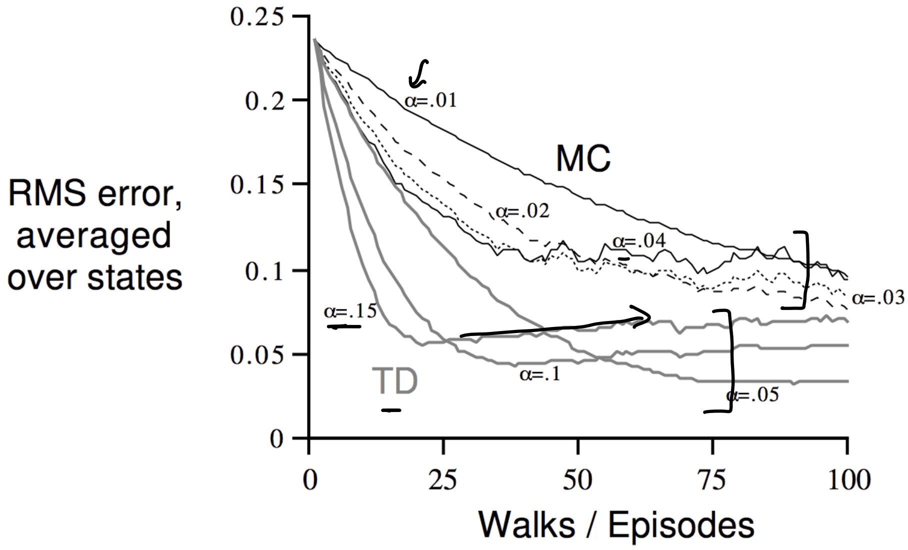
Model free evaluation — monte carlo, TD
control SARSA Q-learning

MC: $V(S_t) \leftarrow V(S_t) + \alpha \left[G_t - V(S_t) \right]$ — (1)

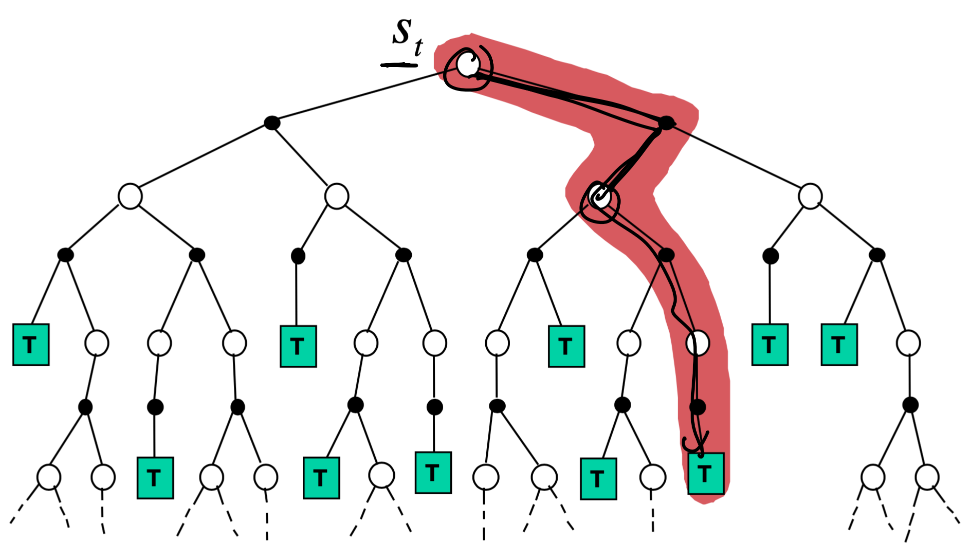
TD: $V(S_t) \leftarrow V(S_t) + \alpha \left[\underbrace{R_{t+1} + \gamma V(S_{t+1})}_{\text{estimate of return}} - V(S_t) \right]$

TD(0)

