huchengyu@cug.edu.cn

http://grzy.cug.edu.cn/huchengyu/

本章内容

一、强化学习概述

二、有模型学习

三、无模型学习

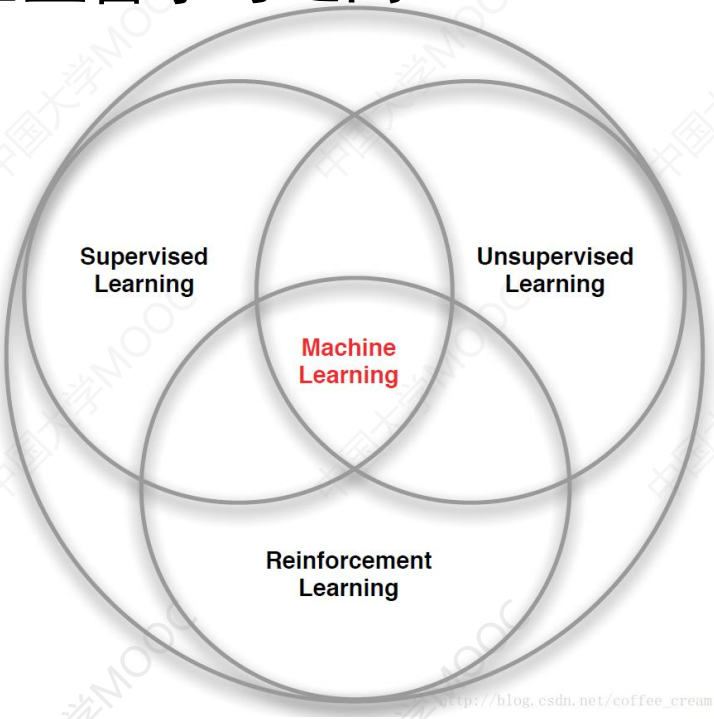
四、对强化学习的理解

一、强化学习概述

基本概念：强化学习又称为增强学习、加强学习、再励学习或激励学习，是一种从环境状态到行为映射的学习，目的是使动作从环境中获得的累积回报值最大。强化学习是机器学习分支之一，介于监督学习和无监督学习之间。

