Online and Reinforcement Learning

2024-2025

Home Assignment 2

Sadegh Talebi
Department of Computer Science
University of Copenhagen

The deadline for this assignment is **19 February 2025, 20:59**. You must submit your *individual* solution electronically via the Absalon home page.

A solution consists of:

- A PDF file with detailed answers to the questions, which may include graphs and tables if needed. Do *not* include your full source code in the PDF file, only selected lines if you are asked to do so.
- A .zip file with all your solution source code with comments about the major steps involved in each question (see below). Source code must be submitted in the original file format, not as PDF. The programming language of the course is Python.

Important Remarks:

- IMPORTANT: Do NOT zip the PDF file, since zipped files cannot be opened in *SpeedGrader*. Zipped PDF submissions will not be graded.
- Your PDF report should be self-sufficient. I.e., it should be possible to grade it without opening the .zip file. We do not guarantee opening the .zip file when grading.
- Your code should be structured such that there is one main file (or one main file per question) that we can run to reproduce all the results presented in your report. This main file can, if you like, call other files with functions, classes, etc.
- Handwritten solutions will not be accepted.

1 Short Questions (8 points) [Sadegh]

Determine whether each statement below is True or False and provide a very brief justification.

1. In a finite discounted MDP, every possible policy induces a Markov Reward Process.

   ☐ True ☐ False

   Justification:

2. Consider a finite discounted MDP, and assume that π is an optimal policy. Then, the action(s) output by π does not depend on history other than the current state (i.e., π is necessarily stationary).

   ☐ True ☐ False

   Justification:

3. In a finite discounted MDP, a greedy policy with respect to optimal action-value function, $Q^\star$, corresponds to an optimal policy.

   ☐ True ☐ False

   Justification:

4. Under the coverage assumption, the Weighted Importance Sampling Estimator $\widehat{V}_{\text{wIS}}$ converges to V^π with probability 1.

   ☐ True ☐ False

   Justification:

2 MDPs with Similar Parameters Have Similar Values (22 points) [Sadegh]

In this exercise, we study a classical result that concerns the difference in value functions between two MDPs that are defined on the same state-action space, and whose transition and reward functions are close in some sense.

Consider two finite discounted MDPs $M_1 = (\mathcal{S}, \mathcal{A}, P_1, R_1, \gamma)$ and $M_2 = (\mathcal{S}, \mathcal{A}, P_2, R_2, \gamma)$. Assume the two reward functions take values in the range $[0, R_{\max}]$. Suppose that for all state-action pairs (s, a),

$$|R_1(s,a) - R_2(s,a)| \leq \alpha, \qquad \|P_1(\cdot|s,a) - P_2(\cdot|s,a)\|_1 \leq \beta$$

for some numbers $\alpha > 0$ and $\beta > 0$. Then, for all stationary deterministic policy π and state-action pair (s, a),

$$\text{(i):} \quad |Q_1^\pi(s,a) - Q_2^\pi(s,a)| \leq \frac{\alpha + \gamma R_{\max}\beta}{(1-\gamma)^2},$$
$$\text{(ii):} \quad |V_1^\pi(s) - V_2^\pi(s)| \leq \frac{\alpha + \gamma R_{\max}\beta}{(1-\gamma)^2}.$$

Prove either (i) or (ii).

3 Policy Evaluation in RiverSwim (20 points) [Sadegh]

Consider the 5-state RiverSwim MDP with $\gamma = 0.96$ (see Figure 1 with $L = 5$). Consider policy π defined as:

$$\pi(s) = \begin{cases} \mathsf{right} & \text{w.p. } 0.65 \\ \mathsf{left} & \text{w.p. } 0.35 \end{cases} \qquad s = 1, 2, 3,$$
$$\pi(s) = \mathsf{right} \qquad s = 4, 5.$$