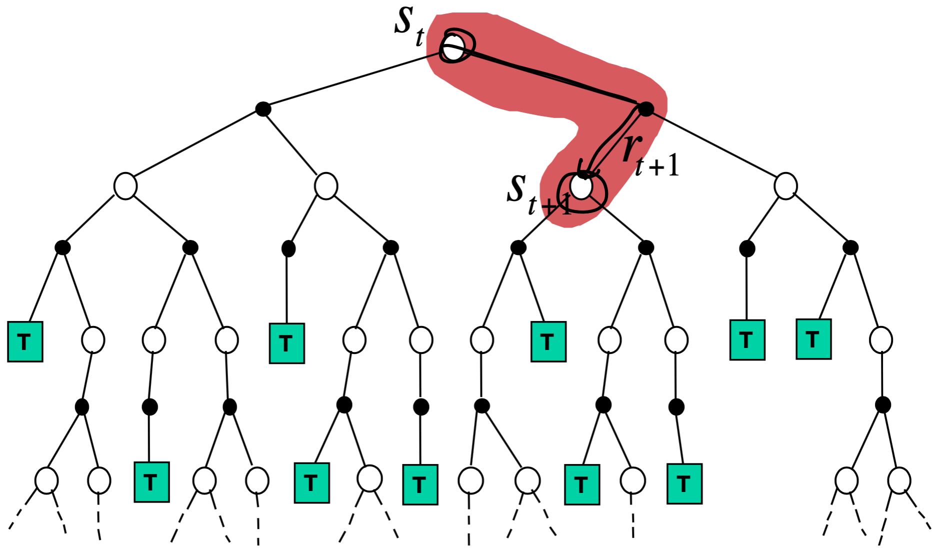
TD	Monte Carlo
- lean from incomplete episodes	
- lower variance but estimates could be biased	- unbiased estimates but high variance
- usually more efficient	
- exploits markovian property	



