Reinforcement Learning: An Introduction notebook

黎雷蕾

2018 年 1 月 23 日

目录

Chapter 12

Eligibility Traces

12.1 The λ-return

首先给出 n-step 回报公式:

$$G_{t:t+n} \doteq R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{n-1} R_{t+n} + \gamma^n \hat{v}(S_{t+n}, \mathbf{w}_{t+n-1}), \qquad (12.1)$$

在这里,平均回报可以由一半两步回报和一半四步回报构成,即 $G = \frac{1}{2}G_{t:t+2} + \frac{1}{2}G_{t:t+4}$。这种更新方式称为复合更新 (compound update), 由此引出的算法称为 TD(λ) 算法，这个平均包含 n 步回报，权重比例为 $\lambda^{n-1}, \lambda \in [0,1]$，加上系数 $(1-\lambda)$ 保证和为 1(极限求和)。其公式可以写作：

$$G_t^\lambda \doteq (1-\lambda)\sum_{n=1}^{\infty} \lambda^{n-1} G_{t:t+n} \qquad (12.2)$$

对其进行一定的分步计算：

$$G_t^\lambda \doteq (1-\lambda)\sum_{n=1}^{T-t-1} \lambda^{n-1} G_{t:t+n} + \lambda^{T-t-1} G_t \qquad (12.3)$$

由此我们可以引出离线 λ 回报算法 (off-line λ-return algorithm)，在该算法进行中，它不会对权重向量进行改变，根据半梯度 (semi-gradient) 思想，有:

$$\mathbf{w}_{t+1} \doteq \mathbf{w}_t + \alpha\left[G_t^{\lambda s} - \hat{v}(S_t, \mathbf{w}_t)\right]\nabla \hat{v}(S_t, \mathbf{w}_t), \qquad (12.4)$$

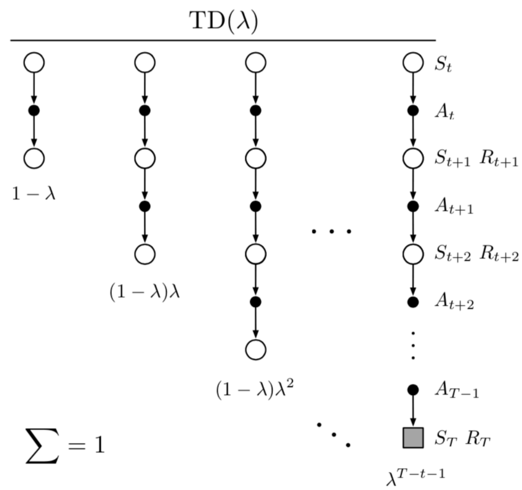

Figure 12.1: The backup digram for TD(λ). If $\lambda = 0$, then the overall backup reduces to its first component, the one-step TD backup, whereas if $\lambda = 1$, then the overall backup reduces to its last component, the Monte Carlo backup.

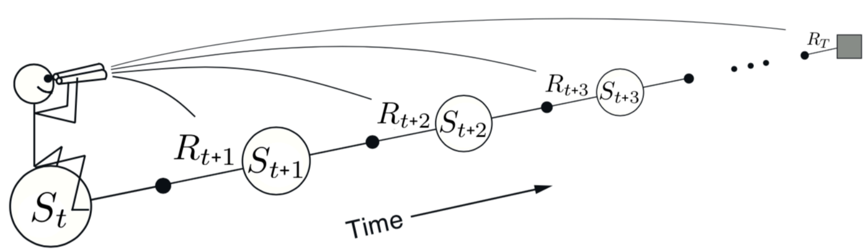

Figure 12.4: The forward view. We decide how to update each state by looking forward to future rewards and states.

12.2 TD(λ)

TD(λ) 算法从三个方面提升离线 λ 回报算法：

1. 在一次模拟的每一步，TD(λ) 都会更新权重向量，而不是模拟结束后再进行更新，这样可以提高当前的估计速度。
2. 这样计算的开销将会等量分布到估计每一时刻，而不是集中在估计结束时候。
3. 除了估计问题，TD(λ) 还可以应用于持续性问题 (continuing problems)。

在函数值逼近中，资格轨迹 (eligibility trace) 向量写为 $\mathbf{e}_t \in \mathbb{R}^d$, 它和权重向量组成形式是一样的。初始值一般设为 0，以值梯度 (value gradient) 进行增量更新，以 $\gamma\lambda$ 作为新的折扣：

$$\begin{aligned} &\mathbf{e}_{-1} \doteq \mathbf{0}, \\ &\mathbf{e}_t \doteq \gamma\lambda\mathbf{e}_{t-1} + \nabla\hat{v}(S_t, \mathbf{w}_t), \quad 0 \le t \le T. \end{aligned} \tag{12.5}$$

对应的 TD error 可以写为：

$$\delta_t \doteq R_{t+1} + \gamma\hat{v}(S_{t+1}, \mathbf{w}_t) - \hat{v}(S_{t+1}, \mathbf{w}_t) \tag{12.6}$$

