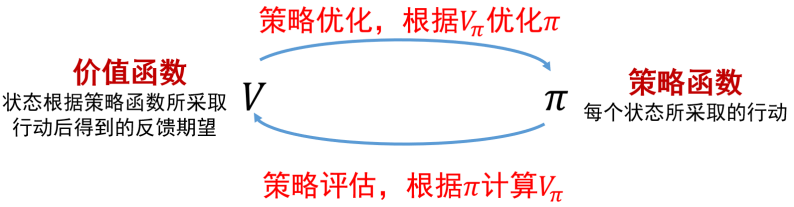
知识点总结（强化学习）

- 强化学习：在与环境交互之中进行学习
- 强化学习中的概念
  - 智能主体
    - 按照某种策略(policy)，根据当前的状态(state)选择合适的动作
    - 状态指的是智能主体对环境的一种解释
    - 动作反映了智能主体对环境主观能动的影响，动作带来的收益称为奖励
    - 智能主体可能知道也可能不知道环境变化的规律
  - 环境
    - 系统中智能主体以外的部分
    - 向智能主体反馈状态和奖励
    - 按照一定的规律发生变化
  - 基于评估： 强化学习利用环境评估当前策略，以此为依据进行优化
  - 交互性： 强化学习的数据在与环境的交互中产生
  - 序列决策过程： 智能体在与环境的交互中需要作出一系列的决策，这些决策往往是前后关联的
  - 现实中常见的强化学习问题往往还具有奖励滞后，基于采样的评估等特点
  - 马尔可夫链
    - 马尔可夫随机过程,为了在序列决策中对目标进行优化，加入奖励机制：
      - 奖励函数$R:S\times S\mapsto R$，其中$R(S_t,S_(t+1))$描述了从第⊠步状态转移到第$\boldsymbol{t}+\mathbf{1}$步状态所获得奖励，简记为$R_(t+1)$
      - 不同状态之间的转移产生了一系列的奖励$(R_1,R_2,\cdots)$
      - 引入奖励机制可以衡量任意序列的优劣，即对决策进行评价
    - 马尔可夫奖励过程,为了比较不同的奖励序列，定义反馈，用来反映累加奖励：
      - $G_t=R_{t+1}+\gamma R_{t+2}+\gamma^2R_{t+3}+\cdots$
      - 其中折扣系数(discountfactor)⊠∈[⊠, ⊠]，假设$\gamma=0.99$
      - 反馈值反映了某时刻后的累加奖励，"$\gamma<1$"时，未来对累加反馈的贡献越少
      - 该模型不能体现机器人能动性，仍缺乏与环境交互的手段
    - 马尔可夫决策过程：引入动作
      - 定义智能主体能够采取的动作集合为A
      - 由于不同的动作对环境造成的影响不同，因此状态转移概率定义为$Pr(S_{t+1}\mid S_t,a_t)$，其中$a_t\in A$为第t步采取的动作
      - 奖励可能受动作的影响，因此修改奖励函数为$R(S_t,a_t,S_{t+1})$

        - 动作集合A可以是有限的，也可以是无限的
        - 状态转移可是确定(deterministic)的，也可以是随机概率性(stochastic)的。
        - 确定状态转移相当于发生从S_t到S_{t+1}的转移概率为1
    - 使用离散马尔可夫过程描述机器人移动问题
        - 随机变量序列$S_{tt=0,1,2,\cdots}$ ： S_t表示机器人第t步所在位置(状态)，每个随机变量S_t的取值范围为$S=\{s_1,s_2,\cdots,s_9,s_d\}$
        - 动作集合：$A=$ 上，右
        - 状态转移概率$Pr(S_{t+1}|S_t,a_t)$： 满足马尔可夫性，其中$a_t\in A$。
        - 奖励函数：$R(S_t,a_t,S_{t+1})$
        - 衰退系数：$\gamma\in[0,1]$
    - 可通过$MDP=\{S,A,Pr,R,\gamma\}$来刻画马尔科夫决策过程
    - 马尔科夫过程中产生的状态序列称为轨迹(trajectory)，可如下表示$(S_0,a_0,R_1,S_1,a_1,R_2,\cdots,S_T)$
        - 轨迹长度可以是无限的，也可以有终止状态S_T 。有终止状态的问题叫做分段的(即存在回合的)，否则叫做持续的
        - 分段问题中，一个从初始状态到终止状态的完整轨迹称为一个片段或回合(episode)。如围棋对弈中一个胜败对局为一个回合。
    - 策略函数：
        - 策略函数$\pi:S\times A\mapsto[0,1]$，其中$\pi(s,a)$的值表示在状态$s$下采取动作$a$的概率。
        - 策略函数的输出可以是确定的，即给定s情况下，只有一个动作a使得概率$\pi(s,a)$取值为1。对于确定的策略，记为$a=\pi(s)$。
    - 为了对策略函数π进行评估，定义
        - 价值函数(Value Function)$V:S\mapsto R$，其中$V_\pi(s)=E_\pi[G_t|S_t=s]$，即在第t步状态为s时，按照策略π行动后在未来所获得反馈值的期望
        - 动作-价值函数(Action-Value Function)$q:S\times A\mapsto R$，其中$q_\pi(s,a)=E_\pi[G_t|S_t=s,A_t=a]$表示在第$t$步状态为$s$时，按照策略$\pi$采取动作$a$后，在未来所获得反馈值的期望
    - 寻找一个最优策略π^*，对任意$s\in S$使得$V_{\pi^*}(s)$值最大
- 给定一个马尔可夫决策过程$MDP=(S,A,P,R,\gamma)$，学习一个最优策略π^* ，对任意$s\in S$使得$V_(\pi^*)(s)$值最大。
    - 价值函数和动作-价值函数反映了智能体在某一策略下所对应状态序列获得回报的期望，它比回报本身更加准确地刻画了智能体的目标。
- 贝尔曼方程
    - 价值函数(Value Function)$V_\pi(s)=E_\pi[R_{t+1}+\gamma R_{t+2}+\gamma^2R_{t+3}+\cdots\Big|S_t=s]$
    - 动作-价值函数(Action-Value Function)$q_\pi(s,a)=E_\pi[R_{t+1}+\gamma R_{t+2}+\gamma^2R_{t+3}+\cdots\Big|S_t=s,A_t=a]$

