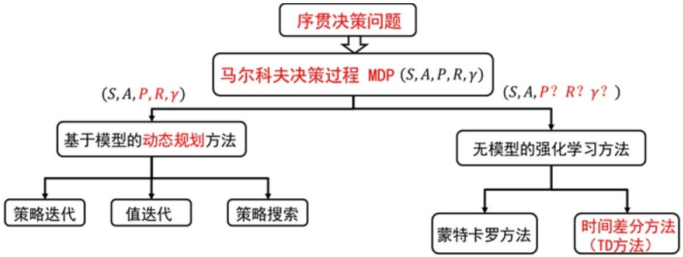
TD 强化学习方法

任永文

中国科学技术大学计算机学院

2023 年 10 月 24 日

TD 引入



蒙特卡罗的方法需要等到每次试验结束，所以学习速度慢，学习效率不高。通过与动态规划方法的比较，我们很自然地会想到：能不能借鉴动态规划中 bootstrapping 的方法，在试验未结束时就估计当前的值函数呢？

TD 思考

- 对于时序差分法来说，我们没有完整的状态序列，只有部分的状态序列，那么如何可以近似求出某个状态的收获呢?
- 回顾 Bellman Equations 推导过程中，值函数的展开 $V(s) = E\ (Gt|St = s) = E\ (Rt+1 + Gt+1 \mid St = s) = E\ (Rt+1 + V(St+1) \mid St = s)$
  $Q(s, a) = E\ (Gt|\ St = s, At = a) = E\ (Rt+1 + Q(St+1, At+1) \mid St = s, At = a)$
- 这启发我们可以用 $Rt+1 + V(St+1)$ 来近似的代替收获 Gt

TD 原理

- 目标：在一个固定的策略 下，从一系列不完整的 episodes 中学习到该策略下的状态价值函数 V (s)
- 回顾蒙特卡罗法的迭代式子是：V(St) ← V(St) + (Gt − V(St))
- 时序差分在预测时，用 Rt+1 + V(St+1) 来估计回报 GtV(St) ← V(St) + (Rt+1 + V(St+1) − V(St)) Q(St, At) ← Q(St, At) + (Rt+1 + Q(St+1, At+1) − Q(St, At))
- Rt+1 + V(St+1) 称为 TD 目标值（target）
- t = Rt+1 + V(St+1)−V(St) 称为 TD 误差（error）
- 将用 TD 目标值近似代替收获 Gt 的过程称为引导（BootStrapping）
- MC 每次更新都需要等到 agent 到达终点之后再更新；而对于 TD learning 来说，agent 每走一步它都可以更新一次，不需要等到到达终点之后才进行更新

TD 代码

Tabular TD(0) for estimating v_π

Input: the policy π to be evaluated
Algorithm parameters: step size $\alpha \in (0, 1]$, small $\varepsilon > 0$
Initialize $V(s)$, for all $s \in \mathcal{S}^+$, arbitrarily except that $V(\mathit{terminal}) = 0$

Loop for each episode:
  Initialize S
  Loop for each step of episode:
    $A \leftarrow$ action given by π for S
    Take action A, observe R, S'
    $V(S) \leftarrow V(S) + \alpha\big[R + \gamma V(S') - V(S)\big]$
    $S \leftarrow S'$
  until S is terminal

TD VS MC

- TD 不需要等到 episode 结束才学习
  - TD 可以在线更新，每一步后都更新
  - MC 必须等到一个 episode 结束后，才能更新
- TD 可以在没有终止状态的环境下学习
  - TD 可以从不完整的 episode 中学习，TD 可以在 continuing（无终止状态）的环境中学习
  - MC 只能从完整的 episode 中学习，MC 只能在 episodic （有终止状态）的环境中学习
- TD 低 variance，有 bias，MC variance 高，无 bias
- TD 体现出了马尔科夫性质，MC 没有