AML, 7 Nov 2019

Policy evaluation in MDPs

Dynamic Programming – One step ahead, bootstrapped, full model

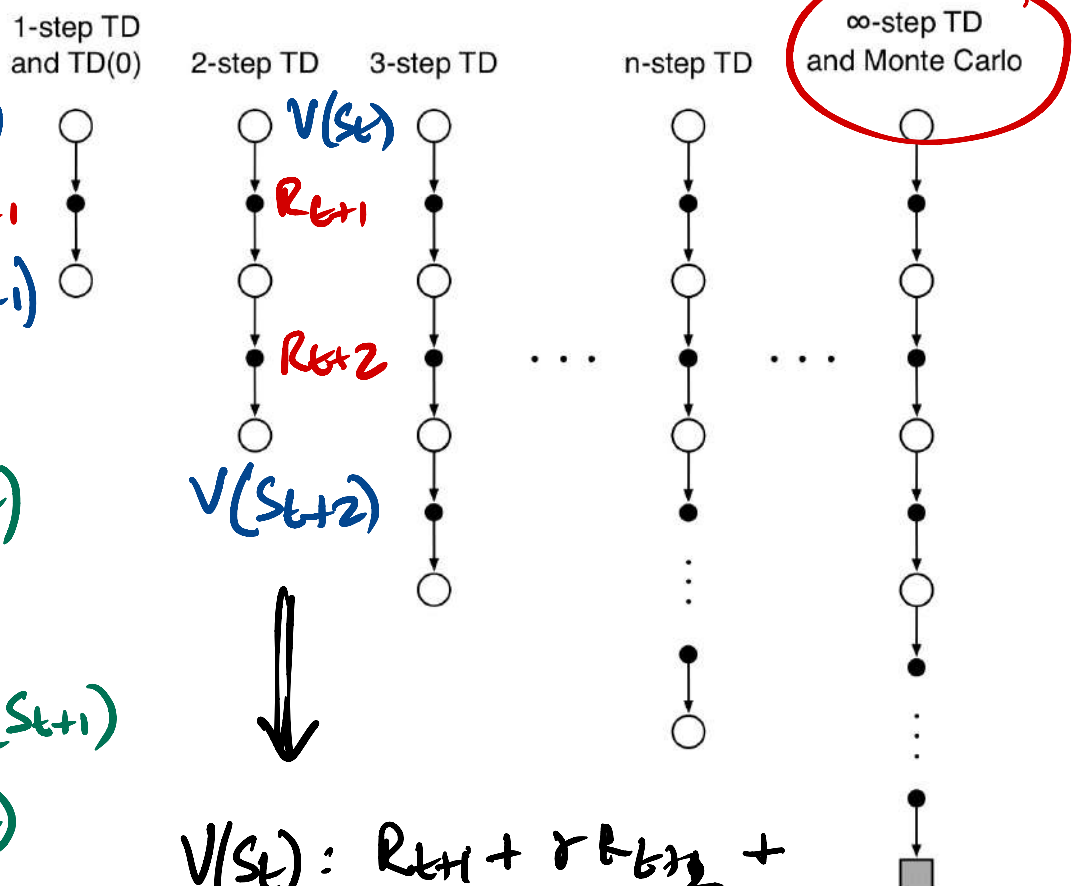
Monte Carlo – Episodes + Mean value

Temporal Difference – TD(0)
– Episodes, bootstrap one step ahead

TD(0) ⟵ TD(n) ⟶ Monte Carlo

TD(0)	TD(n)	Monte Carlo
One Step	n steps	End of Episode

n-step TD



$V(S_t)$

R_{t+1}

$V(S_{t+1})$

$V(S_t)$

R_{t+1}

R_{t+2}

$V(S_{t+2})$

$V(S_t) \leftarrow V(S_t) + \gamma(\delta_t)$

$\delta_t = R_{t+1} + \gamma V(S_{t+1}) - V(S_t)$

$V(S_t): R_{t+1} + \gamma R_{t+2} + \gamma^2 V(S_{t+2})$

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} \cdots + \gamma^{T-t-1} R_T$$

TD(0)

$$G_t = R_{t+1} + \gamma V_t(S_{t+1})$$

$\hookrightarrow$ V at time t

2 steps

$$G_{t:t+2} = R_{t+1} + \gamma R_{t+2} + \gamma^2 V_{t+1}(S_{t+2})$$

$$G_{t:t+n} = R_{t+1} + \gamma R_{t+2} + \cdots \gamma^{n-1} R_{t+n} + \gamma^n V_{t+n-1}(S_{t+n})$$