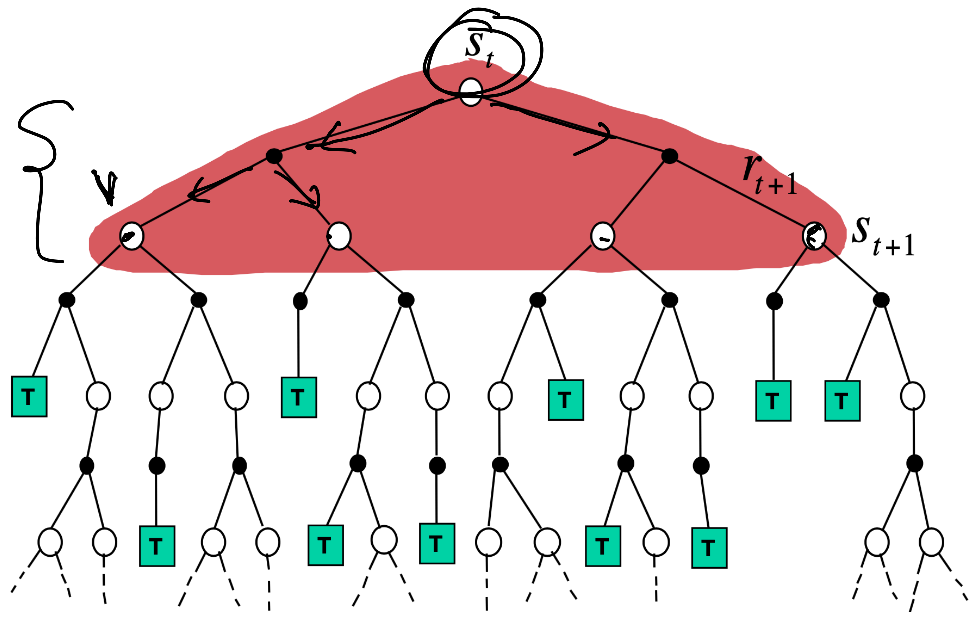
monte carlo



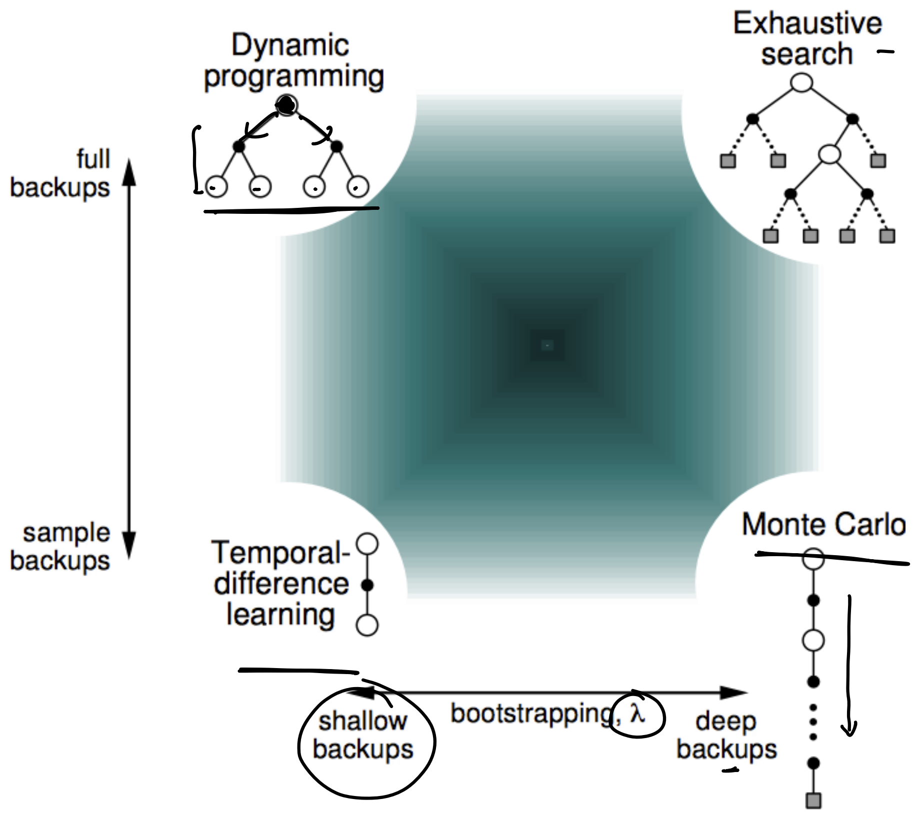
Temporal Diff. Lea



Dynamic Programming



$$\sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a)$$



- $A \xrightarrow{0} B \xrightarrow{0} \blacksquare$
- $B \xrightarrow{1} \blacksquare$
- $B \xrightarrow{1} \blacksquare$
- $B \xrightarrow{1} \blacksquare$
- $B \xrightarrow{0} \blacksquare$
- $B \xrightarrow{1} \blacksquare$
- $B \xrightarrow{1} \blacksquare$
- $B \xrightarrow{1} \blacksquare$

