RLearning:
Short guides to reinforcement learning

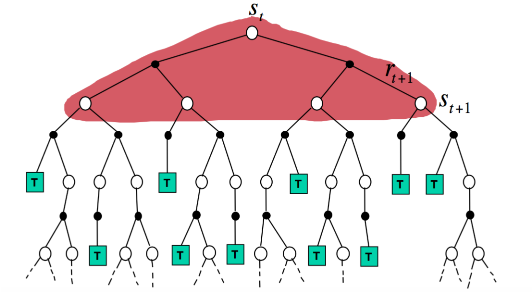
Unit 3-3: Temporal Difference Learning

Davud Rostam-Afschar (Uni Mannheim)

How can we learn by sampling from each step?

Dynamic Programming Backup

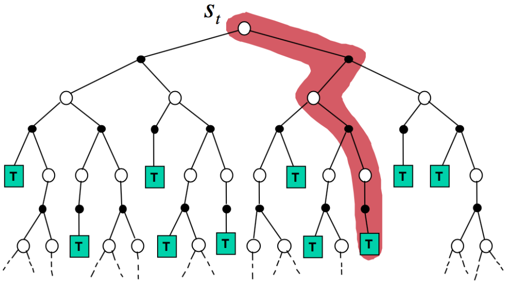
$$V(S_t) \leftarrow \mathbb{E}_\pi \left[R_{t+1} + \gamma V(S_{t+1})\right]$$



Source: David Silver

Monte Carlo Backup

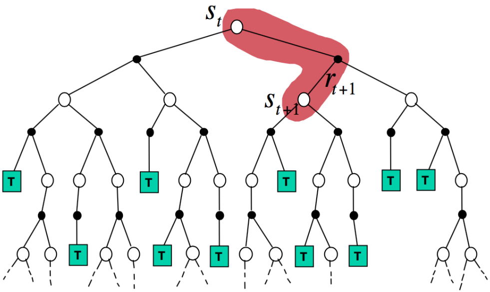
$$V(S_t) \leftarrow V(S_t) + \alpha\left(G_t - V(S_t)\right)$$



Source: David Silver

Temporal Difference Backup

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



Source: David Silver

Model Free Evaluation

- Given a policy π estimate $V^{\pi}(s)$ without any transition or reward model
- **Temporal difference** (TD) evaluation

$$V^{\pi}(s) = E[r \mid s, \pi(s)] + \gamma \sum_{s'} \mathbb{P}\left(s' \mid s, \pi(s)\right) V^{\pi}\left(s'\right)$$

$$\approx r + \gamma V^{\pi}\left(s'\right) \qquad \text{(one draw approximation)}$$

Toy Maze Example

3	**r**	**r**	**r**	**+1**
2	**u**		**u**	**−1**
1	**u**	**l**	**l**	**l**
	1	2	3	4

Start state: (1,1)
Terminal states: (4,2), (4,3)
No discount: $\gamma = 1$

Reward is -0.04 for non-terminal states

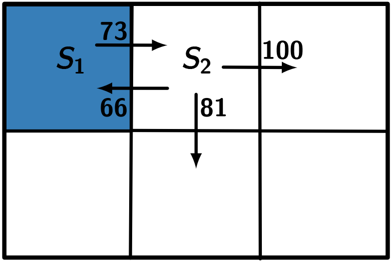
Four actions:

- up (**u**),
- left (**l**),
- right (**r**),
- down (**d**)

Do not know the transition probabilities

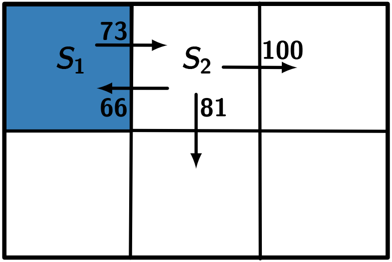
What is the value $V(s)$ of being in state s

Temporal Difference



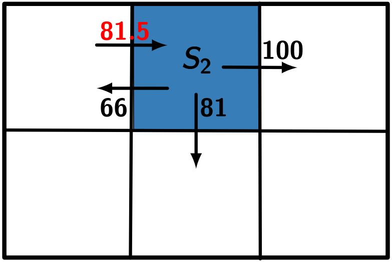
$\gamma = 0.9, \alpha = 0.5, r = 0$ for non-terminal states

Temporal Difference



$\gamma = 0.9, \alpha = 0.5, r = 0$ for non-terminal states

$$
\begin{aligned}
Q(s_1, right) &= Q(s_1, right) + \alpha\left(r + \gamma \max_{a'} Q(s_2, a') - Q(s_1, right)\right) \\
&= 73 + 0.5(0 + 0.9\max\{66, 81, 100\} - 73) \\
&= 73 + 0.5(17) \\
&= 81.5
\end{aligned}
$$



