

Model-free Policy Evaluation

Policy evaluation without knowing how the world works

Cathy Wu

6.7920 Reinforcement Learning: Foundations and Methods

Readings

1. NDP §5.1-5.3

2. [Sutton & Barto \(2018\)](#)

- Chapter 5: Monte Carlo Methods. §5.1
- Chapter 6: Temporal-Difference Learning. §6.1-6.4
- Chapter 12: Eligibility Traces. §12.1-12.2

Outline

1. RL vs DP
2. Model-free policy evaluation

Outline

1. **RL vs DP**

- a. What is learning?
- b. Model-based vs model-free
- c. Why learn from samples?

2. Model-free policy evaluation

What is learning?

Learning =
evidence-driven update (data-driven) to a
representation (persistent parameterized mapping) that
improves expected performance on a target
distribution (generalization)

Persistent = parameters are reused across instances from
the target distribution

Model-free vs model-based methods

- Model: P, r components of MDP $M = (S, A, P \text{ or } f, r, T)$

- **Model-based**: **Yes** direct access to model
- **Model-free**: **No** direct access to model

So far (Part 1,
Lectures 1-7), our
discussion has been
model-based.

- Recall: value iteration

$$V_{i+1}(s) = \mathcal{T}V_i(s) = \max_{a \in A} r(s, a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} [V_i(s')] \quad \text{for all } s$$

Key challenge of huge decision spaces

Arcade Learning Environment (ALE)

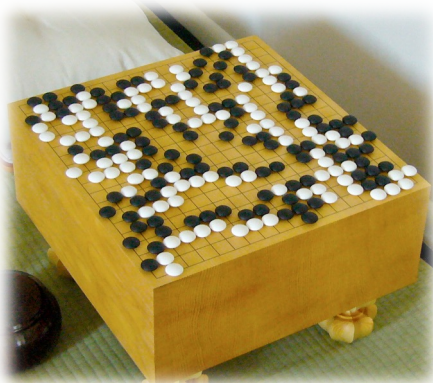


Possible game states: $3^{84 \times 84} \approx 10^{3366}$

For reference:

There are between 10^{78} to 10^{82}
atoms in the observable universe.

Game of Go



Possible game states: $3^{19 \times 19} \approx 10^{172}$

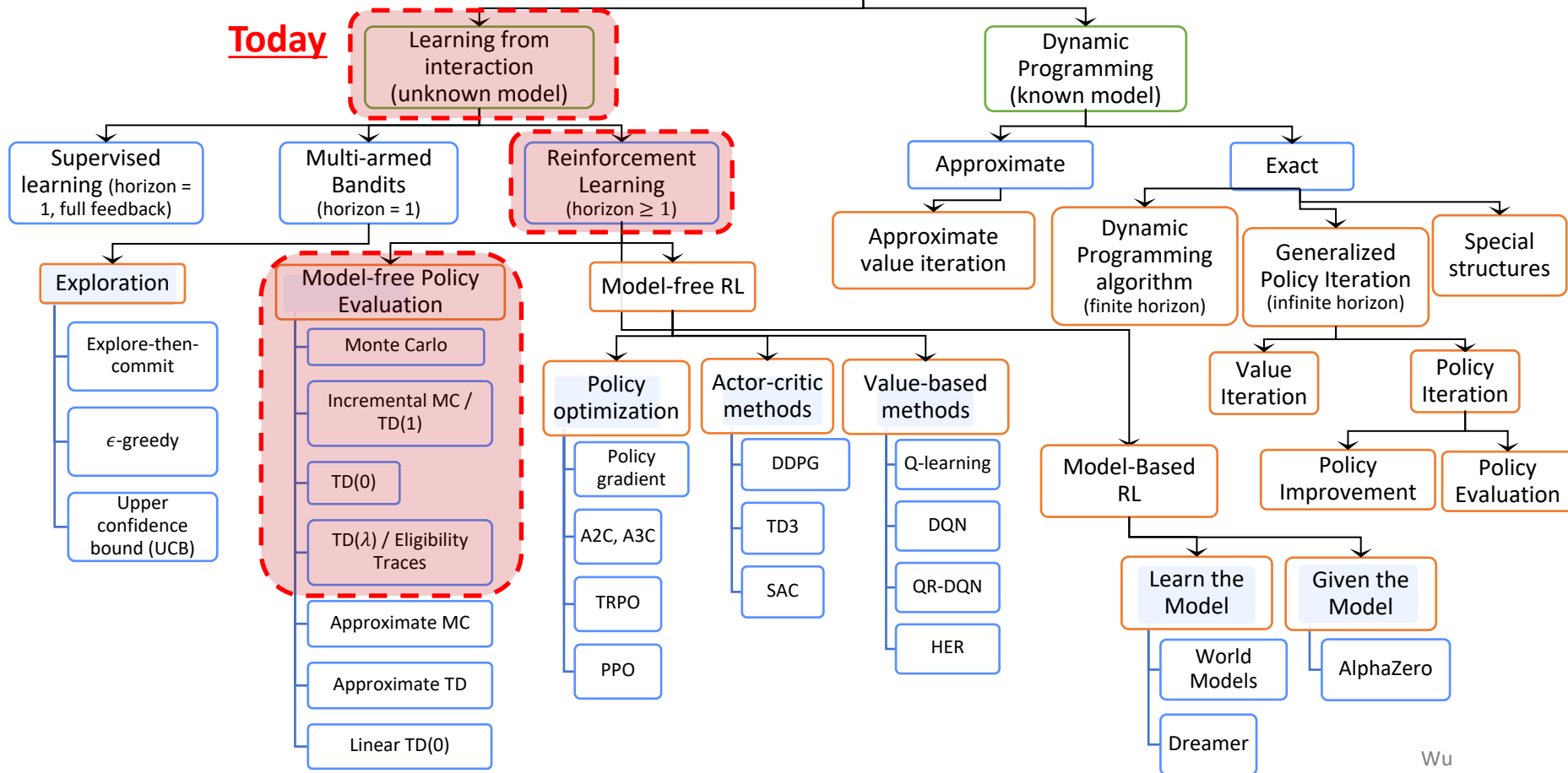
Cannot only explore. Cannot only exploit.
Must trade off exploration and exploitation.

