Off-Policy Learning

Reinforcement Learning Summer School

Martha White
University of Alberta and AMII

Comments for the lecture

- Please ask questions (this is a summer school)
- I will give exercises along the way
- Some of these insights/comments will be available next week, in an arxiv paper by Sina Ghiassian, Adam White, Rich Sutton and myself
    - (probably) called “A Comparison of Off-Policy Policy Evaluation Methods”
- Outcomes: you will understand some of
    - the goals for off-policy learning
    - the key challenges
    - recent algorithmic developments to address some of the challenges

What is off-policy learning?

- Behaviour policy μ different from target policy π
- Example: an RL agent controlling energy storage for batteries could follow the default (acceptable) policy, and
  - evaluate a different proposed policy
  - learn the optimal policy
- Note: agent does not have access to the true model

I'll focus on policy evaluation

Policy evaluation

- Goal: learn the values V (or Q) for the target policy $\boldsymbol{\pi}$

$$\sum_{s} d(s)\mathbb{E}_{\pi}[\delta(S, A, S')|S = s] = 0$$

- For linear value functions,V(s) = < x(s), w >

$$\sum_{s \in \mathcal{S}} d(s)\mathbb{E}_{\pi}[\delta(S, A, S')\mathbf{x}(S)|S = s] = 0$$

where $\mathcal{S}$ is the set of states

$\mathcal{A}$ is the set of actions

$\mathbf{x} : \mathcal{S} \to \mathbb{R}^d$ gives the features

$\delta(S, A, S') = R(S, A, S') + \gamma V(S') - V(S)$

$d : \mathcal{S} \to [0, 1]$ distribution over states

Policy evaluation

$$\mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s]$$
$$= \sum_{a \in \mathcal{A}} \pi(a|s) \sum_{s' \in \mathcal{S}} P(s, a, s')\delta(s, a, s')\mathbf{x}(s)$$

$$\sum_{s \in \mathcal{S}} d(s)\mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s] = 0$$

where $\mathcal{S}$ is the set of states

$\mathcal{A}$ is the set of actions

$\mathbf{x} : \mathcal{S} \to \mathbb{R}^d$ gives the features

$\delta(S, A, S') = R(S, A, S') + \gamma V(S') - V(S)$

$d : \mathcal{S} \to [0, 1]$ distribution over states

How can we find this fixed point in the on-policy setting?

- Temporal difference learning $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \alpha \delta_t \mathbf{x}_t$

$$\text{Expected TD-update: } \mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s]$$

  - Recall goal: $\sum_{s \in \mathcal{S}} d(s) \mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s] = 0$
- What other algorithms can be used? **LSTD. Any Others?**
- How does this change for off-policy learning?

The modifications due to off-policy learning

- Imagine you see a transition (s, a, s', r), by taking actions according to $\mu(\cdot|s)$
- 1. Need to know what the TD-update would have been had we selected 'a' according to $\pi(\cdot|s)$
- 2. If we ran $\boldsymbol{\pi}$, instead of $\boldsymbol{\mu}$, we would likely see the state s with a different frequency. Is this a problem?

Addressing Point 1: Adjust **action** probabilities

- Can generate data from $\boldsymbol{\mu}$, but still get an unbiased sample of TD-update for $\boldsymbol{\pi}$ using importance sampling

$$(s, a, s', r), \ \text{with } a \sim \mu(\cdot|s)$$

$$\begin{aligned}
&\mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S)|S = s] \\
&= \sum_{a \in \mathcal{A}} \pi(a|s) \sum_{s' \in \mathcal{S}} P(s, a, s')\delta(s, a, s')\mathbf{x}(s) \\
&= \sum_{a \in \mathcal{A}} \frac{\mu(a|s)}{\mu(a|s)}\pi(a|s) \sum_{s' \in \mathcal{S}} P(s, a, s')\delta(s, a, s')\mathbf{x}(s) \\
&= \sum_{a \in \mathcal{A}} \mu(a|s) \sum_{s' \in \mathcal{S}} P(s, a, s')\frac{\pi(a|s)}{\mu(a|s)}\delta(s, a, s')\mathbf{x}(s) \\
&= \mathbb{E}_\mu[\rho(S, A)\delta(S, A, S')\mathbf{x}(S)|S = s]
\end{aligned}$$

$$\rho(s, a) = \frac{\pi(a|s)}{\mu(a|s)}$$

Addressing Point 1: Adjust action probabilities

- Can generate data from $\boldsymbol{\mu}$, but still get an unbiased sample of TD-update for $\boldsymbol{\pi}$ using importance sampling

$$(s, a, s', r), \text{ with } a \sim \mu(\cdot|s)$$

$$\begin{aligned}&\mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S)|S = s]\\ &= \mathbb{E}_\mu[\rho(S, A)\delta(S, A, S')\mathbf{x}(S)|S = s]\end{aligned}$$