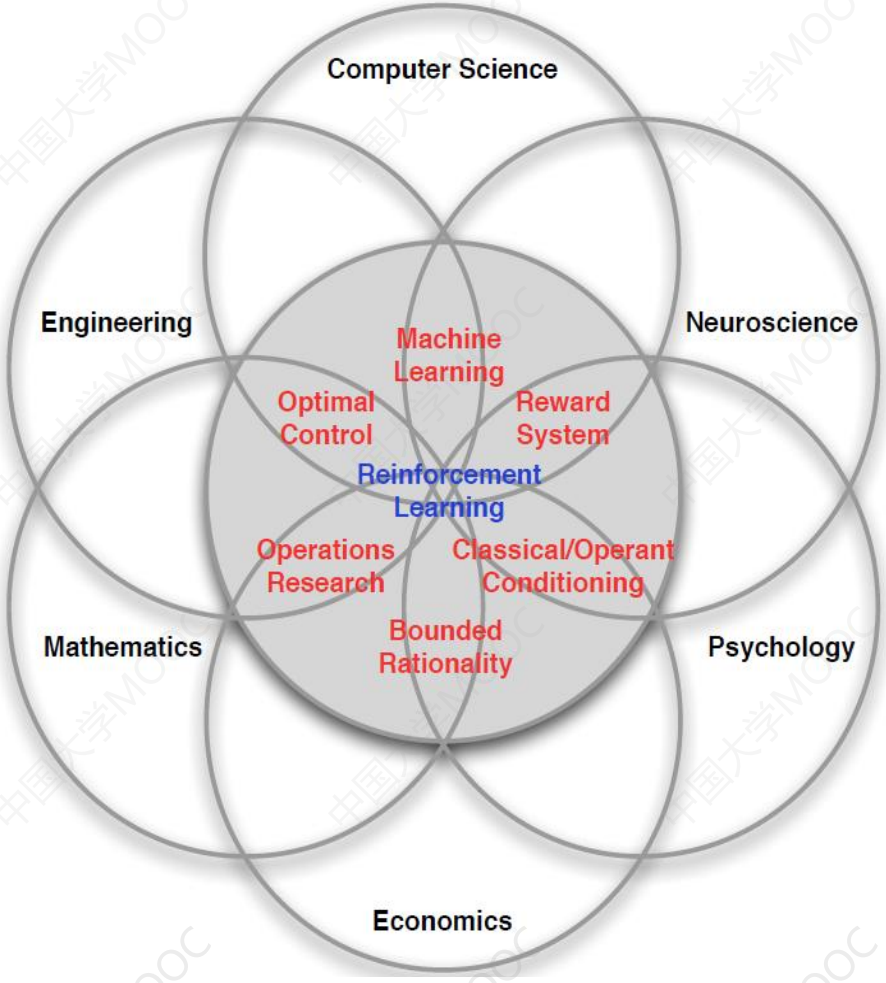
机器学习三大分支：

- 无监督学习
- 监督学习
- 强化学习



一、强化学习概述

- 强化学习技术是从控制理论、统计学、心理学等相关学科发展而来。
- 在人工智能、机器学习和自动控制等领域中得到广泛研究和应用，并被认为是设计智能系统的核心技术之一。
- 随着强化学习的数学基础研究取得突破性进展后，强化学习成为机器学习领域研究热点之一。



一、强化学习概述

强化学习发展历史

- 1954年Minsky首次提出“强化”和“强化学习”的概念。
- 1953到1957年，Bellman提出了求解最优控制问题的一个有效方法---动态规划，同年，还提出了最优控制问题的随机离散版本，就是著名的马尔可夫决策过程
- 1960年Howard提出马尔可夫决策过程的策略迭代方法，这些都成为现代强化学习的理论基础。
- 1972年，Klopf把试错学习和时序差分结合在一起。
- 1988年 Sutton提出了TD算法
- 1989年 Watkins提出了Q学习算法
- 1994年 Rummery等提出了SARSA学习算法
- 2015 Google DeepMind公司提出了深度强化学习DRL

一、强化学习概述

强化学习的特点：

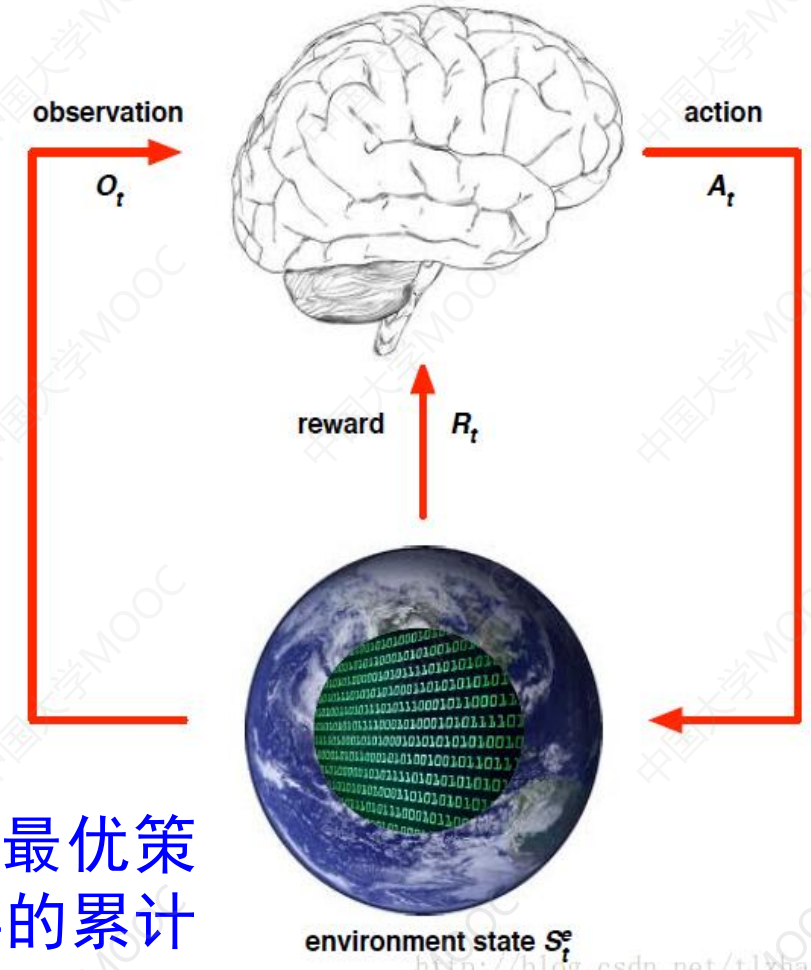
强化学习围绕着如何与环境交互学习，在行动—评价的环境中获得改进的行动方案，以适应环境达到预想的目的。学习者并不会被告知采取哪个动作，而只能通过尝试每个动作，获得环境对所采取动作的反馈信息，从而指导以后的行动。因此，强化学习主要特点包括：