Taxonomy of RL Methods

Methods for
sequential
decision making

Reinforcement learning (RL) is a framework for
sequential decision making under uncertainty.

Today



Outline

1. RL vs DP
2. **Model-free policy evaluation**
 - a. Monte Carlo approximation
 - b. Incremental Monte Carlo, i.e. TD(1)
 - c. Stochastic approximation of a mean
 - d. Temporal difference TD(0)
 - e. TD(λ), eligibility traces

Note on terminology:

In the Sutton & Barto text, policy evaluation is referred to as *prediction*; whereas policy improvement is referred to as *control*. For example, Monte Carlo prediction vs Monte Carlo control (§5.1)

Warm-up: recall policy evaluation

$$V^{\pi}(s) = \mathbb{E} \left[\sum_{t=0}^T \gamma^t r(s_t, \pi(s_t)) | s_0 = s; \pi \right]$$

Notice

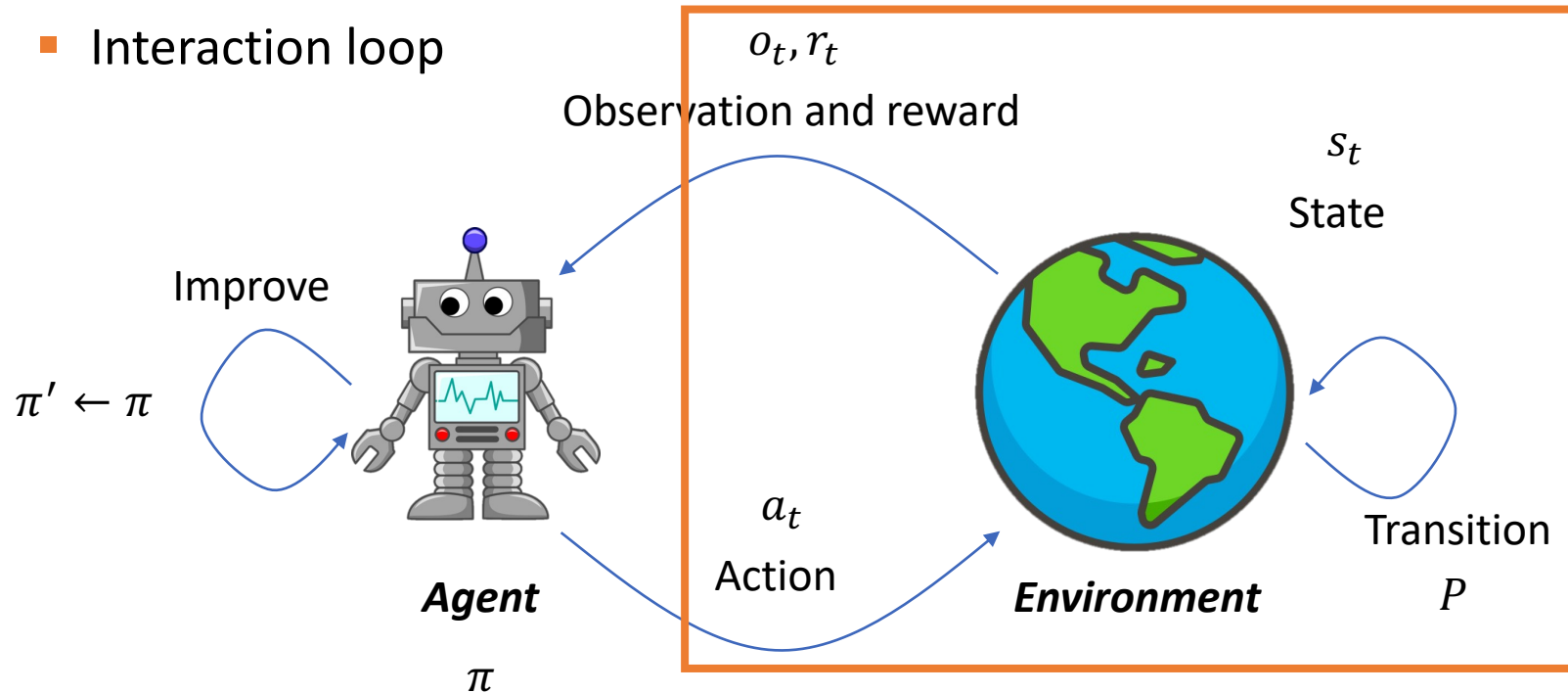
From now on we typically work in the
episodic discounted setting.