$V_t(S_t) \rightarrow$ after n observations $\longrightarrow V_{t+n}(S_t)$

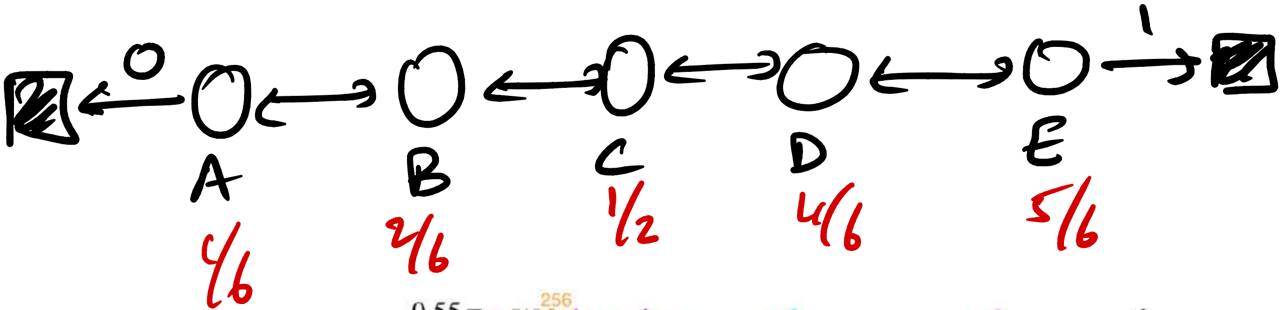
$$V_{t+n}(S_t) = V_{t+n-1}(S_t) + \alpha\left[\underbrace{G_{t:t+n} - V_{t+n-1}(S_t)}_{\delta_{t:t+n}}\right]$$

Algorithm

n-step TD for estimating $V \approx v_\pi$

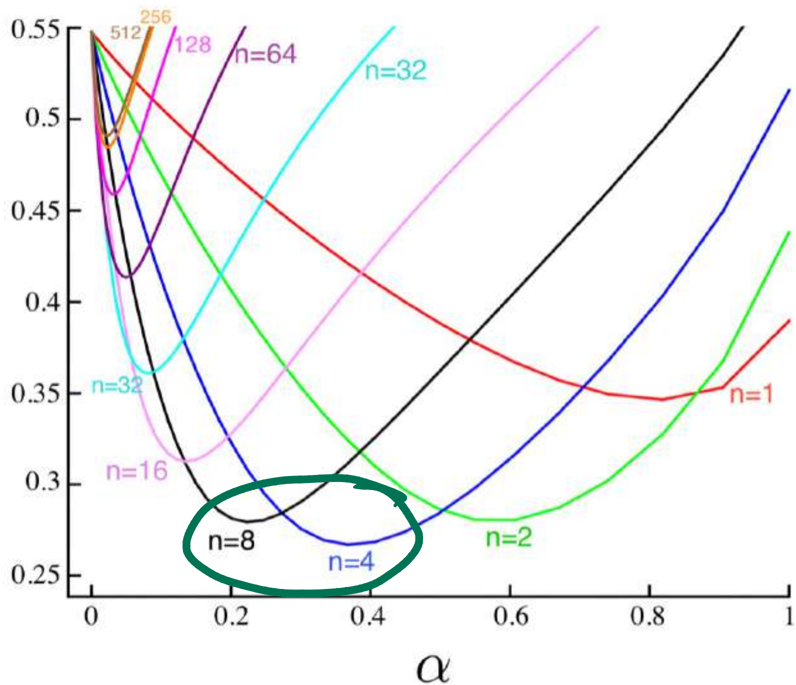
Input: a policy π
Algorithm parameters: step size $\alpha \in (0, 1]$, a positive integer n
Initialize $V(s)$ arbitrarily, for all $s \in \mathcal{S}$
All store and access operations (for S_t and R_t) can take their index mod $n + 1$

Loop for each episode:
    Initialize and store $S_0 \neq$ terminal
    $T \leftarrow \infty$
    Loop for $t = 0, 1, 2, \ldots$:
    | If $\underline{t < T}$, then:
    | Take an action according to $\pi(\cdot|S_t)$
    | Observe and store the next reward as R_{t+1} and the next state as S_{t+1}
    | If S_{t+1} is terminal, then $T \leftarrow t + 1$
    | $\tau \leftarrow \underline{t - n + 1}$ (τ is the time whose state's estimate is being updated)
    | If $\tau \geq 0$:
    | $G \leftarrow \sum_{i=\tau+1}^{\min(\tau+n, T)} \gamma^{i-\tau-1} R_i$ ← $G_{t-n:t}$
    | If $\tau + n < T$, then: $G \leftarrow G + \gamma^n V(S_{\tau+n})$ ← new V $(G_{\tau:\tau+n})$
    | $V(S_\tau) \leftarrow V(S_\tau) + \alpha [G - V(S_\tau)]$ – Update
    Until $\tau = T - 1$