- 试错搜索： Agent通过尝试多个动作，搜索最优策略；
- 延迟回报：其反馈信号是延迟的而非瞬间的；
- 适应性：Agent不断利用环境中的反馈信息来改善其性能；
- 不依赖外部教师信号：因为Agent只根据反馈信号进行学习，因此不需要外部教师信号。

一、强化学习概述

强化学习基本模型

- 在强化学习中，Agent 选择一个动作a作用于环境；
- 环境接收该动作后发生变化，同时产生一个强化信号Reward（奖或罚）反馈给Agent；
- Agent再根据强化信号和环境的当前状态s再选择下一个动作，选择的原则是使受到奖赏值的概率增大。



强化学习的目的就是寻找一个最优策略，使得Agent在运行中所获得的累计期望回报最大。

一、强化学习概述

分类

郭宪等，深入浅出强化学习入门，电子工业出版社，2018

从广义上讲，强化学习是解决序贯决策问题的方法之一，将强化学习纳入马尔科夫决策过程的框架后，可以分为基于模型的动态规划方法和基于无模型的强化学习方法。

二、有模型学习

定义：在已知模型的环境中学习，称为“有模型学习”，也即，对于多步强化学习任务，其对应的马尔可夫决策过程四元组表示<S, A, R, P>均为已知，称为“模型已知”。

- S：环境的状态空间
- A：agent可选择的动作空间
- R(s, a)：奖励函数，返回的值表示在状态下执行a动作的奖励
- P(s'|s, a)：状态转移概率函数，表示从s状态执行a动作后环境转移至s′状态的概率

目标：找到一个策略π能够最大化我们的对未来奖励的期望$E(\sum_{t=0}^{n}\gamma^t R_t)$，$R_t$为$t$时刻的奖励，$\gamma$为折扣因子，代表距离现在遥远的奖励不如现在的奖励大。