Off-policy TD-update:

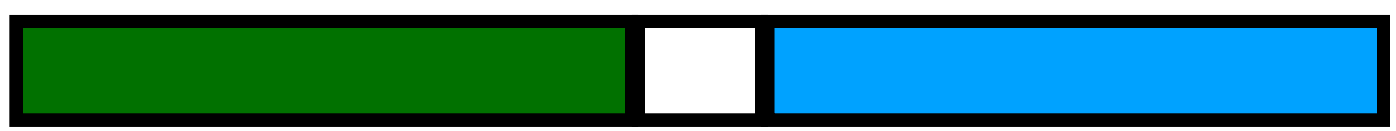
$$\mathbf{w}_{t+1} \leftarrow \mathbf{w} + \alpha\rho(S_t, A_t)\delta(S_t, A_t, S_{t+1})\mathbf{x}(S_t)$$

$$\rho(s, a) = \frac{\pi(a|s)}{\mu(a|s)}$$

Addressing Point 2: Adjust **state** probabilities

- Generating data from $\boldsymbol{\mu}$ induces different state-visitation frequencies (stationary distribution $d_\mu : \mathcal{S} \rightarrow [0,1]$)

$$\mu(\text{left}|S = \text{white}) = 0.95$$



$$\pi(\text{left}|S = \text{white}) = 0.2$$

Let's resolve this later

Summary

- Off-policy different from on-policy in two key ways
- Importance sampling let's us adjust action-probabilities (e.g., TD -> off-policy TD)
- The focus of this lecture:
    - Why should I care about off-policy learning?
    - How do we address point 2 (and should we?) (i.e, what is the objective)
    - Are there other (better) algorithms than off-policy TD?

Why is it important?

- Safe reinforcement learning
  - can learn about other policies, while following a trusted policy
- What-if questions
- Predictive knowledge

General Value Functions

- Options **Doina Precup will talk about this**

What-if Questions

- We get one stream of experience: only get to behave one way
- But want to ask questions about other ways of behaving
- e.g., What if I were to more aggressively discharge energy from the battery, until energy prices were low?
- e.g., How much nicer would it be to take a detour through the park (while walking your usual efficient way home)?
- e.g., How long will it take me to get to the store?

One stream of experience

- Asynchronous methods (e.g., A3C) only applicable in settings where can have multiple instances of the environment (e.g., games, simulators)
    - A3C has multiple policies interacting with their own instance of the environment, updating a central estimate
- General AI agent only has one life and one environment

What-if questions can be encoded as value functions

- Policy-contingent questions about the future (contingent on some other way you will behave π)

- Value function corresponds to expected discounted cumulative sum of a signal, into the future

- Recent generalizations to cumulant instead of reward (to be any signal) and state-dependent discount

see “Horde: A Scalable Real-time Architecture for Learning Knowledge from Unsupervised Sensorimotor Interaction”, Sutton et al., 2011

Example: How long will it take me to get to the store?

- Policy: the policy that goes to the store
- Cumulant: $C_{t+1} = 1$ everywhere
- Discount: $\gamma_t = 0$ when reach store, else 1.0
- $V(s) = E_\pi[C_{t+1} + \gamma_t V(S_{t+1})] = E_\pi[C_{t+1} + \gamma_t C_{t+2} + \gamma_{t+1} \gamma_t V(S_{t+2})] = \ldots$
  - until $\gamma_t = 0$ ends the cumulation when reach the store
- More precisely: how many steps until I can reach the store

see “Horde: A Scalable Real-time Architecture for Learning Knowledge from Unsupervised Sensorimotor Interaction”, Sutton et al., 2011

Need off-policy learning to learn GVFs

- Learning GVFs is usually an off-policy learning problem
  - Policy associated with GVF may not be the behaviour
- To accurately answer a variety of such What-If questions, understanding off-policy learning is important

Exercise: Implications of thinking in terms of off-policy learning

- What are natural directions/open questions, given
  - there is one stream of data (one possibly changing behaviour)
  - we can learn (many things) off-policy

Turn to the person beside you and discuss with them (in twos or threes)

So how do we learn these What-if questions?

- A simplified variant of the MSPBE objective is

$$\Big\| \sum_{s \in \mathcal{S}} d_\pi(s) \mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s] \Big\|_2^2$$

- On-policy TD update obtains minimum of this objective
  - what is the value of the objective at the minimum?
- What are the possible choices for the objective function in off-policy learning?

