EL2805 Reinforcement Learning

Exam – January 2023

Division of Decision and Control Systems
School of Electrical Engineering and Computer Science
KTH Royal Institute of Technology

Exam (tentamen), January 10, 2023, kl 8.00 - 13.00

Aids. Slides of the lectures (**not exercises**), lecture notes (summary.pdf), mathematical tables.

Observe. Do not treat more than one problem on each page. Each step in your solutions must be motivated. Write a clear answer to each question. Write name and personal number on each page. Please only use one side of each sheet. Mark the total number of pages on the cover.

The exam consists in 5 problems. The distribution of points among these problems is indicated below.

Grading.

Grade A: ≥ 43	Grade B: ≥ 38
Grade C: ≥ 33	Grade D: ≥ 28
Grade E: ≥ 23	Grade Fx: ≥ 21

Responsible. Alexandre Proutiere 087906351

Results. Posted no later than January 25, 2023

Good luck!

Problem 1

(1.1) Name two algorithms to solve Bellman's equations in infinite-time horizon discounted MDPs. [1 pt]

(1.2) How can we solve Bellman's equations in the case of finite time horizon MDPs. [1 pt]

(1.3) Is Q-learning a synchronous or asynchronous stochastic approximation algorithm? Explain. [1 pt]

(1.4) Is SARSA with function approximation a stochastic approximation algorithm or a stochastic gradient algorithm? [1 pt]

(1.5) Assume that we run a ϵ-greedy policy in Q-learning algorithm, but ϵ varies over time and is equal to $1/t^2$ at step t. Does this algorithm converge to the Q function? Explain why. [2 pts]

(1.6) In actor-critic algorithms, how many parameters do we need to update? What do they correspond to? [1 pt]

(1.7) Why do we need to update the target network in DQN rarely? [1 pt]

(1.8) For discounted MDPs, the policy gradient involves the so-called discounted stationary distribution of the state. Assume that you are able to restart the system whenever you wish. Provide a way to sample from this discounted stationary distribution. [2 pts]

Problem 2

A student sequentially samples N random observations $y_k \in \{1, \ldots, M\}$ from a multinomial distribution. The random variables $y_k, k = 1, \ldots, N$ are i.i.d. and $\mathbb{P}[y_k = i] = \theta_i$ for all $i = 1, \ldots, M$. The student estimates θ by $\hat{\theta}_i = \sum_{k=1}^{N} I(y_k = i)/N$, where $I(\cdot)$ is the indicator function.

A teacher can interfere with the even observations $(y_k, k = 2, 4, 6 \ldots)$, and may decide to change each of these observations before it is seen by the student. When deciding on changing the observation y_k, the teacher is not aware of observations y_l for $l > k$, but knows the previous observations. The objective of the teacher, who knows the true θ, is to minimize the expected error made by the student at the end of the experiment. This error is characterized by $\|\hat{\theta} - \theta\|_1 = \sum_{i=1}^{M} |\hat{\theta}_i - \theta_i|$.

(2.1) Model the problem that the teacher has to solve as an MDP. [3 pts]

(2.2) Assume now that the teacher can change *any* observation, but has an initial budget of $b \leq N$ observations she can change. Model this new problem as an MDP. [3 pts]

(2.3) What do you expect the optimal policy of the teacher to be for the MDP with budget (2.2)? [1 pt]

(2.4) Consider the problem in (2.2), but assume that the teacher has access to *all* observations at the beginning. Model this new problem as an MDP. [2 pts]

(2.5) What would change in the MDP formulation of (2.2) if the teacher is adversarial and changes the observations to make the student's estimate $\hat{\theta}$ as far away as possible from θ? [1 pt]

Problem 3 – A ski rental problem

You plan to go skiing this season but you do not know in advance for how many days you will go. Based on previous seasons, you found that the probability that at given day, you stop skiing for the remaining of the season is $q \in (0,1)$. The season is assumed to have an infinite number of days. Each day you decide to ski, say the t-th day, you find a pair of skis for sale at a price p_t, that you may buy or not. The prices $p_1, p_2, \dots$ may be viewed as a sequence of random variables that are independent and identically distributed over a finite set $\mathcal{P}$. Let $f(p) = \mathbb{P}(p_t = p)$ for $p \in \mathcal{P}$. When you choose not to buy a pair of skis, you are forced to rent skis at a fixed price c per day. Your objective is to minimize your expected cost during this ski season.