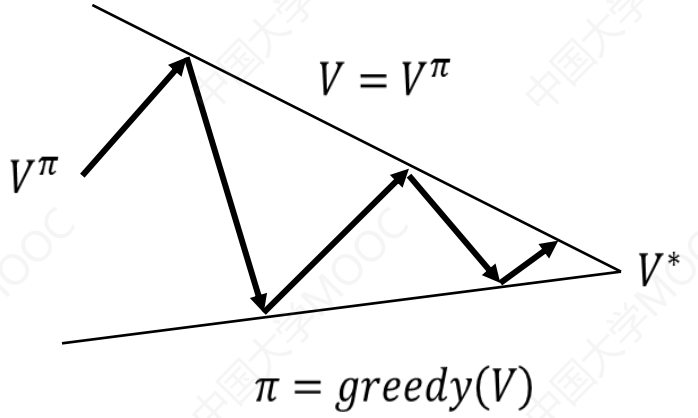
二、有模型学习

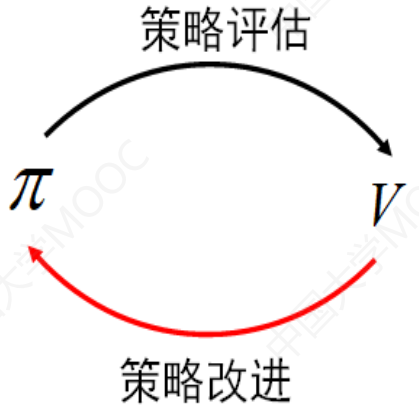
策略迭代算法---流程

① 某一个随机策略作为初始策略

② 策略评价+策略改进+策略评价+策略改进+……

③ 若满足收敛条件，则退出，否则，转入②





策略迭代算法的缺点在于：每次改进策略后都需要重新进行策略评价，计算比较耗时。

二、有模型学习

策略迭代算法---策略评价

[1] 输入：需要评估的策略 π 状态转移概率 $P_{ss'}^{a}$ 回报函数 R_s^a ，折扣因子 γ

[2] 初始化值函数：$V(s)=0$

一次状态扫描

[3] Repeat k=0,1,…

[4] for every s do

[5] $$v_{k+1}(s)=\sum_{a\in A}\pi(a|s)\left(R_s^a+\gamma\sum_{s'\in S}P_{ss'}^{a}v_k(s')\right)$$

[6] end for

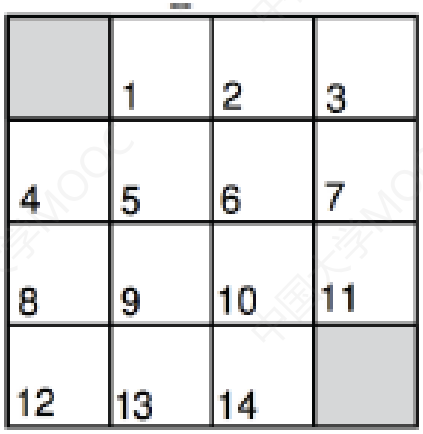
[7] Until $v_{k+1}=v_k$

[8] 输出：$v(s)$

郭宪等，深入浅出强化学习入门，电子工业出版社，2018

二、有模型学习

策略迭代算法
——策略评价举例



https://blog.csdn.net/liweibin1994/article/details/79093453

- 即时奖励：左图是一个九宫格，左上角和右下角是终点，它们的reward是0，其他的状态reward都是-1。
- 状态空间：除了灰色两个格子，其他都是非终点状态
- 动作空间：在每个状态下，都有四种动作可以执行，分别是上下左右(东西南北)。
- 转移概率：任何想要离开大正方形的动作将保持其状态不变，也就是原地不动。其他时候都是直接移动到下一个状态。所以状态转移概率是确定性的。
- 折扣因子：$\gamma=1$
- 当前策略：在任何状态下，agent都采取均匀随机策略，也就是它的动作是随机选择的，即：

$$\pi(e|*)=\pi(w|*)=\pi(s|*)=\pi(n|*)=0.25$$

问题：评价均匀随机策略。也就是说，求解均匀随机策略下所有状态的V值

二、有模型学习

策略迭代算法---策略评价举例

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=0

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=1

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

K=2

Repeat k=0,1,…

for every s do

$$v_{k+1}(s) = \sum_{a \in A} \pi(a|s) \left(R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a v_k(s') \right)$$