Excursions Objective

$$\Big\| \sum_{s \in \mathcal{S}} d_\mu(s) \mathbb{E}_\pi [\delta(S, A, S') \mathbf{x}(S) | S = s] \Big\|_2^2$$

$$= \Big\| \sum_{s \in \mathcal{S}} d_\mu(s) \mathbb{E}_\mu [\rho(S, A) \delta(S, A, S') \mathbf{x}(S) | S = s] \Big\|_2^2$$

- Off-policy TD is trying to find weights to make this objective zero (adjust action distribution)

Off-policy TD-update:

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w} + \alpha \rho(S_t, A_t) \delta(S_t, A_t, S_{t+1}) \mathbf{x}(S_t)$$

Alternative Life

- Imagine you still want to solve for the original fixed-point

$$\left\| \sum_{s \in \mathcal{S}} d_\pi(s) \mathbb{E}_\pi [\delta(S, A, S') \mathbf{x}(S) | S = s] \right\|_2^2$$

- Then have to correct for stationary distribution d_μ

Correcting the stationary distribution (visitation frequency)

See $s_1, a_1, s_2, a_2, \ldots, s_{n+1}$

How likely is this sequence under π?

$$P(s_1, a_1, s_2, a_2, \ldots, s_{n+1}|\pi)$$

$$= \pi(a_1|s_1)P(s_2|s_1, a_1) \ldots \pi(a_n|s_n)P(s_{n+1}|s_n, a_n)$$

Importance sample entire trajectory

$$\frac{P(s_1, a_1, s_2, a_2, \ldots, s_{n+1}|\pi)}{P(s_1, a_1, s_2, a_2, \ldots, s_{n+1}|\mu)} = \rho(a_1|s_1) \ldots \rho(a_n|s_n)$$

Alternative-Life Off-Policy TD

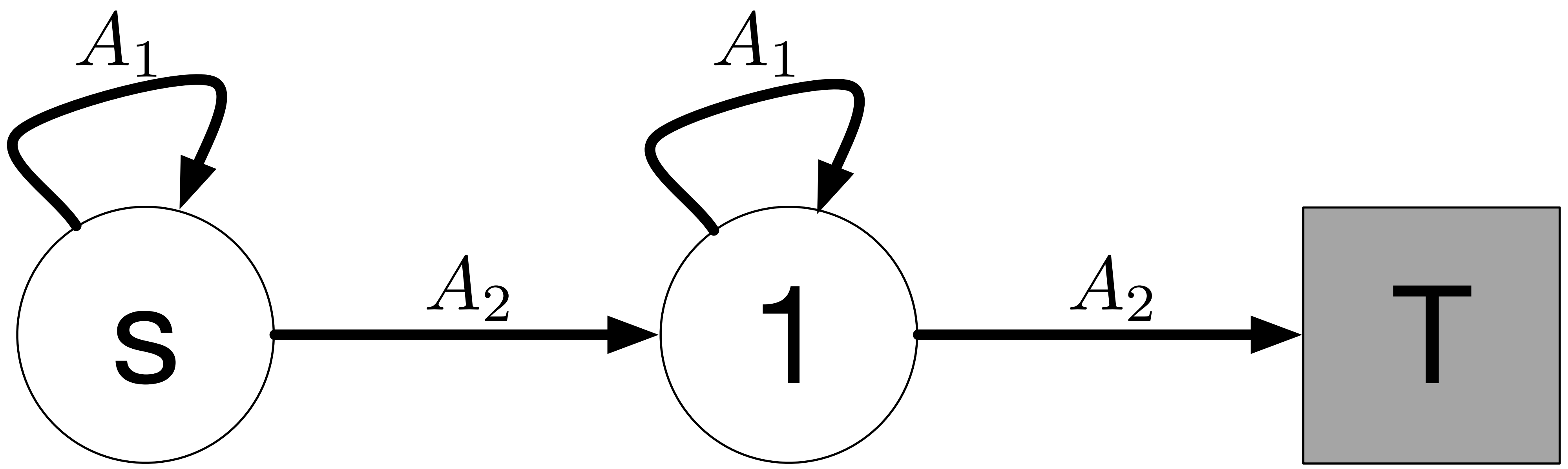
$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \alpha \left(\prod_{k=1}^{t} \rho_k \right) \delta_t \mathbf{x}_t$$

Importance sample trajectory up to t

$$\frac{P(s_1, a_1, s_2, a_2, \ldots, s_{t+1} | \pi)}{P(s_1, a_1, s_2, a_2, \ldots, s_{t+1} | \mu)} = \rho(a_1 | s_1) \ldots \rho(a_t | s_t)$$

How do excursions and alternative-life off-policy TD differ in practice?

