AIL 722: Reinforcement Learning

Lecture 19: Temporal-Difference Control

Raunak Bhattacharyya

ScAI YARDI SCHOOL OF ARTIFICIAL INTELLIGENCE
INDIAN INSTITUTE OF TECHNOLOGY DELHI

Recap

- Off-Policy Monte Carlo Control
- Importance sampling ratios: Unweighted and Weighted
- Incremental value estimation using weighted IS
- Algorithm

Outline

- Back to TD
- On-policy TD control: SARSA
- Off-policy TD control: Q-Learning

Monte Carlo Control: On-policy vs. Off-policy

On-policy first-visit MC control (for ε-soft policies), estimates $\pi \approx \pi_*$

Algorithm parameter: small $\varepsilon > 0$
Initialize:
$\pi \leftarrow$ an arbitrary ε-soft policy
$Q(s,a) \in \mathbb{R}$ (arbitrarily), for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$
$Returns(s,a) \leftarrow$ empty list, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

Repeat forever (for each episode):
Generate an episode following π: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
$G \leftarrow 0$
Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:
$G \leftarrow \gamma G + R_{t+1}$
Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1 \ldots, S_{t-1}, A_{t-1}$:
Append G to $Returns(S_t, A_t)$
$Q(S_t, A_t) \leftarrow \text{average}(Returns(S_t, A_t))$
$A^* \leftarrow \arg\max_a Q(S_t, a)$ (with ties broken arbitrarily)
For all $a \in \mathcal{A}(S_t)$:

$$\pi(a|S_t) \leftarrow \begin{cases} 1 - \varepsilon + \varepsilon/|\mathcal{A}(S_t)| & \text{if } a = A^* \\ \varepsilon/|\mathcal{A}(S_t)| & \text{if } a \neq A^* \end{cases}$$

Off-policy MC control, for estimating $\pi \approx \pi_*$

Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
$Q(s,a) \in \mathbb{R}$ (arbitrarily)
$C(s,a) \leftarrow 0$
$\pi(s) \leftarrow \arg\max_a Q(s,a)$ (with ties broken consistently)

Loop forever (for each episode):
$b \leftarrow$ any soft policy
Generate an episode using b: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
$G \leftarrow 0$
$W \leftarrow 1$
Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:
$G \leftarrow \gamma G + R_{t+1}$
$C(S_t, A_t) \leftarrow C(S_t, A_t) + W$
$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{W}{C(S_t, A_t)} [G - Q(S_t, A_t)]$
$\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$ (with ties broken consistently)
If $A_t \neq \pi(S_t)$ then exit inner Loop (proceed to next episode)
$W \leftarrow W \frac{1}{b(A_t|S_t)}$

Targets for MC, DP and TD

$$V^{\pi}(s^j) = \mathbb{E}_{p_\theta(\tau)}\left[G_t \middle| s_t = s^j\right]$$

Single sample return instead of real expected return

$$V^{\pi}(s^j) = \mathbb{E}_{p_\theta(\tau)}\left[r_{t+1} + \gamma G_{t+1} \middle| s_t = s^j\right]$$

$$V^{\pi}(s^j) = \mathbb{E}_{p_\theta(\tau)}\left[r_{t+1} + \gamma \cdot V^{\pi}(s_{t+1}) \middle| s_t = s^j\right]$$

True V^pi not known and current estimate used instead

Back to Temporal-Difference

Recall our journey between Monte Carlo and Temporal-Difference

How do we learn a state-action value function?

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

How does this compare to the MC approach?

SARSA

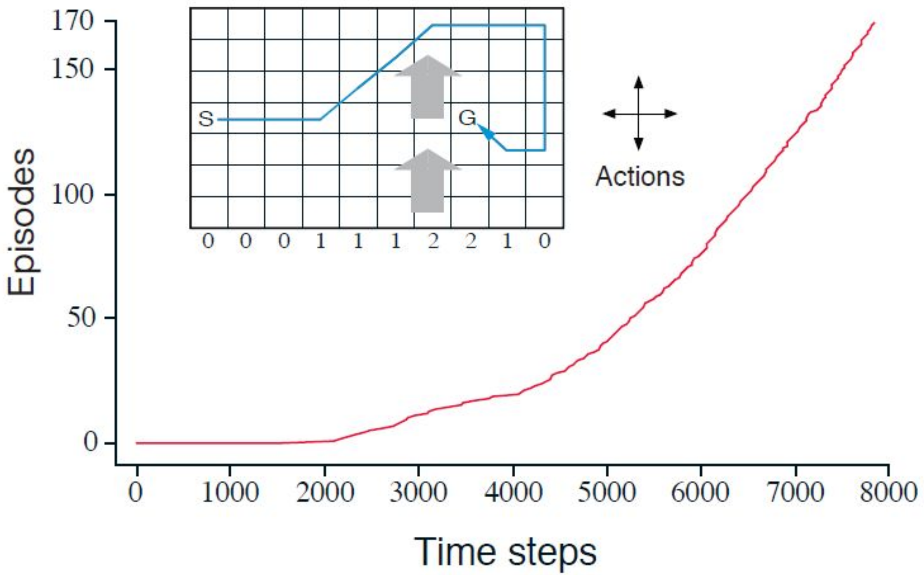
Sarsa (on-policy TD control) for estimating $Q \approx q_*$