权重向量的更新公式可以写为：

$$\mathbf{w}_{t+1} \doteq \mathbf{w}_t + \alpha\delta_t\mathbf{e}_t \tag{12.7}$$

那么半梯度 TD(λ) 算法可以概括为：

Semi-gradient TD(λ) for estimating $\hat{v} \approx v_\pi$

1. 输入：需要被估计的策略 π;
2. 输入：一个误差函数 $\hat{v} : \mathcal{S}^+ \times \mathbb{R}^d \to \mathbb{R}$ 且 $\hat{v}(\text{terminal}, \cdot) = 0$;
3. 随机地初始化价值函数权重 $\mathbf{w}$，如 $\mathbf{w} = \mathbf{0}$;
4. 对于每条轨迹，重复如下步骤直到 S' 到达中断状态：

(a) 初始化 S;

(b) n 维向量 $\mathbf{e} \leftarrow \mathbf{0}$;

(c) 对于一条轨迹的每一步，重复如下步骤：

i. 选择 $A \sim \pi(\cdot|S)$;

ii. 执行动作 A，观察 R, S';

iii. $\mathbf{e} \leftarrow \gamma\lambda\mathbf{e} + \nabla\hat{v}(S, \mathbf{w})$;

iv. $\delta \leftarrow R + \gamma\hat{v}(S', \mathbf{w}) - \hat{v}(S, \mathbf{w})$;

v. $\mathbf{w} \leftarrow \mathbf{w} + \alpha\delta\mathbf{e}$;

vi. $S \leftarrow S'$;

对应的示意图如下：

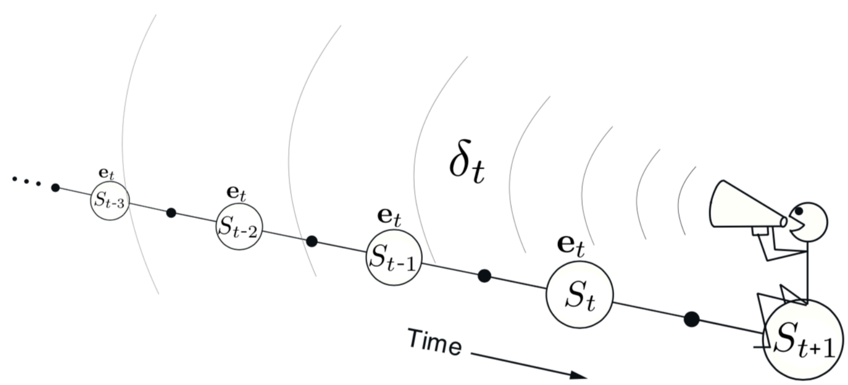

Figure 12.5: The backward or mechanistic view. Each update depends on the current TD error combined with eligibility traces of past events.

12.3 n-step Truncated λ-return Methods

在持续性环境下，λ 回报只有在模拟结束后才能获取，所以我们要定义一个缩短 (truncated) 了的 λ 回报, 即：

$$G_{t:h}^{\lambda} \doteq (1-\lambda)\sum_{n=1}^{h-t-1}\lambda^{n-1}G_{t:t+n} + \lambda^{h-t-1}G_{t:h} \tag{12.8}$$

是一个与 12.3很相近的公式，定义短期 TD(λ) 为 TTD(λ)：

$$\mathbf{w}_{t+n} \doteq \mathbf{w}_{t+n-1} + \alpha \left[G^{\lambda}_{t:t+n} - \hat{v}(S_t, \mathbf{w}_{t+n-1}) \right] \nabla \hat{v}(S_t, \mathbf{w}_{t+n-1}), \tag{12.9}$$

这个算法可以减少每一步计算的开销，但是无法减少计算需要的空间，那么 k 步 λ 回报可以被写成：

$$\begin{aligned} G^{\lambda}_{t:t+k} &= \hat{v}(S_t, \mathbf{w}_{t-1}) + \sum_{i=t}^{t+k-1} (\gamma\lambda)^{i-t} \delta'_t, \\ \delta'_t &\doteq R_{i+1} + \gamma \hat{v}(S_{i+1}, \mathbf{w}_i) - \hat{v}(S_i, \mathbf{w}_{i-1}). \end{aligned} \tag{12.10}$$

12.4 Redoing Updates: The Online λ-return Algorithm

我们希望缩减参数 (truncation parameter)n 尽可能地大，以便于结果接近离线 λ 回报算法；但在同时，我们也希望 n 尽可能小，这样可以使得计算变得比较快。

解决这个矛盾的方法：在每一次模拟时，一旦获得新的数据增量，立马回到模拟初始处并重新进行所有的更新。在这个思想下，更新权重的公式可以写为：

$$\mathbf{w}^h_{t+1} \doteq \mathbf{w}^h_t + \alpha \left[G^{\lambda}_{t:h} - \hat{v}(S_t, \mathbf{w}^h_t) \right] \nabla \hat{v}(S_t, \mathbf{w}^h_t) \tag{12.11}$$