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

人工智能讲义

强化学习

April 25, 2018

Outline

1. 引入问题
2. 强化学习
3. 更新参数
4. 选择行动
5. 函数逼近

MDP/马尔科夫决策过程

- **S**: 状态空间
- 初态：s_0
- 行动：$Action(s)$，给定状态 $s \in \mathbf{S}$，合法行动集合
- 状态转移概率：$T(s, a, s')$，从状态 s 出发，采用行动 a，导致结果状态 s' 的概率
- 奖励：$Reward(s, a, s')$，状态转移 (s, a, s') 得到的收益
- 目标测试：$isEnd(s)$
- 折扣因子 λ

如果描述不完备，会怎样?

MDP/马尔科夫决策过程

- $\mathbf{S}$: 状态空间
- 初态：s_0
- 行动：$Action(s)$，给定状态 $s \in \mathbf{S}$，合法行动集合
- 状态转移概率：$T(s, a, s')$，从状态 s 出发，采用行动 a，导致结果状态 s' 的概率
- 奖励：$Reward(s, a, s')$，状态转移 (s, a, s') 得到的收益
- 目标测试：$isEnd(s)$
- 折扣因子 λ

如何找到缺失的部分?

- 通常现实应用中我们并不知道状态转移概率和奖励的细节。
- 强化学习!

问题描述

- 老虎机（one-arm bandit）是一种用零钱赌博的机器，因为上面有老虎图案的筹码而得名。
- 老虎机有三个玻璃框，里面有不同的图案，投币之后拉下拉杆，就会开始转，如果出现特定的图形（比如三个相同）就会吐钱出来，出现相同图型越多奖金则越高。
- 1895 年——查理·费 (Charlie Fey ,Jzplay Com) 发明第一台商业老虎机。
- 若有两台老虎机 A 或 B，需要投币进行游戏，游戏返回奖励的金额和概率是不知道的，可能是随机的结果；
- 假设赌客可以连续有选择地投币，进行重复游戏。
- 赌客如何选择才能最大获益?

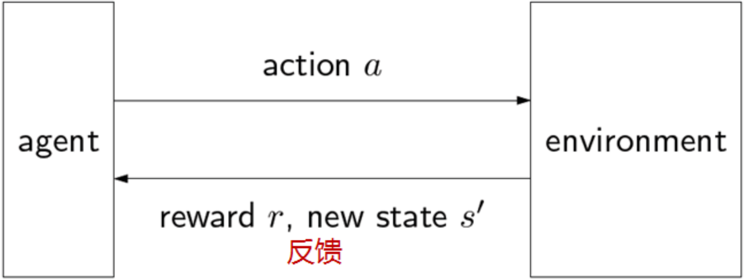
MDP 与强化学习

- MDP：存在一个上帝，知道了所有的转移概率和奖励情况，寻找最优策略；称为“离线”决策
- MDP：所有的决策判断过程都可以在头脑中“虚拟一遍”/仿真一次；
- MDP：f 已知
- 强化学习：没有人知道所有的转移概率和奖励情况，只能看到部分情况，寻找最优策略；称为“在线”决策
- 强化学习：需要花费代价去尝试或“探索”未知的情况（转移概率和奖励），然后逐步调整策略。
- 强化学习：f 未知，在寻找最优策略的过程中，逐步了解/完善 f

思考

- 电脑游戏“高手”和“低手”之间的差别在哪儿?
- 玩牌，打麻将的水平差异在哪儿?
- 招聘时工作经历的作用是什么?

强化学习/Reinforcement Learning



强化学习的例子：生活经验

- 生活中的任何一个行动，会得到好的/不好的收益
- 人会从行动/收益中汲取经验教训，调整自己今后的行动
- 人的一生在不停地行动、收集反馈、学习、行动、收集反馈、……

问题描述

- 已知：序列 $s_0, a_1, r_1, s_1, a_2, r_2, \ldots, a_n, r_n, s_n$，其中 $s_i, i = 0, 1, \ldots, n$ 表示状态，$a_i, i = 1, 2, \ldots, n$ 表示行动，$r_i, i = 1, 2, \ldots, n$ 表示奖励
- 求解：给每个状态确定一个“最佳行动”，即找到最优策略