Simple problem domain



- Behaviour policy is uniform random
- Target policy always chooses action A2

Updates for a trajectory

$$s \to 1 \to 1 \to 1 \to T$$

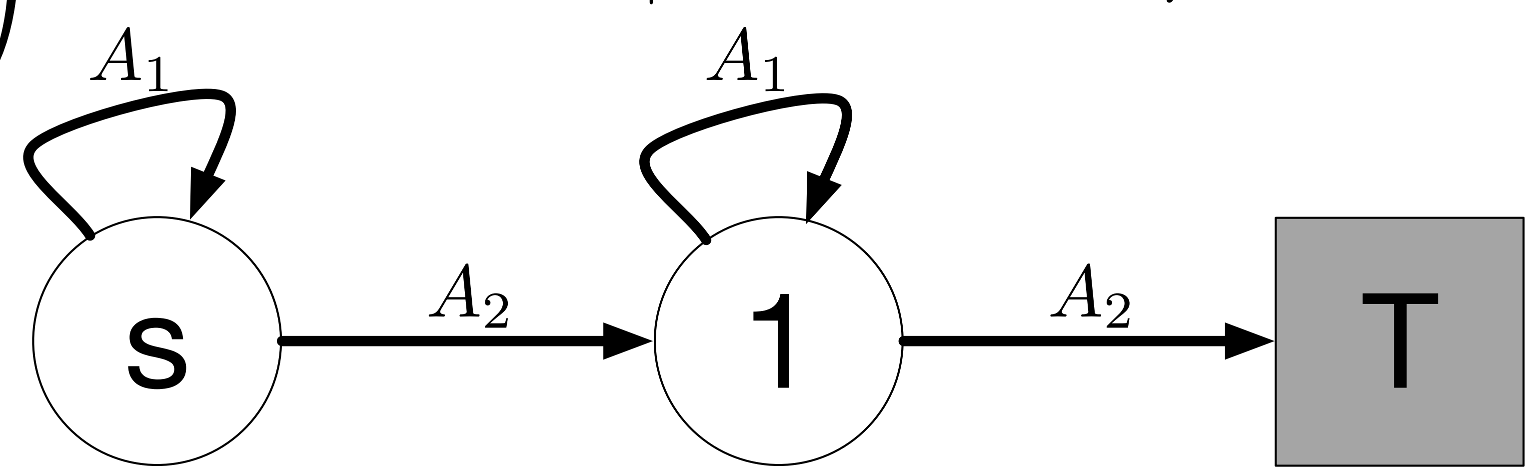
$$s_1 = s, a_1 = A_2, s_2 = 1, a_2 = A_1, s_3 = 1, a_3 = A_2, s_4 = T$$

$\rho_1 = 2$ $\rho_2 = 0$ $\rho_3 = 2$

Both update **Neither update** **Only excursions update**

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \alpha \left(\prod_{k=1}^{t} \rho_k \right) \delta_t \mathbf{x}_t$$

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \alpha \rho_t \delta_t \mathbf{x}_t$$



Consequences

- Including prior corrections is
  - likely high variance
  - unbiased
- Only using posterior corrections is
  - lower variance
  - optimizes a different objective (could say biased)

Exercise: Differences in the solution of the objectives

- Consider the on-policy objective and excursions objective

$$\Big\| \sum_{s \in \mathcal{S}} d_\pi(s) \mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s] \Big\|_2^2 \qquad \Big\| \sum_{s \in \mathcal{S}} d_\mu(s) \mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s] \Big\|_2^2$$

- What are the differences in:
  - the tabular setting?
  - powerful function class (e.g., expressive features)?
  - a limited function class (e.g., poor features, or linear in observations)?

Turn to a **different** person beside you and discuss with them

Tabular setting

$$\Big\| \sum_{s \in \mathcal{S}} d_\pi(s) \mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s] \Big\|_2^2 \qquad \Big\| \sum_{s \in \mathcal{S}} d_\mu(s) \mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s] \Big\|_2^2$$

- Because x(s) is an indicator vector

$$\mathbb{E}_\pi[\delta(S, A, S') | S = s]\mathbf{x}(s) = \mathbf{0} \iff \mathbb{E}_\pi[\delta(S, A, S') | S = s] = 0$$

- On-policy and excursions objectives are zero iff

$$\mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S) | S = s] = \mathbf{0} \quad \forall s$$

Impact of the weighting in the objective on the solution

- Gives more or less preference to having zero expected TD-error in a state, proportional to $d = d_\mu \text{ or } d_\pi$

$$\Big\| \sum_{s \in \mathcal{S}} d(s) \mathbb{E}_\pi [\delta(S, A, S') \mathbf{x}(S) | S = s] \Big\|_2^2$$

- What is the impact if have "good" features (expressive func. class)?
    - little need to trade-off accuracy between states, weighting not critical
- What is the impact if have "bad" features (limited func. class)?
    - need to trade-off accuracy, weighting could have a big impact