(3.1) Model the problem as an MDP. *Precise the state space, action space, transition probabilities, rewards and objective.* [3 pts]

(3.2) Establish that the optimal policy is a thresholded policy, i.e., you decide to buy a pair of skis at price p if and only if $p < p_0$ (you do not need to compute the threshold p_0). [2 pts]

(3.3) Verify that according to the thresholded policy you obtain in (3.2), that if you find a pair of skis at a price $p \leq c$, then you should definitely buy them. [1 pt]

At the start of the t-th day, assuming you have already bought a pair of skis, you may be able to resell the pair of skis at a price w_t. Once you sell your skis, you are forced to rent new skis for the remaining of the day for the same fixed price c. The sequence of prices $w_1, w_2, \dots,$ by which you can sell your skis should you have any can be viewed as a sequence of random variables that are independent and identically distributed over a finite set $\mathcal{W}$; let $g(w) = \mathbb{P}(w_t = w)$ for $w \in \mathcal{W}$.

(3.4) Reformulate the problem as an MDP, and establish once again that the resulting optimal policy is a thresholded policy, i.e. you decide to buy a pair of skis at price p if and only if $p < p_1$ and you decide to sell a pair of skis at price w if and only if $w > w_1$. Furthermore verify that the thresholds p_1 and w_1 are equal. [4 pts]

Problem 4

SARSA. Consider an infinite-time horizon discounted MDP with actions $a \in \{1,2,3\}$ and states $s \in \{1,2,3,4\}$. We wish to apply the SARSA algorithm with ϵ-greedy to learn an efficient policy. We observe the following SARSA experiences (s, a, r, s', a'):

$$\{(1,2,0.3,3,2),(3,2,0.1,4,1),(4,1,-0.7,1,2),(1,2,0.3,3,3),$$
$$(3,3,-0.1,2,2),(2,2,0.3,4,3),(4,3,1,4,1)\}.$$

We initialize the (state, action) value function at 0 (i.e., $Q(s,a) = 0$ for all s, a). The discount factor is $\lambda = 0.5$ and the learning rate $\alpha = 0.5$.

(4.1) What is the (state, action) value function after the SARSA updates corresponding to the above experiences? What is the corresponding greedy policy? [3 pts]

A maze with a trap. Consider the environment in Figure **??** where the agent has to go from starting state S to the goal G in the fastest way possible. The possible actions are {UP, DOWN, RIGHT, LEFT}. The collected reward is -1 in each cell except for when the agent steps on a black tile (the trap), in which case the reward is -100 and the episode ends (the agent goes back to S).

(4.2) We consider SARSA and Q-learning both running an ϵ-greedy policy with respect to the current estimated (state, action) value function (or Q function for Q-learning). After convergence, we look at the value of the policy that the algorithms run. Is this value greater with SARSA or Q-learning? Explain. [2 pts]

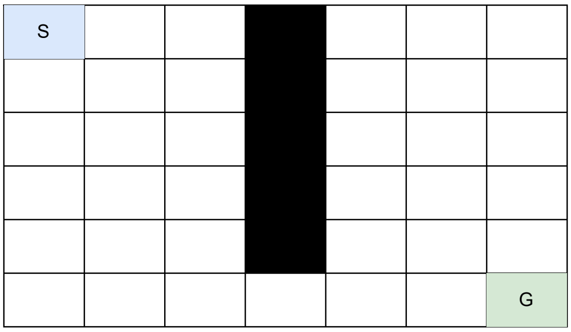

Figure 1: Environment for exercise (B.3)