分析与理解

- 与 MDP 的差异在于已知条件
  - 强化学习，已知样本数据序列，可能不止一个序列；
  - MDP，已知转移概率和奖励的全部信息。
- 强化学习和 MDP 所要求目标是一致的。

算法框架

- for $t = 1, 2, \ldots$
  - 选择行动 $a_t = \pi(s_{t-1})$
  - 收集反馈奖励 r_t，获得新状态 s_t
  - 更新参数

分析与理解

- 通用框架，解释了什么是强化学习。类似于机器学习中的增量式学习、在线学习。
- 选择行动 $a_t = \pi(s_{t-1})$，$\pi(\cdot)$ 从何而来?
- 更新参数，参数是什么？怎么更新?

强化学习：参数及其更新 思想： 强化学习较之于 MDP，就少了转移概率和奖励，那么想方法把转移概率和奖励计算出来，问题得解。

- 已知： $s_0, a_1, r_1, s_1, a_2, r_2, \ldots, a_n, r_n, s_n$
- 求参数： $T(s, a, s')$ 和 $U(s, a, s')$

蒙特卡洛方法：出现频率代替概率

- 从已知数据中任何状态 s 开始，(s, a, r, s') 视为 “一个/一段/一组数据”，原数据序列被分割成 n 段；
- 计数，并计算 $\hat{T}(s, a, s') = \frac{\#(s,a,s')}{\#(s,a)}$, $\hat{U}(s, a, s') = \frac{\sum_{r \ \ in \ \ (s,a,r,s')} r}{\#(s,a,s')}$
- 用 $\hat{T}(s, a, s')$ 近似估计 $T(s, a, s')$, $\hat{U}(s, a, s')$ 近似估计 $U(s, a, s')$

例子：

- 已知数据： $s_1, A, 3, s_2, B, 0, s_1, A, 5, s_1, A, 7, s_1$
- 估计参数/模型： $\hat{T}(s_1, A, s_1) = 2/3, \hat{U}(s_1, A, s_1) = (5+7)/2 = 6$

存在问题

- 样本数据是否满足独立性假设；
- 很多状态的行动没有数据，或者说很多计数是 0
- 样本数据量要求大
- 这一类方法称为：基于模型的蒙特卡洛方法。所谓模型，就是所有的转移概率和奖励构成的集合。

不求模型，可以吗？

$$V_\pi(s) = \sum_{s' \in \mathbf{S}} T(s,a,s')[U(s,a,s') + \lambda V_\pi(s')]$$

来自同一策略的已知数据

- 若所有已知数据来自同一策略 π，能否求得策略 π 的价值/value?
- 当 $T(\cdot), U(\cdot)$ 未知时，用基于模型的蒙特卡洛方法估计其值，无法获得“完美”描述的 MDP，因为很多 (状态, 行动) 对没有出现;

执行策略 π，得到一条随机路径

- 已知数据：$s_0, a_1, r_1, s_1, a_2, r_2, \dots, a_n, r_n, s_n$
- 定义引入折扣因子的时刻 t 的收益：$u_t = r_t + \lambda r_{t+1} + \lambda^2 r_{t+2} + \dots$
- 用 u_t 的均值估计/当成 $Q_\pi(s,a)$：即将 u_t 按 (s,a) 不同取值分组，然后求组内均值。

$$Q_\pi(s,a) = 0, \text{ if } isEnd(s) == T$$
$$Q_\pi(s,a) = V_\pi(s), \text{ if } isEnd(s) == F$$

例子

- 已知数据：$s_1, A, 3, s_2, B, 0, s_1, A, 5, s_1, A, 7, s_1$
- $Q_\pi(s_1, A) = (u_1 + u_3 + u_4)/3 = (15 + 12 + 7)/3 = 34/3$

等价的算法描述：增量式地计算

- 对任意时刻 t，对应数据段 (s, a, r):
  - 计算出 (s, a, u_t)
  - 令：$\xi = \frac{1}{(s,a)\text{更新次数}+1}$
  - 更新：$Q_\pi(s, a) = (1 - \xi) Q_\pi(s, a) + \xi u_t$