$\gamma = 0.9, \alpha = 0.5, r = 0$ for non-terminal states

$$\begin{aligned}
Q(s_1, right) &= Q(s_1, right) + \alpha\left(r + \gamma \max_{a'} Q(s_2, a') - Q(s_1, right)\right) \\
&= 73 + 0.5(0 + 0.9 \max\{66, 81, 100\} - 73) \\
&= 73 + 0.5(17) \\
&= 81.5
\end{aligned}$$

Temporal Difference Evaluation

- Approximate value function: $V_n^{\pi}(s) \approx r + \gamma V^{\pi}(s')$
- **Incremental update** of sample (π, s', s)

$$V_n^{\pi}(s) \leftarrow V_{n-1}^{\pi}(s) + \alpha_n \left(r + \gamma V_{n-1}^{\pi}(s') - V_{n-1}^{\pi}(s) \right)$$

Stochastic approximation (Robbins-Monro algorithm)

- **Theorem**: If α_n is appropriately decreased with number of times a state is visited then $V_n^\pi(s)$ converges to correct value
- **Sufficient conditions** for α_n :

$$\sum_n \alpha_n \to \infty \tag{1}$$

$$\sum_n \alpha_n^2 < \infty \tag{2}$$

- Often $\alpha_n(s) = 1/n(s)$,
  where $n(s) = \#$ of times s is visited

Temporal Difference (TD) Evaluation

TDevaluation (π, V^{π})
  Repeat
    Execute $\pi(s)$
    Observe s' and r
    Update counts: $n(s) \leftarrow n(s) + 1$
    Learning rate: $\alpha \leftarrow 1/n(s)$
    Update value: $V^{\pi}(s) \leftarrow V^{\pi}(s) + \alpha\left(r + \gamma V^{\pi}(s') - V^{\pi}(s)\right)$
    $s \leftarrow s'$
  Until convergence of V^{π}
  Return V^{π}

Temporal Difference Control

- Approximate Q-function:

$$
\begin{aligned}
Q^*(s,a) =& E[r \mid s,a] + \gamma \sum_{s'} \mathbb{P}\left(s' \mid s,a\right) \max_{a'} Q^*\left(s',a'\right) \\
&\approx r + \gamma \max_{a'} Q^*\left(s',a'\right)
\end{aligned}
$$

- **Incremental update**

  $Q^*_n(s,a) \leftarrow Q^*_{n-1}(s,a) + \alpha_n \left(r + \gamma \max_{a'} Q^*_{n-1}\left(s',a'\right) - Q^*_{n-1}(s,a)\right)$

Comparison

- **Monte Carlo evaluation**:
  - Unbiased estimate
  - High variance
  - Needs many trajectories
- **Temporal difference evaluation**:
  - Biased estimate
  - Lower variance
  - Needs less trajectories

References I

DeNero, J., D. Klein, B. Miller, N. Hay, and P. Abbeel (2013): "The Pacman AI Projects," `http://inst.eecs.berkeley.edu/~cs188/pacman/`, Developed at UC Berkeley. Core by John DeNero and Dan Klein; student autograding by Brad Miller, Nick Hay, and Pieter Abbeel.

Poupart, P. (2025): "Pascal Poupart's Homepage," `https://cs.uwaterloo.ca/~ppoupart/`, Accessed: 2025-05-24.

Russell, S. J., and P. Norvig (2016): *Artificial intelligence: a modern approach*. Pearson.

Sigaud, O., and O. Buffet (2013): *Markov decision processes in artificial intelligence*. John Wiley & Sons.

Sutton, R. S., and A. G. Barto (2018): "Reinforcement learning: An introduction," *A Bradford Book*, Available at `http://incompleteideas.net/book/the-book-2nd.html`.

Szepesvári, C. (2022): *Algorithms for reinforcement learning*. Springer nature, Available at `https://sites.ualberta.ca/~szepesva/RLBook.html`.

Takeaways

How Does TD Learning Update Value Estimates Step-by-Step?

- No need to know transition probabilities or reward function
  $\rightarrow$ Model free
- Combine immediate reward with estimated value of next state
  $\rightarrow$ Biased value estimation from bootstrapped samples
- Revises estimates after each observed step
  $\rightarrow$ Needs few trajectories
- Lower variance than Monte Carlo at the cost of some bias
  $\rightarrow$ Bias-variance tradeoff
- Often used in algorithms such as Q-learning and SARSA