Summary so far

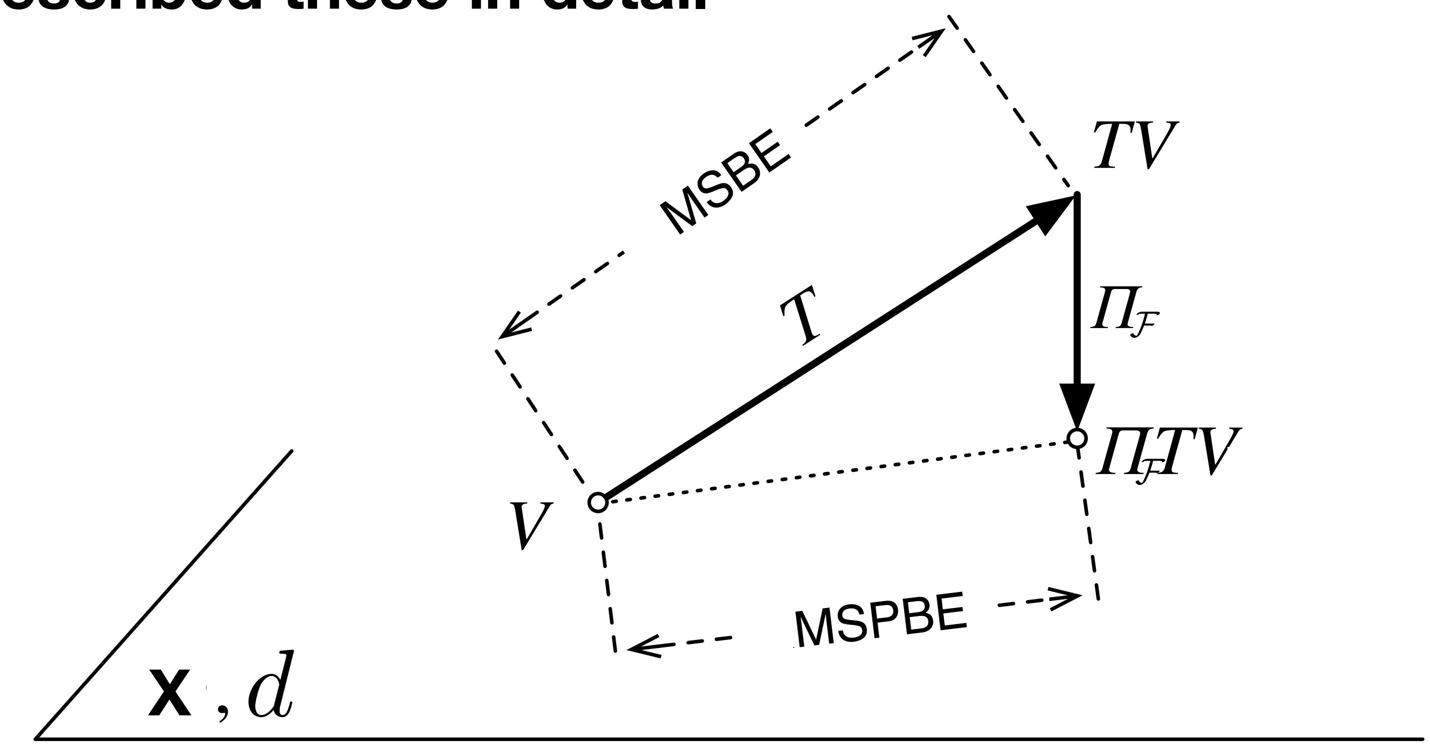
- We've talked about why off-policy learning is **important**
- We've talked about the different **objectives**
    - weighting (state probabilities) can impact final solution, but not as critical as adjusting action probabilities
- Any questions so far about these?
- **Now let's talk about the algorithms**

Does Off-Policy TD Converge?

- Convergence: in the limit of updates, with appropriately chosen stepsizes, does $\mathbf{w}_t \rightarrow \mathbf{w}^*$
- Underlying TD is a **Projected Bellman operator**
- Requirements on behaviour: fixed, support for $\boldsymbol{\pi}$ ✓
- Requirements on operator: contraction ✗

Projected Bellman Operator

***Amir massoud described these in detail**



$$\text{Here } \mathcal{F} = \{V : \mathcal{S} \to \mathbb{R} | V(s) = \mathbf{x}(s)^\top \mathbf{w} \text{ for some } \mathbf{w} \in \mathbb{R}^k\}$$