注意体会和理解 Q_π 的更新计算过程

没用到模型相关的东西！所以称为“模型无关”的方法

Q 值的更新：$Q_\pi(s,a) = (1-\xi)Q_\pi(s,a) + \xi u_t$

- 也可写成：$Q_\pi(s,a) = Q_\pi(s,a) - \xi(Q_\pi(s,a) - u_t)$，其中 u_t 的估计是否准确，称 Q 值更新的关键

模型无关的方法

$u_t = r_t + \lambda r_{t+1} + \lambda^2 r_{t+2} + \ldots$，仅从数据中来估计 u_t

$Q_\pi(s,a) = (1-\xi)Q_\pi(s,a) + \xi u_t$

自助法 $\Longrightarrow$SARSA 算法 $u_t = r_t + \lambda Q_\pi(s', a')$，从以前的积累 $Q_\pi(s', a')$ 和新数据中估计 u_t

$$Q_\pi(s,a) = (1-\xi)Q_\pi(s,a) + \xi(r_t + \lambda Q_\pi(s', a'))$$

最优策略：方法一

- 模型无关的方法评估给定的策略 π
- 利用策略改进算法获得新策略 $\pi \leftarrow \pi_{new}$
- 循环迭代上述过程，需找最优策略。
- 对应 MDP 中的策略迭代算法。

最优策略：方法二

- Q-学习
- 思想：模型无关的方法来获得 Q_{opt}
- 对应 MDP 中的值迭代算法。

MDP 中：值迭代 ⇒ 最优策略

- Q 值更新：
  $$Q_{opt}(s,a) \leftarrow \max_{a \in Action(s)} \sum_{s' \in \mathbf{S}} T(s,a,s')[Reward(s,a,s') + \lambda V_{opt}(s')]$$

强化学习中：Q 学习 ⇒ 最优策略

- Q 值更新：
  $$Q_{opt}(s,a) \leftarrow (1-\xi)Q_\pi(s,a) + \xi(r + \lambda \max_{a' \in Action(s')} Q_\pi(s',a'))$$

比较：SARSA 算法 ⇒ 评估策略 π

- Q 值更新： $Q_\pi(s,a) \leftarrow (1-\xi)Q_\pi(s,a) + \xi(r_t + \lambda Q_\pi(s',a'))$

算法框架

- for $t = 1, 2, \ldots$
  - 选择行动 $a_t = \pi(s_{t-1})$
  - 收集反馈奖励 r_t，获得新状态 s_t
  - 更新参数

解释说明

- 三台老虎机：你选择哪一台去试试？Exploration
- 如果我对某一台老虎机，不妨设为 A，的特点有一定了解了，我想了解得更精确一点，那么我继续试 A。Exploitation
- 了解少的老虎机也许意味着更大的机遇，相应存在更多风险；了解多的老虎机能够给出较稳定的回报，但是我们要追求回报“最大化”

特例 1：贪婪策略

- 选择行动：$\arg \max_{a \in Action(s)} Q_{opt}(s, a)$
- 最强 Exploitation

特例 2：随机策略

- 选择行动：$random(Action(s))$
- 最强 Exploration

平衡二者

- 通常认为要兼顾二者。
- $\epsilon-$ 贪婪：(ϵ 随时间减小)

  $\pi(s) = \arg \max_{a \in Action(s)} Q_{opt}(s, a)$ with probability $1 - \epsilon$

  $\pi(s) = random(Action(s))$ with probability ϵ

Q 学习：能处理已经出现过的状态和行动，

- 现实应用中，可能还有更多的状态和行动从来没有出现过，如何计算 $Q(s, a)$?

用“函数逼近”来近似未出现的 (s, a)

- 如线性回归模型，用已有的 (s, a)Q 值来估计没出现的 (s, a) 的 Q 值，相似 (s, a) 的 Q 值的线性组合
- 何谓相似 (s, a)? 特征相似，状态 s 和行动 a 的特征，比如都能带来好收益的特征，带来糟糕收益的行动等

例子：线性回归逼近

- 定义 $\phi(s, a)$ 是 (s, a) 的特征向量，而定义 $Q_{opt}(s, a; w) = \mathbf{W} \cdot \phi(s, a)$
- 根据已有数据（用机器学习的算法）训练出 $\mathbf{W}$，然后就可以对任意未观测到的 (s, a) 实现 Q 值的估计。

问题

- 函数模型/形式如何确定?
- 特征提取函数如何设计?