Expand to 19 states

Average RMS error over 19 states and first 10 episodes



= TD(0)

Useful to choose intermediate n

Left −1
Right 0

Similarly

SARSA for policy iteration

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \left[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t) \right]$$

$$(S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1})$$

$$G_{t:t+n} = R_{t+1} + \gamma R_{t+2} + \dots \gamma^{n-1} R_{t+n} + \gamma^n Q_{t+n-1}(S_{t+n}, A_{t+n})$$

$$Q_{t+n}(S_t, A_t) \leftarrow Q_{t+n-1}(S_t, A_t) + \alpha \left[G_{t:t+n} - Q_{t+n-1}(S_t, A_t) \right]$$

Expected SARSA

$$G_t = R_{t+1} + \mathop{\mathbb{E}}_{a}\left[Q(S_t, A_{t+1})\right]$$

Extend to n-steps

Off Policy Learning

policy being computed

Discount each step by $\rho_k = \dfrac{\pi(A_k|S_k)}{b(A_k|S_k)}$

aux policy

$$V_t(S_t) \leftarrow V_t(S_t) + \alpha \cdot \underline{\underline{\rho_t}} \left[G_t - V_t(S_t)\right]$$

Sequence :

$$\rho_{t:t+n} = \prod_{k=t}^{\min(t+n,\,T)} \frac{\pi(A_k \mid S_k)}{b(A_k \mid S_k)}$$

$$V_{t+n}(S_t) \leftarrow V_{t+n-1}(S_t) + \alpha \cdot \rho_{t:t+n} \left[G_{t:t+n} - V_{t+n-1}(S_t)\right]$$

IMPORTANCE SAMPLING

Off policy without importance sampling

TD(0) — Q learning

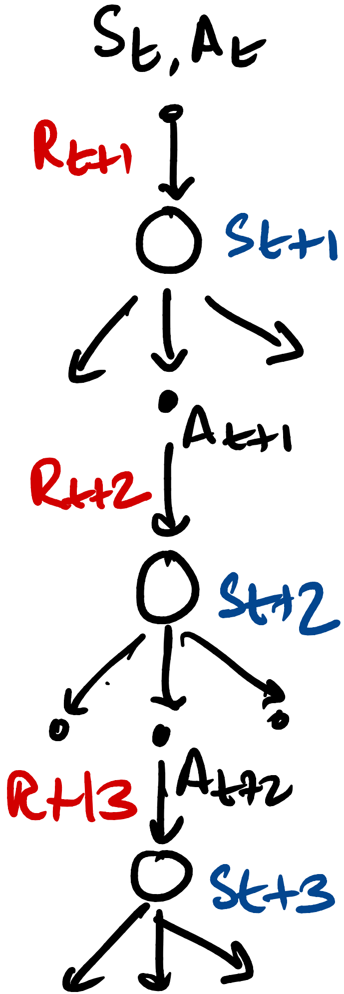
SARSA

$Q(S_{t+1}, A_{t+1})$

$\downarrow$

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \left[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t) \right]$$

Can we do n-step Q learning?



$$G_{t:t+1} \stackrel{def}{=} R_{t+1} + \gamma \sum_a \pi(a|S_{t+1}) \cdot Q_t(S_{t+1}, a)$$

EXPECTED SARSA

$$G_{t:t+2} \stackrel{def}{=} R_{t+1} + \gamma \sum_{a \neq A_{t+1}} \pi(a|S_{t+1}) \, Q_t(S_{t+1}, a)$$

$$+ \gamma \pi(A_{t+1}|S_{t+1}) \left[R_{t+2} + \gamma \sum_a \pi(a|S_{t+2}) \, Q_t(S_{t+2}, a) \right]$$

$$G_{t:t+n} = R_{t+1} + \gamma \sum_{a \neq A_{t+1}} \pi(a|S_{t+1}) Q_{t+n-1}(S_{t+1}, a) + \gamma \pi(A_{t+1}|S_{t+1}) \cdot \underline{G_{t+1:t+n}}$$

n-step TD
n-step SARSA, expected SARSA
n-step TREE LEARNING