Until $v_{k+1} = v_k$

二、有模型学习

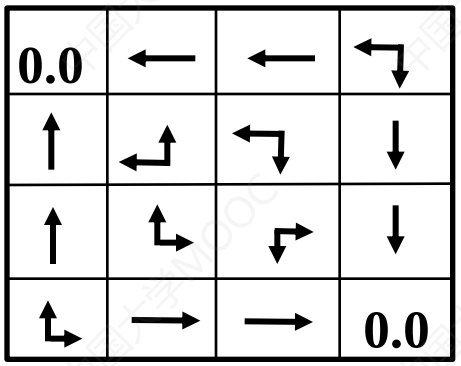
策略迭代算法---策略改进：在每个状态采用贪心策略，从而找到更好的策略。

$$\pi_{l+1}(s) \in \arg\max_{a} q^{\pi_l}(s,a)$$

K=10

π_0 均匀策略

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



π_1 贪心策略

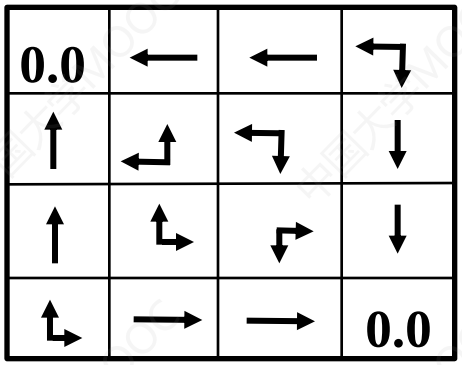
二、有模型学习

值迭代算法：基于策略迭代的方法是交替进行策略评价和策略改进，其中策略评价中需要迭代多次，以保证当前策略评价收敛。因此，算法收敛较慢。为了解决该问题，提出了值迭代算法。

K=10

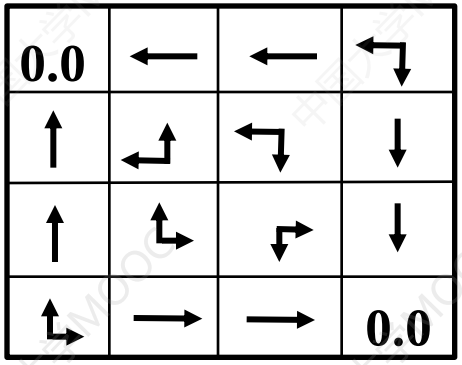
π_0 均匀策略

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



$K = \infty$

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



π_1 贪婪策略

策略改进不一定要等到值函数收敛！

二、有模型学习

值迭代算法：

[1] 输入：状态转移概率 $P_{ss'}^{a}$ 回报函数 R_s^a ，折扣因子 γ

初始化值函数：$v(s)=0$　初始化策略 π_0

[2] Repeat l=0,1,…

[3] for every s do

[4] $$v_{l+1}(s)=\max_a R_s^a+\gamma\sum_{s'\in S}P_{ss'}^{a}v_l(s')$$

[5] Until $v_{l+1}=v_l$

[6] 输出：$$\pi(s)=\operatorname{argmax}_a R_s^a+\gamma\sum_{s'\in S}P_{ss'}^{a}v_l(s')$$

三、无模型学习

- 当模型未知，即状态转移概率、奖赏函数往往我们是不知道的，甚至很难知道环境中一共有多少状态。此时我们无法直接利用Bellman方程来求解得到最优策略。

- 若学习算法不依赖环境建模，则称为“无模型学习”，或称模型无关的学习（Model-free Learning）。

- 模型无关的强化学习，是在不知道马尔科夫决策过程的情况下学习到最优策略。模型无关的策略学习主要有两种算法:蒙塔卡洛强化学习，时序差分强化学习。而时序差分强化学习又包括SARSA 和 Q-learning两种算法。

三、无模型学习

蒙特卡洛采样

- MDP是通过5元组：<S,P,A,R,γ>来做决策的。对于这种已知模型的情况，也就是知道了这个5元组，我们可以通过求解贝尔曼方程获得奖赏最大化。
- 但是，在现实世界中，我们无法同时知道这个5元组。比如状态转移概率就很难知道，我们无法使用bellman方程来求解V和Q值。
- 一个想法是，虽然我不知道状态转移概率P，但是这个概率是真实存在的。我们可以直接去尝试，不断采样，然后会得到奖赏，通过奖赏来评价值函数。

三、无模型学习

同策略

蒙特卡洛强化学习

周志华 著. 机器学习，北京：清华大学出版社，2016年1月，pp:384

输入： 环境 E;
动作空间 A;
起始状态 x_0;
策略执行步数 T.