Most results smoothly extend to other settings.

Assume: The value functions can be represented *exactly* (e.g. tabular setting).

Recall: The RL setup

- Interaction loop

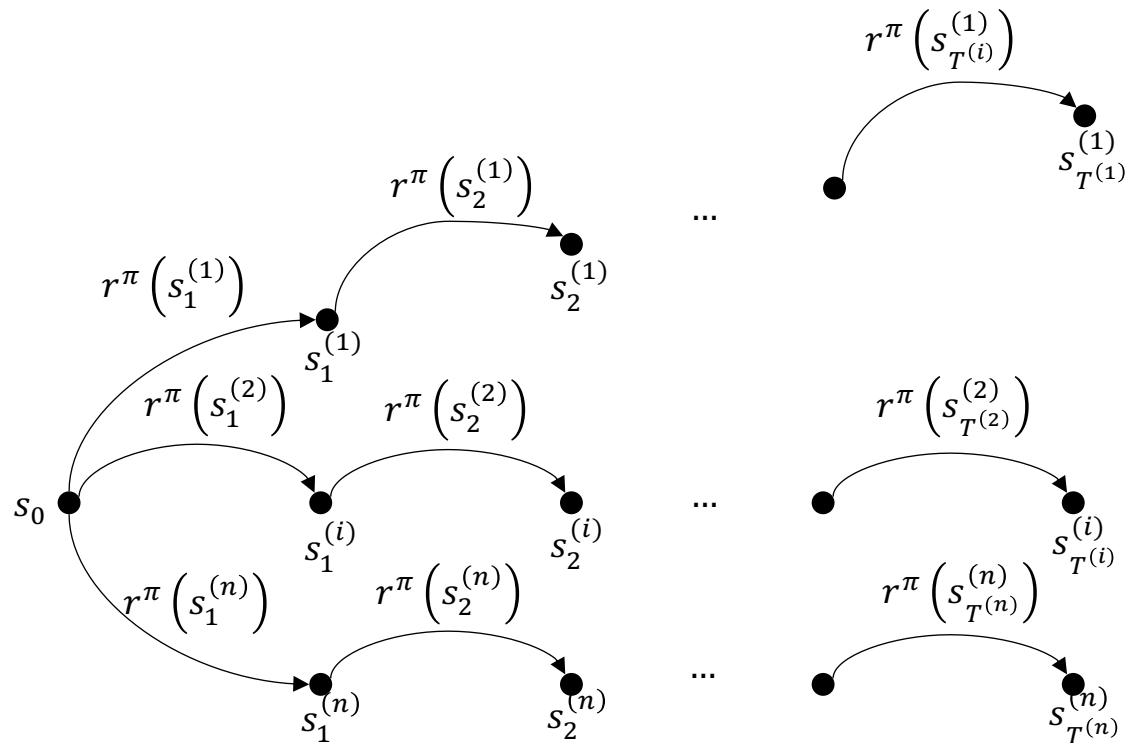


Goal: maximize reward **over time**
(returns, cumulative reward)

$$\max_{\pi \in \Pi} \mathbb{E} \left[\sum_{\tau=0}^{T-1} r(s_\tau, \pi(s_\tau)) \mid s_0 = s; \pi \right]$$



The RL Interaction Protocol



Policy Evaluation

Fixed policy π

For $i = 1, \dots, n$ [each of n episodes]

1. Set $t = 0$
2. Set initial state s_0
3. **While** ($s_{t,i}$ not terminal) [execute one trajectory]
 1. Take action $a_{t,i} = \pi(s_{t,i})$
 2. Observe next state $s_{t+1,i}$ and reward $r_{t+1,i} = r(s_{t,i}, a_{t,i})$
 3. Set $t = t + 1$