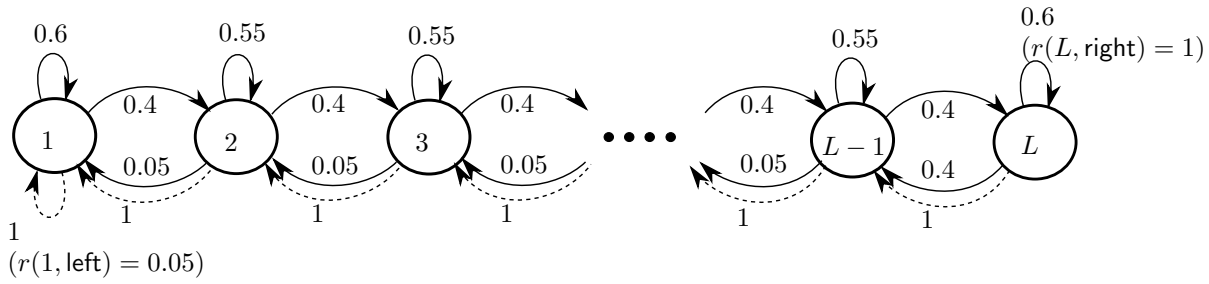

Figure 1: The L-state RiverSwim MDP

(i) Compute V^π using a Monte Carlo simulation as follows. For each state s, generate n trajectories of length T via interacting with the MDP with starting state $s_1 = s$. Let $h^i = (s_1^i, a_1^i, r_1^i, \ldots, s_T^i, a_T^i, r_T^i)$ denote the i-th trajectory (and note that $s_1^i = s$). Then, $\widehat{V}^\pi(s)$ defined as

$$\widehat{V}^\pi(s) = \frac{1}{n} \sum_{i=1}^{n} \sum_{t=1}^{T} \gamma^{t-1} r_t^i$$

is a Monte Carlo approximation to $V^\pi(s)$. Compute a Monte Carlo approximation to V^π using $T = 300$ and $n = 50$.

(Note: You can make use of the Python implementation of RiverSwim provided in Absalon.)

(ii) Compute the exact value of V^π (using direct computation) and compare it with the result of Part (i).

4 Solving a Discounted Grid-World (25 points) [Sadegh]

In this exercise, we model a grid-world game as a discounted MDP, and solve it using `PI` and `VI`.

Consider the 4-room Grid-World MDP shown in Figure 2. It is made of a grid of size 7×7, which has $S = 20$ accessible states (after removing walls). The agent starts in

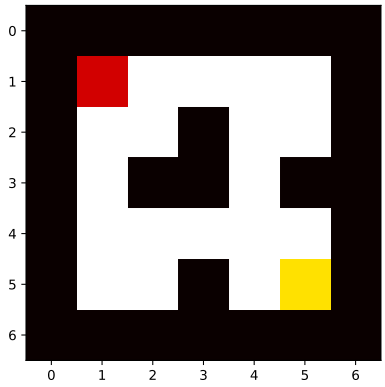

Figure 2: The 4-Room Grid-World MDP

the upper-left corner (shown in red). A reward of 1 is placed in the lower-right corner (shown in yellow), and the rest of the states give no reward. Once in the rewarding state (in yellow), the agent stays there forever (and continues receiving the reward). The agent can perform the 4 compass actions going up, left, down, or right (of course, when away from walls). However, the floor is slippery and brings stochasticity to the next-state. Specifically, under each of the four aforementioned actions, she moves in the chosen direction (with probability 0.7), stays in the same state (with probability 0.1), or goes in each of the two perpendicular directions (each with probability 0.1) —this environment is sometimes referred to as the *frozen lake* MDP. Walls act as reflectors, i.e., they cause moving back to the current state. A Python implementation of this MDP is provided in `HA2_gridworld.py`. Rewards are discounted with rate $\gamma = 0.97$.

We can model this task as a discounted MDP.

(i) Solve the grid-world task above using `PI`. (You may use the Python implementation of `PI` provided in the same file.) Report an optimal policy along with the optimal value function $V^\star$. Furthermore, visualize the derived optimal policy using arrows in the figure or by arranging it using a suitably defined matrix.

(ii) Implement `VI` and use it to solve the grid-world task above. Your implementation should receive an MDP M and an accuracy parameter ε as input, and output a policy and the corresponding value. (Note that to ensure that `VI` returns an optimal policy, ε must be sufficiently small; here, $\varepsilon = 10^{-6}$ suffices.)

(iii) Repeat Part (ii) with $\gamma = 0.998$ and discuss how this new discount affects the convergence speed of `VI`.

Now we would like to try a new accelerated variant of `VI`, called *Anchored Value Iteration* `Anc-VI`, which is recently proposed and analyzed in Lee and Ryu (2024). `Anc-VI` is similar to the classical `VI`, but relies on a more sophisticated value update that is inspired by *the anchor mechanism* developed for general non-expansive (and contraction) operators.