过程：

$Q(x,a)=0,\ \mathrm{count}(x,a)=0,\ \pi(x,a)=\frac{1}{|A(x)|}$;
for $s=1,2,\ldots$ **do**
  在 E 中执行策略 π 产生轨迹
  $< x_0,a_0,r_1,x_1,a_1,r_2,\ldots,x_{T-1},a_{T-1},r_T,x_T >$;
  for $t=0,1,\ldots,T-1$ **do**
    $R=\frac{1}{T-t}\sum_{i=t+1}^{T} r_i$;
    $Q(x_t,a_t)=\frac{Q(x_t,a_t)\times\mathrm{count}(x_t,a_t)+R}{\mathrm{count}(x_t,a_t)+1}$;
    $\mathrm{count}(x_t,a_t)=\mathrm{count}(x_t,a_t)+1$
  end for
  对所有已见状态 x :

$$\pi(x,a)=\begin{cases}\arg\max_{a'} Q(x,a'), & \text{以概率 } 1-\epsilon;\\ \text{以均匀概率从 } A \text{ 中选取动作}, & \text{以概率 } \epsilon\ .\end{cases}$$

end for
输出： 策略 π

图 16.10　同策略蒙特卡罗强化学习算法

三、无模型学习

异策略

蒙特卡洛强化学习

周志华 著. 机器学习，北京：清华大学出版社，2016年1月，pp:386

输入： 环境 E;
动作空间 A;
起始状态 x_0;
策略执行步数 T.

过程：

$Q(x,a)=0,\ \mathrm{count}(x,a)=0,\ \pi(x,a)=\frac{1}{|A(x)|}$;
for $s=1,2,\ldots$ **do**
  在 E 中执行 π 的 ϵ-贪心策略产生轨迹
  $< x_0, a_0, r_1, x_1, a_1, r_2, \ldots, x_{T-1}, a_{T-1}, r_T, x_T >$;
  $p_i = \begin{cases} 1-\epsilon+\epsilon/|A|, & a_i = \pi(x); \\ \epsilon/|A|, & a_i \neq \pi(x), \end{cases}$
  for $t=0,1,\ldots,T-1$ **do**
    $R=\frac{1}{T-t}\sum_{i=t+1}^{T}(r_i \times \prod_{j=i}^{T-1}\frac{1}{p_j})$;
    $Q(x_t,a_t)=\frac{Q(x_t,a_t)\times \mathrm{count}(x_t,a_t)+R}{\mathrm{count}(x_t,a_t)+1}$;
    $\mathrm{count}(x_t,a_t)=\mathrm{count}(x_t,a_t)+1$
  end for
  $\pi(x)=\arg\max_{a'} Q(x,a')$
end for

输出： 策略 π

图 16.11　异策略蒙特卡罗强化学习算法

三、无模型学习

时序差分强化学习（Temporal-Difference）

在蒙特卡洛学习中，却需要一条完整的轨迹，才能估计某个状态—动作值函数，从而进行更新，导致了算法效率低下。

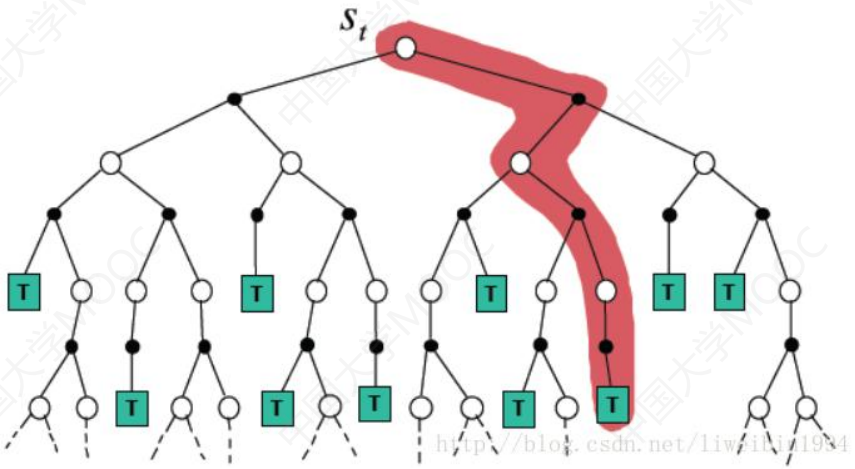
$$Q_t^{\pi}(x,a) = \frac{1}{t}\sum_{i=1}^{t} r_i$$