EndWhile

Endfor

Return: Estimate of the value function $\hat{V}^\pi(\cdot)$

Policy Evaluation

Approach #1: Utilize the definition of **State Value Function**

Cumulative sum of rewards

$$V^\pi(s) = \mathbb{E} \left[\sum_{t=0}^{T-1} \gamma^t r(s_t, \pi(s_t)) \mid s_0 = s; \pi \right]$$

- Return of trajectory i starting from s_0

$$\hat{R}_i(s_0) = \sum_{t=0}^{T-1} \gamma^t r_{t+1}^{(i)}$$

- Estimated value function

$$\hat{V}_n^\pi(s_0) = \frac{1}{n} \sum_{i=1}^n \hat{R}_i(s_0)$$

Convention for rewards:

We follow the convention of Sutton & Barto (see Chapter 3): $(s_t, a_t, r_{t+1}, s_{t+1})$. Rewards are computed from the 'previous' state and action: $r_{t+1} = r(s_t, a_t)$. That is, rewards start at r_1 , not r_0 .

Monte-Carlo Approximation of a Mean

Definition

Let X be a random variable with mean $\mu = \mathbb{E}[X]$ and variance $\sigma^2 = \mathbb{V}(X)$ and $x_n \sim X$ be n *i.i.d.* realizations of X .

The **Monte-Carlo approximation** of the mean (i.e., the empirical mean) built on n i.i.d. realizations is defined as:

$$\mu_n = \frac{1}{n} \sum_{i=1}^n x_i$$

Monte-Carlo Approximation: Properties

Theorem

The returns used in the Monte-Carlo estimation starting from an initial state s_0 are unbiased estimators of $V^\pi(s_0)$

$$\mathbb{E}[\hat{R}_i(s_0)] = \mathbb{E}\left[r_1 + \gamma r_2^{(i)} + \dots + \gamma^{T_i} r_{T_i}^{(i)}\right] = V^\pi(s_0)$$

Furthermore, the Monte-Carlo estimator converges to the value function

$$\hat{V}_n^\pi(s_0) \xrightarrow{a.s.} V^\pi(s_0)$$

- Proof: Strong law of large numbers
- It applies to any state s used as the beginning of a trajectory (sub-trajectories could be used in practice)
- Finite-sample guarantees are possible (after n trajectories)

Monte-Carlo Approximation: Extensions

Non-episodic problems:

- Interrupt trajectories after H steps:

$$\hat{R}_i(s_0) = \sum_{t=0}^H \gamma^t r_{t+1}^{(i)}$$

- Every return is ignoring a term:

$$\sum_{t=H+1}^{\infty} \gamma^t r_{t+1}^{(i)}$$

Monte-Carlo Approximation: Properties

Theorem

The Monte-Carlo estimator computed over H steps converges to a **biased** value function

$$\hat{V}_n^\pi(s_0) \xrightarrow{\text{a.s.}} \bar{V}_H^\pi(s_0)$$

Such that

$$|\bar{V}_H^\pi(s_0) - V^\pi(s_0)| \leq \gamma^H \frac{r_{\max}}{1 - \gamma}$$

- Proof: by geometric series.

Monte-Carlo: an Incremental Implementation

- Approach #2: **Incremental version** of state value function definition
- Return of trajectory i starting from s_0

$$\hat{R}_i(s_0) = \sum_{t=0}^{T_i-1} \gamma^t r_{t+1}^{(i)}$$

- Estimated value function

$$\hat{V}_n^\pi(s_0) = \frac{1}{n} \sum_{i=1}^n \hat{R}_i(s_0) = \frac{n-1}{n} \hat{V}_{n-1}^\pi(s_0) + \frac{1}{n} \hat{R}_n(s_0)$$

$$\approx (1 - \eta(n)) \hat{V}_{n-1}^\pi(s_0) + \eta(n) \hat{R}_n(s_0)$$

Incremental Monte-Carlo Policy Evaluation

Fixed policy π

For $i = 1, \dots, n$ [each of n episodes]

1. Set $t = 0$
2. Set initial state s_0
3. **While** (s_t not terminal) [execute one trajectory]
 1. Take action $a_t = \pi(s_t)$
 2. Observe next state x_{t+1} and reward $r_{t+1} = r^\pi(s_t)$
 3. Set $t = t + 1$

EndWhile

4. **Update** $\hat{V}_i^\pi(s_0)$ using $TD(1)$ approximation

Endfor

~~Collect trajectories and compute $\hat{V}_n^\pi(s_0)$ using Monte Carlo approximation~~

TD(λ) = temporal differences
with parameter λ
(to be explained later)