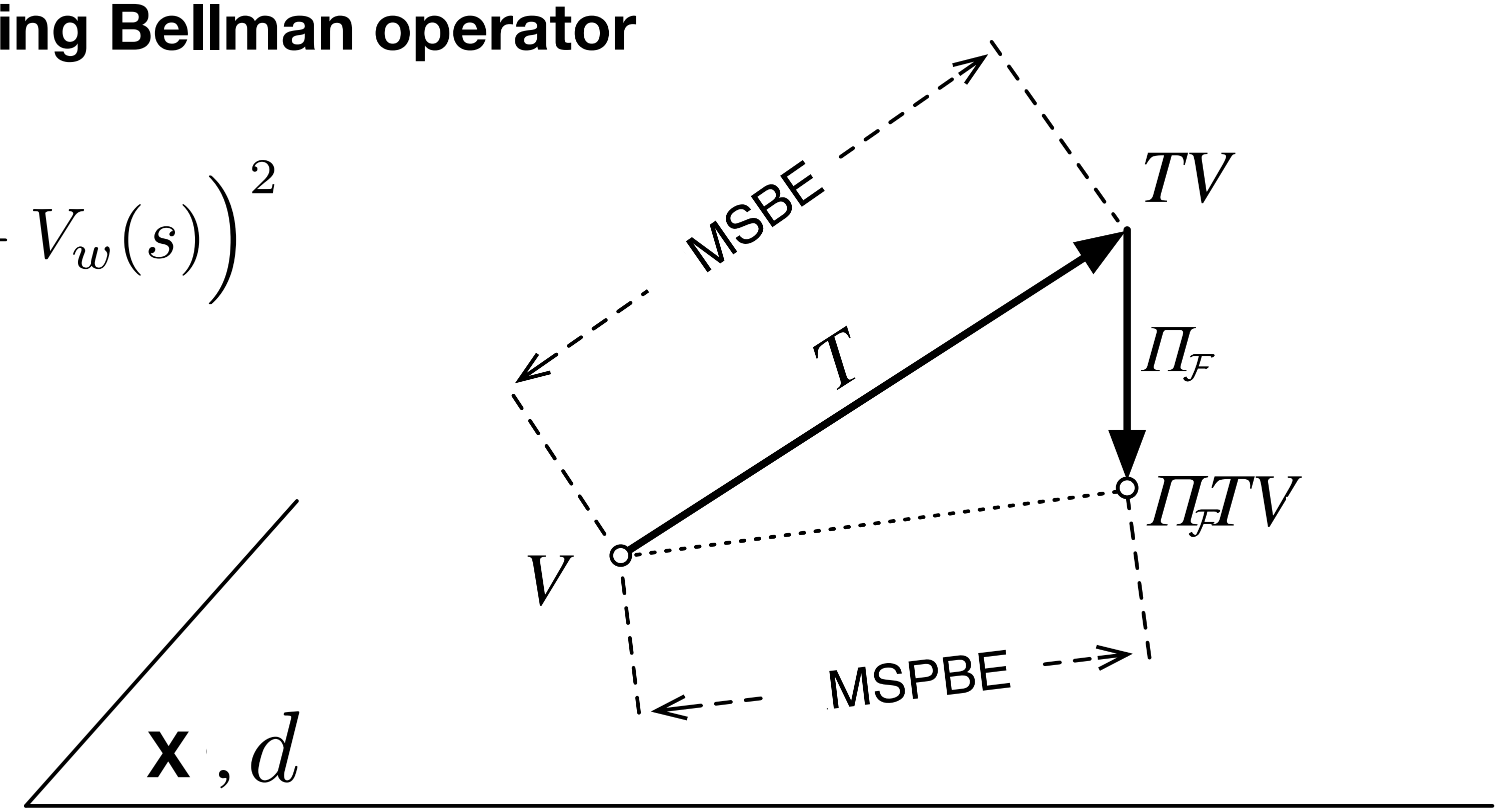
“Fast gradient-descent methods for temporal-difference learning with linear function approximation”, Sutton et al., 2011

Why is the Projected Bellman Operator related to TD?