Algorithm parameters: step size $\alpha \in (0, 1]$, small $\varepsilon > 0$
Initialize $Q(s, a)$, for all $s \in \mathcal{S}^+, a \in \mathcal{A}(s)$, arbitrarily except that $Q(terminal, \cdot) = 0$

Loop for each episode:
    Initialize S
    Choose A from S using policy derived from Q (e.g., ε-greedy)
    Loop for each step of episode:
        Take action A, observe R, S'
        Choose A' from S' using policy derived from Q (e.g., ε-greedy)
        $Q(S, A) \leftarrow Q(S, A) + \alpha\big[R + \gamma Q(S', A') - Q(S, A)\big]$
        $S \leftarrow S'$; $A \leftarrow A'$;
    until S is terminal

Section 6.4, Reinforcement Learning: An Introduction, Sutton & Barto

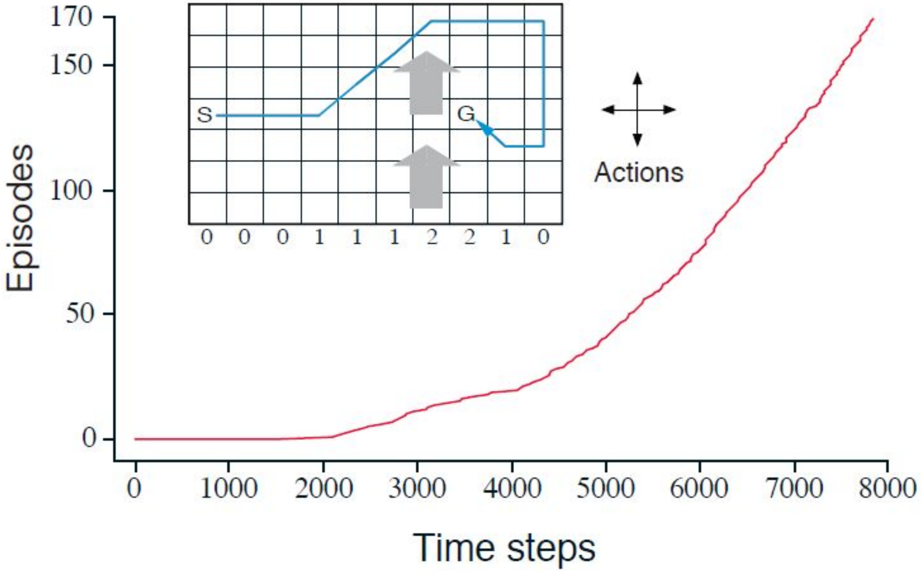
Windy Gridworld: Description



- 4 det actions
- Upward crosswind
  - Strength below grid
- Undiscounted episodic task
  - Reward -1 each timestep until G

Example 6.5, [Reinforcement Learning: An Introduction](), Sutton & Barto

Windy Gridworld: Solution



- SARSA
  - Eps: 0.1
  - Alpha: 0.5
  - Q init: 0

Why MC may not be suitable in this example?

Termination not guaranteed for all policies

What does the increasing slope tell us?

Example 6.5, [Reinforcement Learning: An Introduction](), Sutton & Barto

Q-Learning

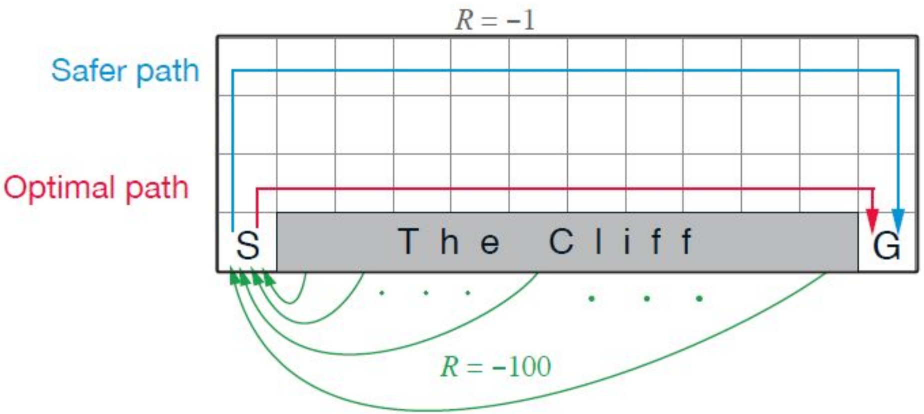
Q-learning (off-policy TD control) for estimating $\pi \approx \pi_*$