Incremental Monte-Carlo: Properties

Theorem

Let the **incremental** Monte-Carlo estimator be computed using a learning rate $\{\eta(n)\}_n$ such that

$$\sum_{i=0}^{\infty} \eta(i) = \infty \quad \sum_{i=0}^{\infty} \eta(i)^2 < \infty \quad [\text{Robbins Monro's condition}]$$

Then

$$\hat{V}_n^{\pi}(s_0) \xrightarrow{a.s.} V^{\pi}(s_0)$$

- Need some new mathematical tools
- Incremental Monte-Carlo estimation converges to V^{π} for a wide range of choices of learning rate schemes.
- This scheme is often referred to as $TD(1)$, for reasons that will be clear shortly.

Stochastic Approximation of a Mean

Definition

Let X be a random variable bounded in $[0,1]$ with mean $\mu = \mathbb{E}[X]$ and $x_n \sim X$ be n *i.i.d.* realizations of X . The stochastic approximation of the mean is,

$$\mu_n = (1 - \eta_n)\mu_{n-1} + \eta_n x_n$$

With $\mu_1 = x_1$ and where (η_n) is a sequence of learning steps.

Stochastic Approximation of a Mean

Proposition

If for any n , $\eta_n \geq 0$ are such that

$$\sum_{n \geq 0} \eta_n = \infty \quad \sum_{n \geq 0} \eta_n^2 < \infty$$

Then

$$\mu_n \xrightarrow{\text{a.s.}} \mu$$

And we say that μ_n is a **consistent** estimator.

Remark: When $\eta_n = \frac{1}{n}$, this is the recursive (incremental) definition of the empirical mean.

Temporal Difference $TD(1)$: Extensions

- **Non-episodic problems**: Truncated trajectories
- **Multiple sub-trajectories**
 - Updates of all the states using sub-trajectories
 - **State-dependent learning rate** $\eta_i(x)$
 - i is the index of the number of updates in that specific state

Note on terminology:

In the Sutton & Barto text, updating the policy evaluation estimate using sub-trajectories but for only the first visit to a state s is called *first-visit* Monte Carlo (§5.1). The use of multiple sub-trajectories, i.e., every visit to a state s , is called *every-visit* Monte Carlo.

Incremental Monte-Carlo Policy Evaluation

Fixed policy π

For $i = 1, \dots, n$ [each of n episodes]

1. Set $t = 0$
2. Set initial state s_0
3. **While** (s_t not terminal) [execute one trajectory]
 1. Take action $a_t = \pi(s_t)$
 2. Observe next state x_{t+1} and reward $r_{t+1} = r^\pi(s_t)$
 3. Set $t = t + 1$

EndWhile

4. **Update** $\hat{V}_i^\pi(s_0)$ using $TD(1)$ approximation

Endfor

~~Collect trajectories and compute $\hat{V}_n^\pi(s_0)$ using Monte Carlo approximation~~

TD(λ) = temporal differences
with parameter λ
(to be explained later)

Temporal-Difference $TD(0)$ Estimation

- Approach #3: Conduct **incremental updates** within trajectories, leveraging the **Bellman equation**
- Recall: The Bellman equation

$$V^\pi(s) = r(s, \pi(s)) + \gamma \mathbb{E}_{s' \sim P(\cdot | s, \pi(s))} [V^\pi(s')]$$

- Incremental update: At each step t , observe s_t, r_{t+1}, s_{t+1} and update estimate $\hat{V}^\pi$ as

$$\hat{V}^\pi(s_t) \leftarrow (1 - \eta) \hat{V}^\pi(s_t) + \eta \left(r_{t+1} + \gamma \hat{V}^\pi(s_{t+1}) \right)$$

Temporal-Difference $TD(0)$: Estimation

- Equivalently temporal difference (TD) error δ_t

$$\hat{V}^\pi(s_t) \leftarrow \hat{V}^\pi(s_t) + \eta \left(r_{t+1} + \gamma \hat{V}^\pi(s_{t+1}) - \hat{V}^\pi(s_t) \right)$$

- Interpretation: temporal-difference error

- Bellman error for function $\hat{V}$ at state s :

$$\begin{aligned} \mathcal{B}^\pi(\hat{V}; s) &= \mathcal{T}^\pi \hat{V}(s) - \hat{V}(s) \\ &= r^\pi(s) + \gamma \mathbb{E}_{s'|s}[\hat{V}(s')] - \hat{V}(s) \quad [\mathcal{B}^\pi(V^\pi; s) = 0] \end{aligned}$$