- 贝尔曼方程描述了价值函数或动作-价值函数的递推关系，是研究强化学习问题的重要手段。

- 基于价值的强化学习
  - 强化学习会寻找一个最优策略π^*，在策略π^*作用下使得任意状态$s \in S$对应的价值函数$V_{\pi^*}(s)$取值最大。



  - 给定当前策略π、价值函数V_π和行动-价值函数q_π时，可如下构造新的策略π'，只要π'满足如下条件：π'便是对π的一个改进

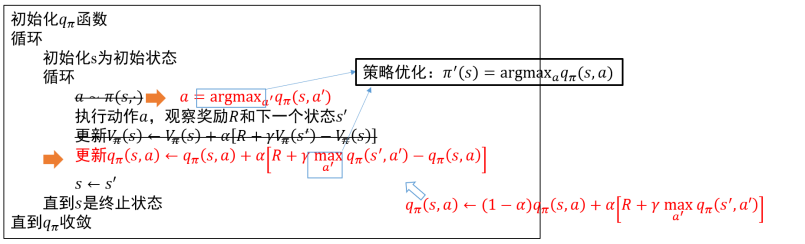
$$\pi'(s) = \mathrm{argmax}_a q_\pi(s,a) \text{ (对于任意} s \in S\text{)}$$

- 强化学习中的策略评估方法
  - 动态规划
    - $V_\pi(s) \leftarrow \sum_{a\in A} \pi(s,a) \sum_{s'\in S} Pr(s'|s,a)[R(s,a,s') + \gamma V_\pi(s')]$
    - 动态规划法的缺点：
      - 智能主体需要事先知道状态转移概率;
      - 无法处理状态集合大小无限的情况
  - 蒙特卡洛采样
    - 选择不同的起始状态，按照当前策略π采样若干轨迹，记它们的集合为D枚举$s \in S$计算D中s每次出现时对应的反馈$G_1, G_2, \cdots, G_k$
    - $V_\pi(s) \leftarrow \frac{1}{k} \sum_{i=1}^{k} G_i$
    - 蒙特卡洛采样法的优点
      - 智能主体不必知道状态转移概率
      - 容易扩展到无限状态集合的问题中
    - 蒙特卡洛采样法的缺点
      - 状态集合比较大时，一个状态在轨迹可能非常稀疏，不利于估计期望
      - 在实际问题中，最终反馈需要在终止状态才能知晓，导致反馈周期较长
  - 时序差分
    - 由于通过采样进行计算，所得结果可能不准确，因此时序差分法并没有将这个估计值照单全收，而是以α作为权重来接受新的估计值，
    - 即把价值函数更新为$(1-\alpha)V_\pi(s) + \alpha[R + \gamma V_\pi(s')]$
    - 对这个式子稍加整理就能得到算法7.2.3中第7行形式：$V_\pi(s) \leftarrow V_\pi(s) + \alpha[R + \gamma V_\pi(s') - V_\pi(s)]$。
    - 这里$R + \gamma V_\pi(s')$为时序差分目标，$R + \gamma V_\pi(s') - V_\pi(s)$为时序差分偏差。

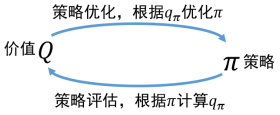
更新$V_\pi(s)$的值：$V_\pi(s) \leftarrow (1-\alpha)V_\pi(s) + \alpha[R(s,a,s') + \gamma V_\pi(s')]$