Algorithm parameters: step size $\alpha \in (0, 1]$, small $\varepsilon > 0$
Initialize $Q(s, a)$, for all $s \in \mathcal{S}^+, a \in \mathcal{A}(s)$, arbitrarily except that $Q(\mathit{terminal}, \cdot) = 0$

Loop for each episode:
  Initialize S
  Loop for each step of episode:
    Choose A from S using policy derived from Q (e.g., ε-greedy)
    Take action A, observe R, S'
    $Q(S, A) \leftarrow Q(S, A) + \alpha\big[R + \gamma \max_a Q(S', a) - Q(S, A)\big]$
    $S \leftarrow S'$
  until S is terminal

Section 6.5, Reinforcement Learning: An Introduction, Sutton & Barto

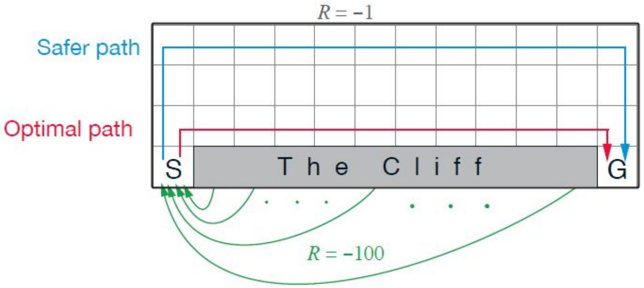
Cliff Walking

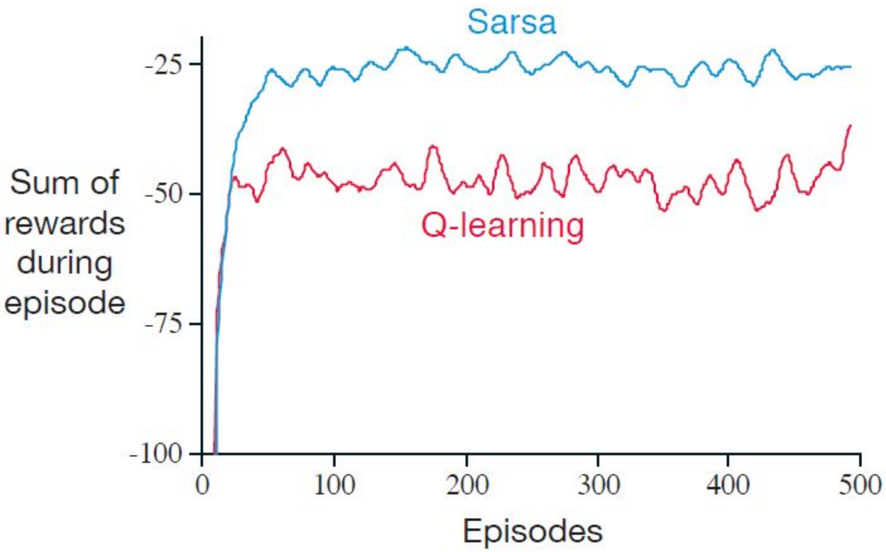


- 4 det actions
- Undiscounted episodic task
  - Reward -1 each timestep
  - Reward -1 cliff and agent back to S

Example 6.6, Reinforcement Learning: An Introduction, Sutton & Barto

Cliff Walking





Why do Q-learning and SARSA learn different policies?

Example 6.6, Reinforcement Learning: An Introduction, Sutton & Barto

Implementation: Gym

Gym Documentation

Search

INTRODUCTION

Basic Usage

API

Core

Spaces

Wrappers

Vector

Utils

ENVIRONMENTS

Atari

MuJoCo

Toy Text

Classic Control

Box2D

Third Party Environments

Basic Usage

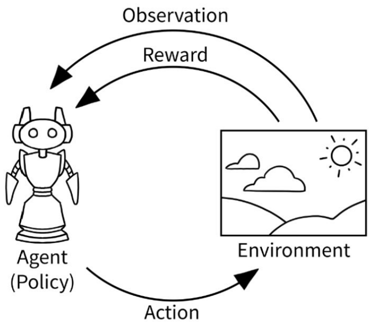
Initializing Environments

Initializing environments is very easy in Gym and can be done via:

```
import gym
env = gym.make('CartPole-v0')
```

Interacting with the Environment

Gym implements the classic “agent-environment loop”:



The agent performs some actions in the environment (usually by passing some control inputs to the environment, e.g. torque inputs of motors) and observes how the environment’s state changes. One such action-observation exchange is referred to as a *timestep*.

Summary & Announcements

- Summary
  - TD control
  - SARSA
  - Q-Learning

- Announcements
  - Sign up for Assgn 1 viva slots
    - To be held this Sat, 7/9/24

[Viva sign up link]