在时序差分学习中，算法在每执行一步策略后就进行值函数的更新，因此效率较高。

$$Q_{t+1}^{\pi}(x,a) = Q_t^{\pi}(x,a) + \frac{1}{t+1}\left(r_{t+1} - Q_t^{\pi}(x,a)\right)$$

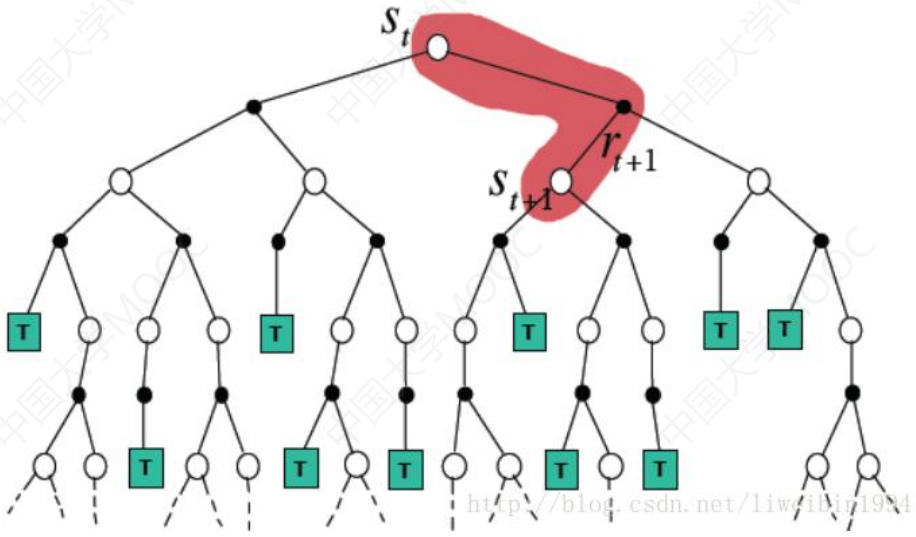
三、无模型学习

$$V(S_t) \leftarrow V(S_t) + \alpha(G_t - V(S_t))$$



蒙特卡洛强化学习

$$V(S_t) \leftarrow V(S_t) + \alpha(R_{t+1} + \gamma V(S_{t+1}) - V(S_t))$$



时序差分强化学习

时序差分方法：分为同策略的Sarsa和异策略的Q-learning

三、无模型学习

Sarsa算法

Initialize $Q(s,a)$ arbitrarily
Repeat (for each episode):
  Initialize S
  Choose A from S using policy derived from Q (e.g., ε-greedy)
  Repeat (for each step of episode):
    Take action A, observe R, S'
    Choose A' from S' using policy derived from Q (e.g., ε-greedy)
    $Q(S,A) \leftarrow Q(S,A) + \alpha[R + \gamma Q(S',A') - Q(S,A)]$
    $S \leftarrow S'$; $A \leftarrow A'$;
  until S is terminal

