Online and Reinforcement Learning 2023-2024

Christian Igel Yevgeny Seldin Sadegh Talebi

Home Assignment 1

Deadline: 21:00, Wednesday, 14 February 2024

The assignments must be answered individually - each student must write and submit his/her own solution. We encourage you to work on the assignments on your own, but we do not prevent you from discussing the questions in small groups. If you do so, you are requested to list the group partners in your individual submission.

Submission format: *Please, upload your answers in a single* `.pdf` *file and additional* `.zip` *file with all the code that you used to solve the assignment. (The* `.pdf` *should* ***not*** *be part of the* `.zip` *file.)*

IMPORTANT: *We are interested in how you solve the problems, not in the final answers. Please, write down the calculations and comment your solutions.*

1 Follow The Leader (FTL) algorithm for i.i.d. full information games (25 points) [Yevgeny]

Follow the leader (FTL) is a playing strategy that at round t plays the action that was most successful up to round t ("the leader"). Derive a bound for the pseudo regret of FTL in i.i.d. full information games with K possible actions and outcomes bounded in the $[0,1]$ interval (you can work with rewards or losses, as you like). You can use the following guidelines (which assume a game with rewards):

1. You are allowed to solve the problem for $K = 2$. (The guidelines are not limited to $K = 2$.)

2. It may be helpful to write the algorithm down explicitly. For the analysis it does not matter how you decide to break ties.

3. Let $\mu(a)$ be expected reward of action a and let $\hat{\mu}_t(a)$ be empirical estimate of the reward of action a at round t (the average of rewards observed so far). Let a^* be an optimal action (there may be more than one optimal action, but then things only get better [convince yourself that this is true], so we can assume that there is a single a^*). Let $\Delta(a) = \mu(a^*) - \mu(a)$. FTL may play $a \neq a^*$ at rounds t for which $\hat{\mu}_{t-1}(a) \geq \max_{a'} \hat{\mu}_{t-1}(a')$ (in the case of two arms it means $\hat{\mu}_{t-1}(a) \geq \hat{\mu}_{t-1}(a^*)$). So you should analyze how often this may happen.

4. Note that the number of times an action a was played can be written as $N_T(a) = \sum_{t=1}^{T} \mathbb{1}(A_t = a)$, where $\mathbb{1}$ is the indicator function, and that $\mathbb{E}\left[\mathbb{1}(A_t = a)\right] \leq \mathbb{P}(\hat{\mu}_{t-1}(a) \geq \max_{a'} \hat{\mu}_{t-1}(a')) \leq \mathbb{P}(\hat{\mu}_{t-1}(a) \geq \hat{\mu}_{t-1}(a^*))$.

5. Bound $\mathbb{P}(\hat{\mu}_{t-1}(a) \geq \hat{\mu}_{t-1}(a^*))$.

6. At some point in the proof you will need to sum up a geometric series. A geometric series is a series of a form $\sum_{t=0}^{\infty} r^t$, and for $r < 1$ we have $\sum_{t=0}^{\infty} r^t = \frac{1}{1-r}$. In your case r will be an exponent $r = e^{\alpha}$ for some constant α.

7. At the end you should get a bound of a form $\bar{R}_T \leq \sum_{a:\Delta(a)>0} \frac{c}{1-\exp(-\Delta(a)^2/2)} \Delta(a)$, where c is a constant.

Important observations to make:

1. Note that in the full information i.i.d. setting the regret does not grow with time!!! (Since the bound is independent of T.)
2. Note that even though you have used $\Delta(a)$ in the analysis of the algorithm, you do not need to know it in order to define the algorithm! I.e., you can run the algorithm even if you do not know $\Delta(a)$.

2 Improved Parametrization of UCB1 (25 points) [Yevgeny]

In this question we improve the UCB1 algorithm presented in the lecture notes.

1. [Optional, 0 points] Show that if we define the upper confidence bounds in UCB1 by

$$\hat{\mu}_{t-1}(a) + \sqrt{\frac{\ln t}{N_{t-1}(a)}}$$

   then its pseudo-regret satisfies

$$\bar{R}_T \leq 4 \sum_{a:\Delta(a)>0} \frac{\ln T}{\Delta(a)} + (2\ln(T) + 3) \sum_a \Delta(a).$$

   Hint: The T-th harmonic number, $\sum_{t=1}^T \frac{1}{t}$, satisfies $\sum_{t=1}^T \frac{1}{t} \leq \ln(T) + 1$.

2. Write a simulation to compare numerically the performance of UCB1 from the lecture notes with performance of UCB1 with modified confidence intervals proposed above. Instructions for the simulation:
   - Generate Bernoulli rewards for two actions, a^* and a, so that $\mathbb{E}\left[r_{t,a^*}\right] = \frac{1}{2} + \frac{1}{2}\Delta$ and $\mathbb{E}\left[r_{t,a}\right] = \frac{1}{2} - \frac{1}{2}\Delta$. (The rewards may be generated dynamically as you run the algorithms and, actually, you only need them for the actions that are played by the algorithms.)
   - Run the experiment with $\Delta = \frac{1}{4}$, $\Delta = \frac{1}{8}$, and $\Delta = \frac{1}{16}$. (Three different experiments.)
   - Take $T = 100000$. (In general, the time horizon should be large in relation to $\frac{1}{\Delta^2}$.)
   - Plot the empirical pseudo regret defined by $\hat{R}_t = \sum_{s=1}^t \Delta(A_s)$ for the two algorithms as a function of time for $1 \leq t \leq T$. (To remind you: A_s is the action taken by the algorithm in round s and $\Delta(a) = \max_{a'} \mathbb{E}\left[r_{s,a'}\right] - \mathbb{E}\left[r_{s,a}\right]$.) To make the plot you should make 20 runs of each algorithm and plot the average pseudo regret over the 20 runs and the average pseudo regret + one standard deviation over the 20 runs. Do not forget to add a legend to your plot.
   - Answer the following questions:
     - Which values of Δ lead to higher regret?
     - What can you say about the relative performance of the two parametrizations?