- Conditioned on s_t , δ_t is an unbiased estimate of $\mathcal{B}^\pi$:

$$\mathbb{E}_{r_t, s_{t+1}}[\delta_t | s_t] = r^\pi(s_t) + \gamma \mathbb{E}_{s_{t+1}|s_t}[\hat{V}^\pi(s_{t+1})] - \hat{V}^\pi(s_t) = \mathcal{B}^\pi(\hat{V}^\pi, s_t)$$

Temporal-Difference $TD(0)$: Properties

Theorem

Let $TD(0)$ run with learning rate $\eta(N_t(s_t))$ where $N_t(s_t)$ is the number of visits to the state s_t . If all states are visited **infinitely often** and the learning rate is set such that:

$$\sum_{t=0}^{\infty} \eta(t) = \infty \quad \sum_{t=0}^{\infty} \eta(t)^2 < \infty \quad [\text{Robbins Monro's condition}]$$

Then for any state $s \in \mathcal{S}$

$$\hat{V}^{\pi}(s) \xrightarrow{a.s.} V^{\pi}(s)$$

Temporal Difference $TD(0)$

For $i = 1, \dots, n$ [each of n episodes]

1. Set $t = 0$
2. Set initial state s_0
3. **While** (s_t not terminal) [execute one trajectory]
 1. Take action $a_{t,i} = \pi(s_{t,i})$
 2. Observe next state $s_{t+1,i}$ and reward $r_{t+1,i} = r(s_{t,i}, a_{t,i})$
 3. Set $t = t + 1$
 4. Update $\hat{V}^\pi(s_{t,i})$ using $TD(0)$ estimation

EndWhile

~~4. Update $\hat{V}_t^\pi(s_0)$ using $TD(1)$ approximation~~

Endfor

~~Collect trajectories and compute $\hat{V}_n^\pi(s_0)$ using Monte Carlo approximation~~

Comparison of $TD(1)$ [Incremental MC] and $TD(0)$

Temporal difference $\delta_t = r_{t+1} + \gamma \hat{V}^\pi(s_{t+1}) - \hat{V}^\pi(s_t)$

- Incremental Monte-Carlo, i.e. $TD(1)$:

$$\hat{V}^\pi(s_0) = \hat{V}^\pi(s_0) + \eta[\delta_0 + \gamma\delta_1 + \dots + \gamma^{T-1}\delta_T]$$

⇒ No bias, large variance [long trajectory]

- $TD(0)$:

$$\hat{V}^\pi(s_0) = \hat{V}^\pi(s_0) + \eta\delta_0$$

⇒ Large bias [“bootstrapping” on wrong values], small variance

Temporal Difference $TD(\lambda)$

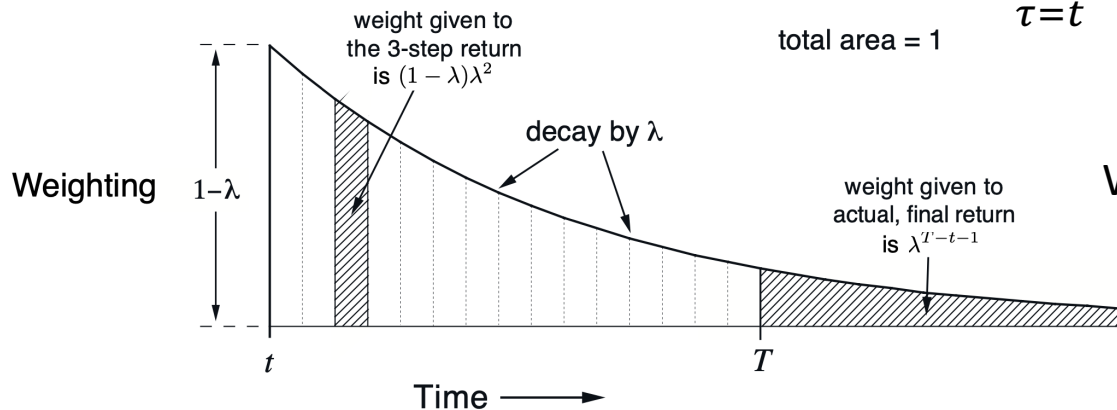
Idea: Use the whole series of temporal differences to update $\hat{V}^\pi$

- Temporal difference of a function $\hat{V}^\pi$ for a transition $\langle s_t, r_{t+1}, s_{t+1} \rangle$

$$\delta_t = r_{t+1} + \gamma \hat{V}^\pi(s_{t+1}) - \hat{V}^\pi(s_t)$$

- Estimated value function

$$\hat{V}^\pi(s_t) = \hat{V}^\pi(s_t) + \eta(s_t) \sum_{\tau=t}^T (\gamma \lambda)^{\tau-t} \delta_\tau$$



Weighting given in the λ -return to each of the n-step returns

The $\mathcal{T}_\lambda^\pi$ Bellman Operator

Definition

Given $\lambda < 1$, then the Bellman operator $\mathcal{T}_\lambda^\pi$ is:

$$\mathcal{T}_\lambda^\pi = (1 - \lambda) \sum_{m \geq 0} \lambda^m (\mathcal{T}^\pi)^{m+1}$$

Remark: Convex combination of the m -step Bellman operators $(\mathcal{T}^\pi)^m$ weighted by a sequence of coefficients defined as a function of a λ .