Specifically, the update under `Anc-VI` is:

$$V_{n+1} = \beta_{n+1} V_0 + (1 - \beta_{n+1}) \mathcal{T} V_n,$$

for $n = 0, 1, 2, \ldots$, where V_0 is an initial point and $\beta_n = 1/(\sum_{i=0}^{n} \gamma^{-2i})$. Recall that $\mathcal{T}$ denotes the optimal Bellman operator.

The term $\beta_n V_0$, called *anchor*, serves to pull the iterates toward the starting point V_0. The strength of the anchor mechanism diminishes as n grows since $(\beta_n)_{n \geq 1}$ is a decreasing sequence. In other words, the role of the optimal Bellman operator becomes increasingly more salient.

(iv) Implement `Anc-VI` and use it to solve the grid-world task above. Consider three different choices of initial point: (a) $V_0 = \mathbf{0}$, (b) $V_0 = \mathbf{1}$, and (c) V_0 sampled uniformly at random (i.e., for each s, $V_0(s)$ is sampled from uniform distribution supported on $[0, 1/(1-\gamma)]$).

(v) Compare the convergence speed of `VI` and `Anc-VI` (with different choices of V_0).

5 Off-Policy Evaluation in Episode-Based RiverSwim `(25 points) [Sadegh]`

In this exercise, we examine off-policy evaluation in a variant of RiverSwim, which we call *episode-based* RiverSwim.[1]

An L-state episode-based RiverSwim is quite similar to the conventional RiverSwim, *except* that: (i) the state L (corresponding to the right bank of the river) is a terminal state with no action. When in L, the agent receives a terminal reward of $R(L) = 1$ and the episode ends; (ii) Each episode starts in state 1 (corresponding to the left bank). Finally, suppose that if an episode ends in time t, a new one begins in the next slot $t+1$.

Consider the 6-state episode-based RiverSwim MDP, as explained above, with $\gamma = 0.96$. Consider a dataset $\mathcal{D}$ (provided via `dataset0.csv`) generated according to π_{b} specified as: For all $s \in \mathcal{S}$, $\pi_{\mathrm{b}}(s) = \begin{cases} \mathsf{right} & \text{w.p. } 0.65 \\ \mathsf{left} & \text{w.p. } 0.35. \end{cases}$ The dataset above comprises 200 episodes, but the data is reported as a long trajectory where all episodes are concatenated.

We wish to estimate $V^{\pi}(1)$ using $\mathcal{D}$, with π being the policy prescribing to take the right action in all non-terminal states.

(i) Estimate $V^{\pi}(1)$ using the following methods: `MB-OPE`, `IS`, `wIS`, and `PDIS`. As output, you should report 4 value estimates.

(ii) For each method, plot the error $|V^{\pi}(1) - \widehat{V}_t^{\pi}(1)|$ as a function of t, where $V^{\pi} = (I - \gamma P^{\pi})^{-1} r^{\pi}$ denotes the true value function of π. (When computing V^{π}, note that the MDP here has a slightly different transition function than classical RiverSwim.)

[1] We refrain from calling it *episodic RiverSwim* to avoid conflicting with the notion of episodic MDPs where the duration of the episode is a fixed number.

(iii) Now consider 9 additional datasets `dataset1.csv`,...,`dataset9.csv` generated in a similar fashion. For a given OPE method, each dataset yields an estimate of $V^\pi(1)$. Report the empirical variance of these estimates across the 10 datasets for each of the methods mentioned in Part (i).

(iv) Compare the 4 methods above in terms of empirical error and variance.

6 Bounds on H-step Values `(0 points)` `[Sadegh]` (Optional)

Consider a discounted MDP M with rewards supported on $[0, 1]$. Let's define the H-step value function of a policy π, for a given $H \in \mathbb{N}$, as

$$U^{\pi,H}(s) = \mathbb{E}^\pi\Big[\sum_{t=1}^{H} \gamma^{t-1} r_t \Big| s_1 = s\Big], \quad \forall s \in \mathcal{S}.$$

(i) Given $\varepsilon > 0$, determine values of H such that $|U^{\pi,H}(s) - V^\pi(s)| \leq \varepsilon$ for all s and for all π.

(ii) How do you interpret the derived bound?

References

Jongmin Lee and Ernest Ryu. Accelerating value iteration with anchoring. *Advances in Neural Information Processing Systems*, 36, 2024.