Comment: The UCB1 algorithm in the lecture notes takes confidence intervals $\sqrt{\frac{3\ln t}{2N_{t-1}(a)}} = \sqrt{\frac{\ln t^3}{2N_{t-1}(a)}}$, corresponding to confidence parameter $\delta = \frac{1}{t^3}$. The modified UCB1 algorithm in this question takes confidence intervals $\sqrt{\frac{2\ln t}{2N_{t-1}(a)}} = \sqrt{\frac{\ln t^2}{2N_{t-1}(a)}}$, corresponding to confidence parameter $\delta = \frac{1}{t^2}$. The original algorithm and analysis by Auer et al. (2002) uses confidence intervals $\sqrt{\frac{4\ln t}{2N_{t-1}(a)}} = \sqrt{\frac{\ln t^4}{2N_{t-1}(a)}}$, corresponding to confidence parameter $\delta = \frac{1}{t^4}$, due to one unnecessary union bound. The reason we can move from $\delta = \frac{1}{t^3}$ to $\delta = \frac{1}{\delta^2}$ is not due to elimination of additional union bounds (we still need two of them), but due to a compromise on the last term in the regret bound.

3 Example of Policies in RiverSwim (10 points) [Sadegh]

Consider the following policies defined in the RiverSwim MDP (Figure 1). For each case, determine to which class the policy belongs (i.e., $\Pi^{\mathsf{SD}}, \Pi^{\mathsf{SR}}, \Pi^{\mathsf{HD}}, \Pi^{\mathsf{HR}}$). Provide a short explanation and state any assumptions you may make.

(i) π_a defined as: Swim to right if the current state is different than s_1; otherwise swim to left.

(ii) π_b defined as: If t is even, Swim to right; otherwise, flip a fair coin, then swim to right (resp. left) if the outcome is 'Head' (resp. 'Tail').

(iii) π_c defined as: Swim to right if the index of the previous state is odd; otherwise swim to left.

(iv) π_d defined as: Flip a fair coin. If the outcome is 'Head' and the current state is in $\{s_{L-1}, s_L\}$, then swim to right; otherwise swim to left.

(v) π_e defined as: If rainy, swim to right; otherwise, swim to left. (It rains independently of the agent's action and state.)

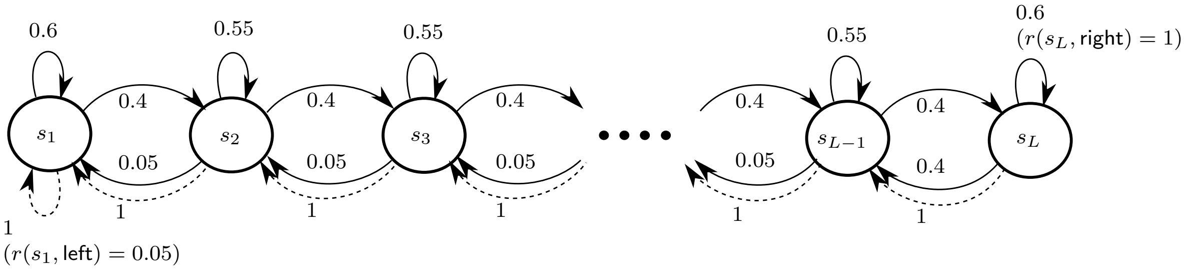

Figure 1: The L-state RiverSwim MDP (Strehl and Littman, 2008)

4 Bounds on H-step Values (15 points) [Sadegh]

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 $|U^{\pi,H}(s) - V^{\pi}(s)| \le \varepsilon$ for all s and for all π.

(ii) How do you interpret the derived bound?

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

Consider the 5-state RiverSwim MDP with $\gamma = 0.95$ (Figure 1). Consider policy π defined as:

$$\pi(s) = \begin{cases} \mathsf{right} & \text{w.p. } 0.5 \\ \mathsf{left} & \text{w.p. } 0.5 \end{cases} \qquad s = s_1, s_2, s_3,$$

$$\pi(s) = \mathsf{right} \qquad\qquad s = s_4, s_5.$$

(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 $h^i = (s_1^i, a_1^i, r_1^i, \dots, s_T^i, a_T^i, r_T^i)$ 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 $T = 200$.

(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).

6 Modifications to RiverSwim (5 points) [Sadegh]

Consider the RiverSwim MDP.

(i) Consider a beach guard monitoring the right-hand side bank (state s_L) at random. If the agent is caught being in s_L, she is fined DKK 5000; otherwise she collects an item worth of DKK 1000 in *each* visit to s_L (irrespective of the action chosen in s_L). Assume the guard visits there with probability 0.08 in an i.i.d. fashion and independently of the agent's locations and actions. Explain how the MDP could be tailored to model this task.

(ii) (Optional) Now assume the task is to swim to the right-hand side bank, pick up an item, and deliver it to the left-hand side bank (state s_1). In each successful delivery a reward of 1 is given to the agent. How would you model this task by modifying the RiverSwim MDP?

Good luck!
Christian, Yevgeny, and Sadegh

References

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47, 2002.

Alexander L Strehl and Michael L Littman. An analysis of model-based interval estimation for Markov decision processes. *Journal of Computer and System Sciences*, 74(8):1309–1331, 2008.