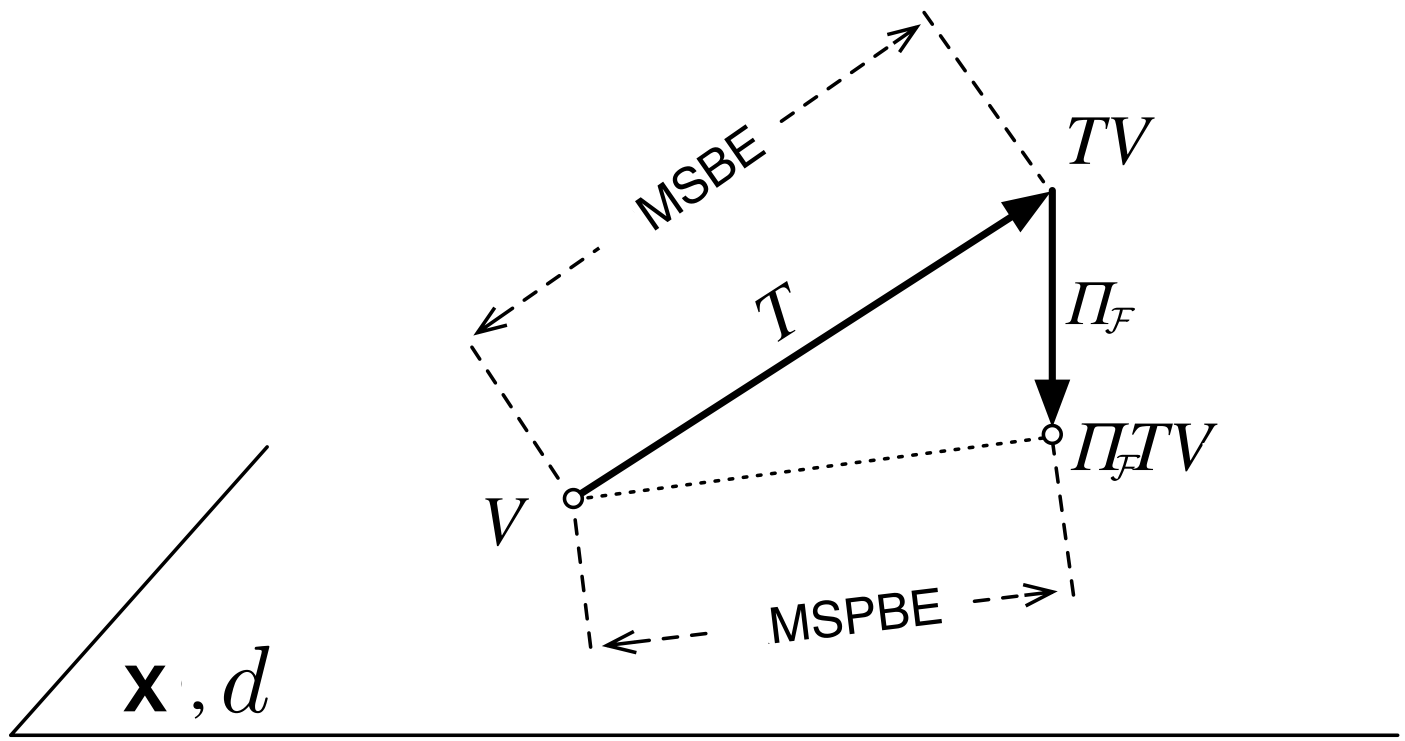
$$\mathbb{E}_\pi[\delta(S,A,S')|S=s] = \mathbb{E}_\pi[R(S,A,S') + \gamma V_w(S')|S=s]$$
$$= (TV_w)(s)$$

May not be representable by function class, so project after applying Bellman operator

$$\text{MSPBE}(\mathbf{w}) = \sum_s d(s)\Big((\Pi TV_w)(s) - V_w(s)\Big)^2$$



Projected Bellman Operator under Off-Policy sampling may not be a contraction



$$\text{Here } \mathcal{F} = \{V : \mathcal{S} \to \mathbb{R} | V(s) = \mathbf{x}(s)^\top \mathbf{w} \text{ for some } \mathbf{w} \in \mathbb{R}^k\}$$

If $d = d_\pi$ operator is a contraction, for linear value functions

If $d = d_\mu$ operator may not be a contraction!

“Fast gradient-descent methods for temporal-difference learning with linear function approximation”, Sutton et al., 2011

(Excursions) Off-Policy TD can diverge

- Several counter-examples exist
- The underlying expected update diverges
- If expected update diverges, the stochastic update (which just adds more noise) will diverge
- So now what?

A Brief History of Off-Policy Learning

- The promise of Q-learning was great!
  - Can generate data from any behaviour policy, to learn optimal policies for other tasks
- In 90s, several divergence examples shown
- In following years, used Alternative-Life TD to fix these
- **Issues**: convergence counter-examples (Q-learning) and variance problems (off-policy TD with products of ρ)

Reasons for problems

- Without prior corrections, the projected Bellman operator may not be a contraction
    - issue for iterated fixed-point approaches, like off-policy TD
- With prior corrections, can have infinite variance
    - “Least Squares Temporal Difference Methods: An Analysis Under General Conditions”, Yu, 2010

Solutions

- **Convergence without prior corrections:** Using gradient-based approaches (on the MSPBE)
  - Gradient TD and related methods
- **Lower-variance prior corrections:** Incorporating prior corrections into the excursions formulation
  - Emphatic TD