过去的价值函数值　　学习得到的价值函数值

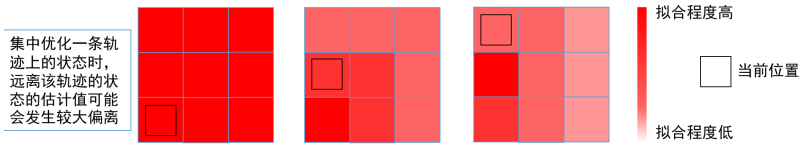
- 基于时序差分的方法 – Q学习(Q-Learning)[Q: quality]



- 基于价值的方法不直接对策略建模，因此策略优化在采样和更新两步中之max操作上得以间接体现
- 在同一次循环中策略评估和策略优化交替进行
- 由于策略优化要求计算动作-价值函数q，因此Q学习直接利用q函数的贝尔曼方程进行更新



- $q_\pi(s,a) \leftarrow q_\pi(s,a) + \alpha[R + \gamma max_{a'} q_\pi(s',a') - q_\pi(s,a)]$
- 基于价值的方法不直接对策略建模，因此策略优化在采样和更新两步中之max操作上得以间接体现
- 根据目前已知的最优策略来选择动作，被称为利用(exploitation)
- 不根据当前策略而去尝试未知的动作被称为探索(exploration)
- 使用ε贪心策略的Q学习
- 用神经网络拟合(行动)价值函数：Deep Q-learning
  - 状态数量太多时，有些状态可能始终无法采样到，因此对这些状态的q函数进行估计是很困难的
  - 状态数量无限时，不可能用一张表(数组)来记录q函数的值
  - 将q函数参数化(parametrize)，用一个非线性回归模型来拟合q函数，例如(深度)神经网络
    - 能够用有限的参数刻画无限的状态
    - 由于回归函数的连续性，没有探索过的状态也可通过周围的状态来估计
- 深度Q学习的两个不稳定因素
  - 相邻的样本来自同一条轨迹，样本之间相关性太强，集中优化相关性强的样本可能导致神经网络在其他样本上效果下降。



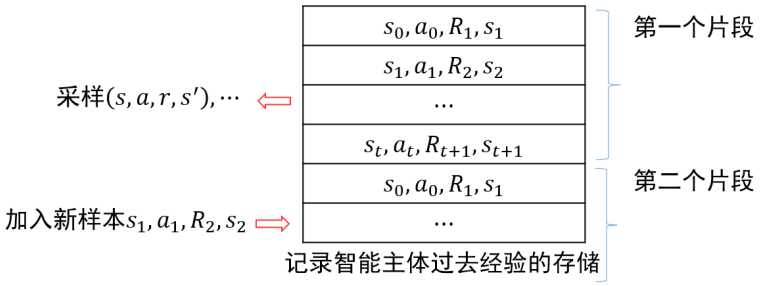
  - 在损失函数中，q函数的值既用来估计目标值，又用来计算当前值。现在这两处的q函数通过θ有所关联，可能导致优化时不稳定

$$\frac{1}{2}\left[R + \gamma \max_{a'} q_\pi(s',a';\theta) - q_\pi(s,a;\theta)\right]^2$$

预测值　　当前值

- 经验重现

- 将过去的经验存储下来，每次将新的样本加入到存储中去，并从存储中采样一批样本进行优化



  - 解决了样本相关性强的问题
  - 重用经验，提高了信息利用的效率
- 目标网络
  - 在损失函数中，q函数的值既用来估计目标值，又用来计算当前值。现在这两处的q函数通过θ有所关联，可能导致优化时不稳定
  - 损失函数的两个q函数使用不同的参数计算
    - 用于计算估计值的q使用参数θ^-计算，这个网络叫做目标网络
    - 用于计算当前值的q使用参数θ计算
    - 保持θ^-的值相对稳定，例如θ每更新多次后才同步两者的值
    - $\theta^- \leftarrow \theta$
- 基于策略的强化学习
  - 基于价值的强化学习:以对价值函数或动作-价值函数的建模为核心。
  - 基于策略的强化学习:直接参数化策略函数，求解参数化的策略函数的梯度。
  - 策略函数的参数化可以表示为⊠_⊠ (⊠,⊠)，其中⊠为一组参数，函数取值表示在状态⊠下选择动作⊠的概率。和Q学习的⊠贪心策略相比，这种参数化的一个显著好处是：选择一个动作的概率是随着参数的改变而光滑变化的，实际上这种光滑性对算法收敛有更好的保证。
  - **⊠-贪心** 像是一个只有几个预设频道的收音机，你只能在这些频道间切换，或者随机跳到一个频道。如果最佳频道在两个预设频道之间，你很难精确调到。
  - 参数化的策略 ⊠_⊠像是一个可以平滑旋转调谐的收音机旋钮。你可以非常细微地调整旋钮（参数 ⊠），使得接收到的信号（策略性能）也平滑地变化，从而更容易精确地找到信号最好的那个点（最优策略）。