Same contraction properties as before.

Temporal Difference $TD(\lambda)$

- Estimated value function

$$\hat{V}^{\pi}(s_t) = \hat{V}^{\pi}(s_t) + \eta(s_t) \sum_{\tau=t}^T (\gamma\lambda)^{\tau-t} \delta_{\tau}$$

⇒ Once again requires the whole trajectory before updating...

- Eligibility Traces: book keeping to track which states need to be updated and by how much (due to bias-variance control) as data comes in

Temporal Difference $TD(\lambda)$: Eligibility Traces

- **Eligibility** traces $z \in \mathbb{R}^S$. **Short-term memory vector**.

- At the start of the episode, reset the traces: $z = 0$

- For every transition $s_t \rightarrow s_{t+1}$

1. Compute the temporal difference

$$\delta_t = r_{t+1} + \gamma \hat{V}^\pi(s_{t+1}) - \hat{V}^\pi(s_t)$$

2. Update the eligibility traces

$$z(s) = \begin{cases} \gamma \lambda z(s) & \text{if } s \neq s_t \\ 1 + \gamma \lambda z(s) & \text{if } s = s_t \end{cases} \quad \begin{array}{l} \text{[decay the contribution]} \\ \text{[increment the contribution]} \end{array}$$

3. For all state $s \in S$ [all states are updated at each step]

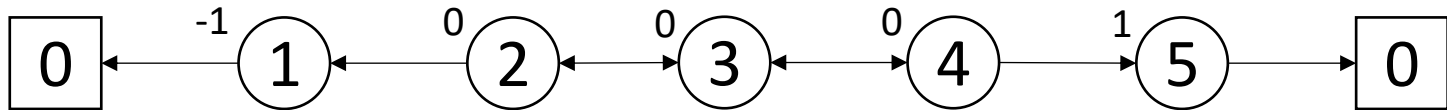
$$\hat{V}^\pi(s) \leftarrow \hat{V}^\pi(s) + \eta(s) z(s) \delta_t$$

Sensitivity to λ

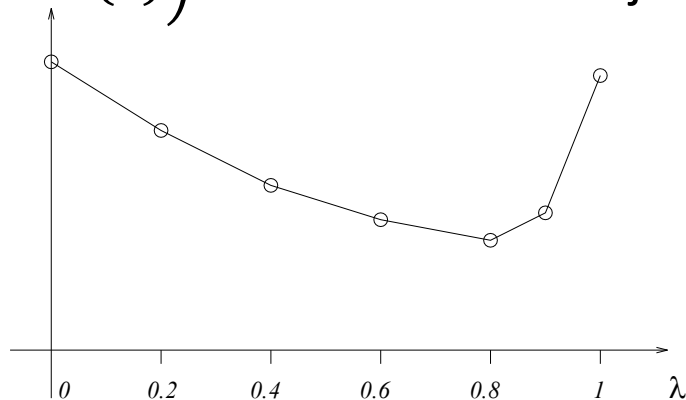
- $\lambda < 1$: smaller variance w.r.t. $\lambda = 1$ ($\approx$ incremental Monte-Carlo)
- $\lambda > 0$: faster propagation of rewards w.r.t. $\lambda = 0$

Example: Sensitivity to λ

Linear chain example



Error $\sum_{s \in S} \left(\hat{V}^{\pi}(s) - V^{\pi}(s) \right)^2$ after $n = 100$ trajectories



Summary of methods

		TD(1)	TD(0)
	Dynamic Programming	Monte Carlo	Temporal Difference
Model Free?	No	Yes	Yes
Non-episodic domains?	Yes	No	Yes
Non-Markovian domains?	No	Yes	No
Converges to true value	Yes	Yes	Yes
Unbiased Estimate	N/A	Yes	No
Variance	N/A	High	Low

Summary

- Reinforcement learning vs dynamic programming
 - **Model-free** vs **model-based**
 - **Learning** = updating a representation using data to improve expected performance on unseen cases
 - **Planning** = compute decisions from a known model
- Learning for **policy evaluation**: **Monte-Carlo** and **Temporal Difference** methods
 - MC $\rightarrow$ Incremental MC / TD(1) $\rightarrow$ TD(0) $\rightarrow$ TD(λ) $\rightarrow$ Eligibility Traces
 - **Incremental mean**: warm-up for stochastic approximation theory
 - Pros and cons: **Bias-variance tradeoff**, update timing