Monte Carlo

$\gamma = 1$

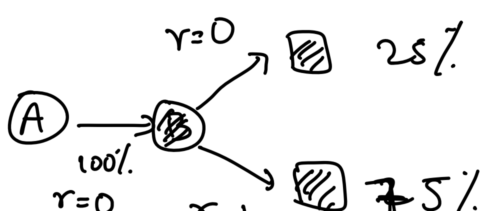
$V(A) = 0$

$V(B) = 3/4$

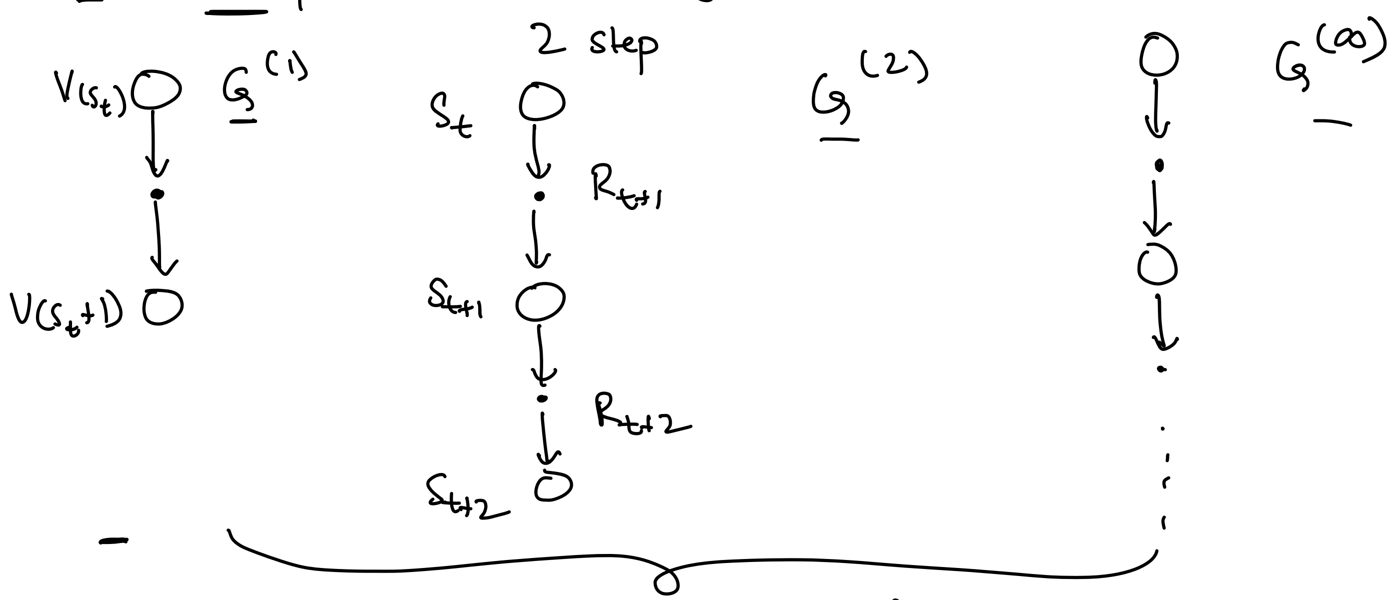
TD

$V(A) = 0.25$

$V(B) = 0.25$



TD 1-step Something in b/w MC



$$V(S_t) \leftarrow V(S_t) + \alpha\left[R_{t+1} + \gamma R_{t+2} + \gamma^2 V(S_{t+2}) - V(S_t)\right]$$

TD(λ)

$$G^{\lambda} = \frac{G^{(1)} + \lambda G^{(2)} + \lambda^2 G^{(3)} + \cdots}{1-\lambda}$$

$$G^{0} = G^{(1)} \qquad \text{TD}(0)$$

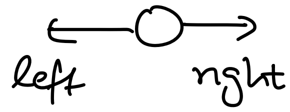
Model-free Control

- Policy Iteration w/ monte carlo estimates.
  - policy evaluation [monte carlo $V = \cancel{v_\pi(s)}$] $q_\pi(s,a)$
  - policy improvements [greedy ....]

won't quite work.

$$\begin{cases} \underset{a}{\operatorname{argmax}}\; q_\pi(s,a) \\ \underset{a}{\operatorname{argmax}} \sum p(s', r \mid s,a)\left[r + \gamma v_\pi(s')\right] \end{cases}$$

Stochastic Environment.



$Q(\text{left}) = 0$

$Q(\text{right}) = 1$

ϵ - greedy policies.

m action

$$\pi(a|s) = \begin{cases} 1-\epsilon + \epsilon/m & \text{if } a = \arg\max Q_\pi(s,a) \\ \epsilon/m & \text{ow} \end{cases}$$

ϵ greedy policy improvement

for any ϵ soft policy π, the ϵ greedy policy π' wrt $q_\pi(s,a)$ is an improvement over π.

$$v_{\pi'}(s) \geq v_\pi(s) \quad \forall s.$$

Monte Carlo Control.

- policy evaluation $\rightarrow$ estimate $q_\pi(s,a)$ & not $v_\pi(s)$
- policy improvement $\rightarrow$ ϵ greedy wrt $Q_\pi(s,a)$

TD Control Estimate $Q_\pi(s,a)$ with TD.

SARSA

Repeat (for episodes)

Sample initial state s

Act A in state s [using ϵ greedy policy wrt $Q(s,a)$]

Repeat (for steps in episode)

execute A & observe S', R

Sample A' using ϵ greedy wrt $Q(s,a)$

* $Q[S,A] \leftarrow Q[S,A) + \alpha\,[R + \gamma\, \underline{Q(S',A')} - Q(S,A)]$

$S \leftarrow S'$

$A \leftarrow A'$

$$Q(S,A) \rightarrow \underline{Q^{*}_{\epsilon}}(S,A) \qquad \epsilon \rightarrow 1/t$$

$$Q^{*}(S,A)$$

It turns out SARSA can be converted into an <u>off policy</u> algorithm.

<u>Q - Learning</u>

- act as per some behavior policy $\mu(\cdot \mid s_t)$

  continuing exploration : as long as it contines to experience all state action pairs.

- But, when updating Q function, use the Q-value of the policy that you are trying to learn.

$$\underline{Q_{\pi}}(S_t, A_t) \leftarrow Q_{\pi}(S_t, \underset{\mu}{A_t}) + \alpha\,(R_{t+1} + \gamma\, Q_{\pi}(S_{t+1}, \overset{A'}{\cancel{A_{t+1}}}) - Q_{\pi}(S_t, A_t))$$

$$\underline{(S_t, A_t, R_{t+1}, S_{t+1})} \xrightarrow{\mu} (S_{t+1}, A_{t+1}, R_{t+2}, S_{t+2}) \longrightarrow \ldots$$

$$A' \sim \pi(S_{t+1})$$

$$Q_{\pi}$$

Q learning for control

- Behaviour policy μ $\leftarrow$ ϵ greedy wrt Q_π function
- Target policy π $\rightarrow$ $\underbrace{\text{greedy}}_{\downarrow\ \arg\max_a Q_\pi(s,a)}$ policy wrt Q.

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

Watkins 92

Q learning converges to $Q^*(S, A)$.