在Sarsa算法中，选择动作时遵循的策略和更新动作值函数时遵循的策略是相同的，均为ϵ-贪心策略

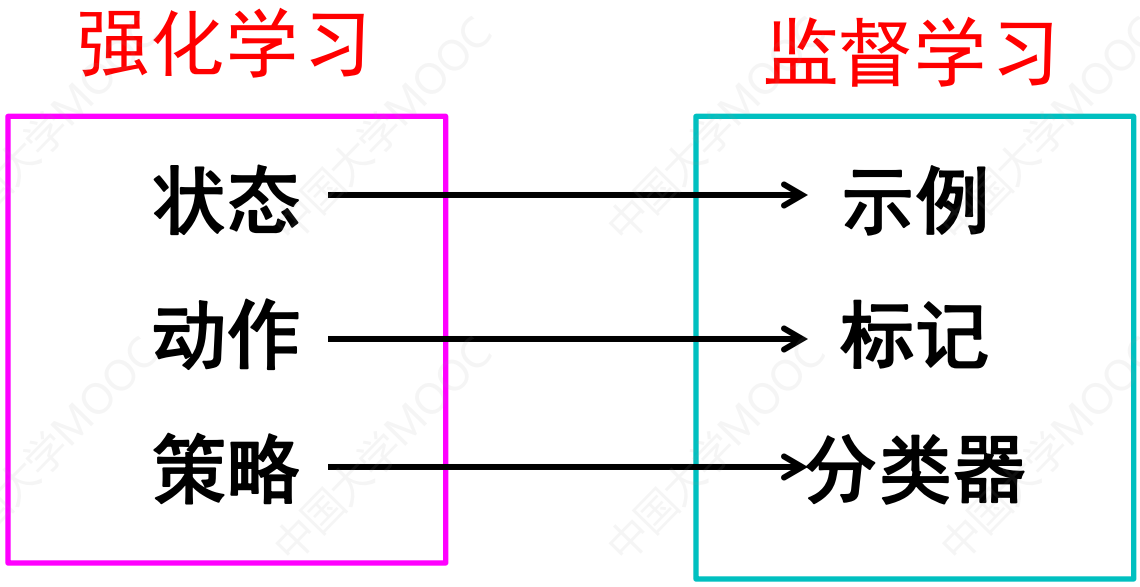
三、无模型学习

Q-learning算法

Initialize $Q(s,a)$ arbitrarily
Repeat (for each episode):
  Initialize S
  Repeat (for each step of episode):
    Choose A from S using policy derived from Q (e.g., ε-greedy)
    Take action A, observe R, S'
    $Q(S,A) \leftarrow Q(S,A) + \alpha[R + \gamma \max_a Q(S',a) - Q(S,A)]$
    $S \leftarrow S'$;
  until S is terminal

在Q-learning算法中，选择动作时遵循的策略是ϵ-贪心策略，更新动作值函数时，直接使用了最大的$Q(S_{t+1},a)$

四、对强化学习的理解



强化学习在某种意义上可看作具有“延迟标记信息”的监督学习问题。

四、对强化学习的理解

强化学习可以分为基于模型的方法与无模型的方法。前者发展主要来自最优控制领域。而后者发展更多的来自机器学习领域。无模型的强化学习算法通过大量采样，估计智能体的状态--动作值函数或回报函数，从而获得最优的策略。

但是，无模型的强化学习可能面临的一些问题：

① 奖励函数难以设计，缺乏理论指导。

② 不对具体问题进行建模，而是尝试用一个通用的算法解决所有问题，没有利用问题固有的信息。

③ 因为没有模型，解释性不强，调试困难。

……

四、对强化学习的理解

- 深度学习（DL）技术和强化学习（RL）的结合，形成了深度强化学习（DRL），迅速成为人工智能界的焦点。
- 在视频游戏、棋类游戏、机器人控制等领域取得了巨大成功。

可能面临的问题：

① 难以平衡“探索”和“利用”，以致算法陷入局部极小；

② 样本利用较低；

③ 对环境容易出现过拟合；

④ 灾难性的不稳定性。

……

四、对强化学习的理解

潜在的研究方向包括：

- 提高无模型方法的数据利用率和扩展性；
- 设计高效的探索策略。平衡“探索”与“利用”；
- 与模仿学习结合，既能更快地得到反馈、又能更快地收敛；
- 探索好的奖励机制。奖励机制对强化学习算法性能的影响是巨大的，因此该方向一直是强化学习的研究热点。
- 混合迁移学习和多任务学习。当前强化的采样效率较低，而且学到的知识不通用，迁移学习与多任务学习可以有效解决这些问题。

……