References

1. Alessandro Lazaric. INRIA Lille. Reinforcement Learning. 2017, Lectures 2-3.
2. Sutton & Barto (2018). Chapter 12: Eligibility Traces. §12.1-12.2

Appendix

Sampling settings

- **Learning with generative model.** A **black-box simulator** f of the environment is available. Given (s, a) ,
$$f(s, a) = \{s', r'\} \text{ with } s' \sim p(\cdot | s, a), r' = r(s, a)$$
- **Episodic learning.** Multiple **trajectories** can be repeatedly generated from some initial states and terminating when a **reset** condition is achieved:

$$(s_{0,i}, s_{1,i}, \dots, s_{T,i})_{i=1}^n$$

- **Online learning.** At each time t the agent is at state s_t , it takes action a_t , it observes a transition to state s_{t+1} , and it receives a reward r_{t+1} . We assume that $s_{t+1} \sim p(\cdot | s_t, a_t)$ and $r_{t+1} = r(s_t, a_t)$ (i.e., MDP assumption). No **reset**.

Monte-Carlo Approximation of a Mean

- **Unbiased estimator:** Then $\mathbb{E}[\mu_n] = \mu$ (and $\mathbb{V}(\mu_n) = \frac{\mathbb{V}(X)}{n}$)
- **Weak law of large numbers:** $\mu_n \xrightarrow{P} \mu$
- **Strong law of large numbers:** $\mu_n \xrightarrow{a.s.} \mu$
- **Central limit theorem (CLT):** $\sqrt{n} (\mu_n - \mu) \xrightarrow{D} \mathcal{N}(0, \mathbb{V}(X))$
- **Finite sample guarantee:**

$$\underbrace{\mathbb{P} \left[\underbrace{\left| \frac{1}{n} \sum_{i=1}^n X_t - \mathbb{E}[X_1] \right|}_{\text{deviation}} > \underbrace{\epsilon}_{\text{accuracy}} \right]}_{\text{confidence}} \leq 2 \exp \left(- \frac{2n\epsilon^2}{(b-a)^2} \right)$$

$$\mathbb{P} \left[\left| \frac{1}{n} \sum_{i=1}^n X_t - \mathbb{E}[X_1] \right| > \epsilon \right] \leq \delta$$

$$\text{If } n \geq \frac{(b-a)^2 \log\left(\frac{2}{\delta}\right)}{2\epsilon^2}$$

Reference: Convergence of Random Variables

Let X be a random variable and $\{X_n\}_{n \in \mathbb{N}}$ a sequence of random variables.

- $\{X_n\}$ converges to X **almost surely**, $X_n \xrightarrow{a.s.} X$, if:

$$\mathbb{P}\left(\lim_{n \rightarrow \infty} X_n = X\right) = 1$$

- $\{X_n\}$ converges to X **in probability**, $X_n \xrightarrow{P} X$, if for any $\epsilon > 0$:

$$\lim_{n \rightarrow \infty} \mathbb{P}[|X_n - X| > \epsilon] = 0$$

- $\{X_n\}$ converges to X **in law**, $X_n \xrightarrow{D} X$, if for any bounded continuous function f :

$$\lim_{n \rightarrow \infty} \mathbb{E}[f(X_n)] = \mathbb{E}[f(X)]$$

- $\{X_n\}$ converges to X **in expectation**, $X_n \xrightarrow{L^1} X$, if:

$$\lim_{n \rightarrow \infty} \mathbb{E}[X_n] = \mathbb{E}[X]$$

See HW0 for examples & counterexamples

Remark: $X_n \xrightarrow{a.s.} X \implies X_n \xrightarrow{P} X \implies X_n \xrightarrow{D} X$

Stochastic Approximation of a Mean

If $\eta_n = \frac{1}{n}$, then $\mu_n = \frac{1}{n} \sum_{i=1}^n x_i$.

Proof: Base case ($n = 1$): $\mu_1 = x_1$ (given).

Induction step. Assume $\mu_n = \frac{1}{n} \sum_{i=1}^n x_i$.

$$\begin{aligned}
 \mu_{n+1} &= \left(1 - \frac{1}{n+1}\right) \mu_n + \frac{1}{n+1} x_{n+1} \\
 &= \left(\frac{n}{n+1}\right) \mu_n + \frac{1}{n+1} x_{n+1} \\
 &= \left(\frac{n}{n+1}\right) \frac{1}{n} \sum_{i=1}^n x_i + \frac{1}{n+1} x_{n+1} \\
 &= \left(\frac{1}{n+1}\right) \sum_{i=1}^n x_i + \frac{1}{n+1} x_{n+1} \\
 &= \left(\frac{1}{n+1}\right) \sum_{i=1}^{n+1} x_i
 \end{aligned}$$