Difficulties in getting the gradient of the MSPBE

$$\mathbf{a}(\mathbf{w}) \stackrel{\text{def}}{=} \sum_s d_\mu(s)\mathbb{E}_\pi[\delta(S, A, S')\mathbf{x}(S)|S = s]$$

$$\mathbf{C} \stackrel{\text{def}}{=} \sum_s d_\mu(s)\mathbf{x}(s)\mathbf{x}(s)^\top$$

Contrast with

$$\nabla\mathbb{E}[(\mathbf{x}^\top\mathbf{w} - y)^2]$$
$$= \nabla 2\mathbb{E}[(\mathbf{x}^\top\mathbf{w} - y)\mathbf{x}]$$

$$\text{MSPBE}(\mathbf{w}) = \mathbf{a}(\mathbf{w})^\top\mathbf{C}^{-1}\mathbf{a}(\mathbf{w})$$

$$\nabla\text{MSPBE}(\mathbf{w}) = 2\left(\nabla\mathbf{a}(\mathbf{w})\right)^\top\mathbf{C}^{-1}\mathbf{a}(\mathbf{w})$$

Double sampling problem: $\mathbb{E}[XY] \neq \mathbb{E}[X]\mathbb{E}[Y]$

Need independent samples for a(w) and ∇a(w)

Potential solutions

- Quadratic cost algorithms, like LSTD

- Gradient TD:

  $$\text{Incrementally estimate } \mathbf{h} = \mathbf{C}^{-1}\mathbf{a}(\mathbf{w})$$

  $$\text{Use } (\mathbf{x}_t - \gamma_{t+1}\mathbf{x}_{t+1})\mathbf{x}_t^\top \text{ as an unbiased sample of } -\nabla\mathbf{a}(\mathbf{w})^\top$$

  $$\text{Update weights with } (\mathbf{x}_t - \gamma_{t+1}\mathbf{x}_{t+1})\mathbf{x}_t^\top\mathbf{h}$$

- Proximal methods: reformulate as a saddlepoint problem which no longer has products of expectations

$$\text{MSPBE}(\mathbf{w}) = \max_{\mathbf{h}} \ \mathbf{h}^\top\mathbf{a}(\mathbf{w}) - \|\mathbf{h}\|_2^2$$

"Finite-Sample Analysis of Proximal Gradient TD Algorithms", Liu et al., 2015

Other algorithms

- Two timescale approaches
  - TDC (also called GTD) - more popular GTD variant
  - ABQ (or ABTD) - “Multi-step Off-policy Learning Without Importance Sampling Ratios”, Mahmood et al., 2017
- Saddlepoint methods (proximal methods)
  - SVRG approach - “Stochastic Variance Reduction Methods for Policy Evaluation”, Du et al, 2017
  - GTB and GRetrace - “Convergent Tree-Backup and Retrace with Function Approximation”, Touati et al. 2018

Emphatic TD

- Incorporate prior corrections into excursions model
- Starting from states under behaviour policy, correct the visitation **for the excursion**
  - rather than since the beginning of the episode or since the beginning of time
- Obtain a different weighting than d$\boldsymbol{\mu}$ or d$\boldsymbol{\pi}$,

Emphatic Weighting

$$f(s') = d_\mu(s') + \gamma \sum_{s,a} d_\mu(s)\pi(a|s)P(s,a,s') + \ldots$$

- MSPBE has weighted expected TD-error

$$\sum_s f(s)\mathbb{E}_\pi[\delta(S,A,S')\mathbf{x}(S)|S=s]$$

- Emphatic algorithm uses estimate

$$F_t = \gamma\rho_{t-1}F_{t-1} + 1$$

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \alpha F_t \delta_t \mathbf{x}_t$$

What does all this mean for nonlinear function approximation?

- We’ve talked about linear value function approximation
  - Doesn’t have to be linear in inputs, but has to be for a set of fixed features
- For nonlinear functions (e.g., neural networks), the projection operator is different —> MSPBE is different
  - need to use the nonlinear MSPBE
  - choice of weightings (d$\boldsymbol{\mu}$, d$\boldsymbol{\pi}$, f) still applies

Exercise: Designing an off-policy learning system

- Think of a setting where you might use off-policy learning
- What choices will you have to consider?
- What properties might you care about for the algorithms?
- How does your new understanding impact how you approach nonlinear function approximation?

Turn to a **different** person beside you and discuss with them (in twos or threes)

Additional topics

- Connection to a different off-policy policy evaluation
- Extensions to eligibility traces
- Policy gradient methods

Off-Policy PE

- Goal is to get the value of a policy, v($\boldsymbol{\pi}$)

$$v(\pi) = \sum_{s \in \mathcal{S}} d(s) V^{\pi}(s)$$

- Data is generated from a different behaviour policy

- May not need to estimate $V^{\boldsymbol{\pi}}$ to get v($\boldsymbol{\pi}$)