(4.3) Consider again the environment in Figure **??**. Compute the *regret* of a policy π against an optimal policy $\pi^\star$, on a horizon of three steps $H = 3$. The optimal policy is the one reaching the goal in the least possible amount of steps. The policy π is defined as follows

$$\pi(a|s) = \begin{cases} 0.8 & \text{if } a = \text{DOWN} \\ 0.2 & \text{if } a = \text{RIGHT} \end{cases} \qquad \forall s \tag{1}$$

The regret of a policy π, on an horizon H is defined as $\mathbb{E}_{\pi^\star}\left[\sum_{t=1}^{H} r_t\right] - \mathbb{E}_{\pi}\left[\sum_{t=1}^{H} r_t\right]$. [2 pts.]

(4.4) TD-learning vs Monte-Carlo. One can establish that the Monte-Carlo error can be written in terms of the TD error

$$G_t - V(S_t) = \sum_{k=t}^{T-1} \lambda^{k-t} \delta_k, \tag{2}$$

where G_t is the cumulative discounted return, and $\delta_k = R_{t+1} + \lambda V(S_{t+1}) - V(S_t)$.

Now, consider the action-value version of the Monte-Carlo error

$$G_t - Q(S_t, A_t). \tag{3}$$

Show that, assuming values do not change from step to step, the action-value MC error can be expressed in the same way as (**??**) but with $\delta_k = R_{t+1} + \lambda Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)$. [3 pts]

Problem 5

The softmax policy. In this exercise we consider an MDP with continuous state space $S = \mathbb{R}^2$ and discrete action space $A = \{0, 1, 2\}$. Let $x(s,a) = \begin{bmatrix} s \\ a \end{bmatrix}$ and define a parametrized policy π_θ as follows

$$\pi_\theta(a|s) = \texttt{softmax}(\theta^\top x(s,\cdot))_a = \frac{e^{\theta^\top x(s,a)}}{\sum_{b=0}^{2} e^{\theta^\top x(s,b)}}, \quad \theta^\top = \begin{bmatrix} 1 & 2 & 0 \end{bmatrix}. \tag{4}$$

(5.1) Compute the eligibility vector $\nabla_\theta \ln \pi_\theta(a|s)$ in $s^\top = (1, 0), a = 1$. [2 pts]

(5.2) Assume the stationary distribution ρ of the state induced by the policy π_θ is a Gaussian distribution of mean $\mu = 0$ and covariance $\Sigma = I$. Compute the average reward $\mathbb{E}_{s\sim\rho, a\sim\pi_\theta(\cdot|s)}[r(s,a)]$, where $r(s,a) = \|s\|_2^2$. [2 pts]

(5.3) We assume that the MDP is stationary with terminal state (no discount). We further assume that the Q-values of π_θ are given by

$$Q^{\pi_\theta}(s,a) = \ln(1 + \theta^\top x(s,a)).$$

Let $\theta = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$. Approximate the policy gradient $\nabla_\theta V^{\pi_\theta}$ using the following trajectory of observations $((0,0),0), ((1,0),0), ((0,1),1), ((2,1),1)$ (each tuple is a state-action pair) generated according to π_θ.
Hint: approximate the expectation involved in the policy gradient using the empirical average. [2 pts]

Estimation of the value function. A classical problem in Reinforcement Learning is that of estimating the value function of a policy π. Consider an infinite time horizon MDP. Let V_{θ_t} be the estimate at step t of the discounted value of a policy π. Define

$$\hat{A}_t^{(k)} = r_t + \lambda r_{t+1} + \cdots + \lambda^{k-1} r_{t+k-1} + \lambda^k V_{\theta_t}(s_{t+k}) - V_{\theta_t}(s_t), \quad k \geq 1, \lambda \in [0,1).$$

In the previous equation $\hat{A}_t^{(k)}$ is the k-steps advantage function estimator.

(5.4) In the following we try to derive the generalized advantage estimator (GAE). Let $\delta_t = r_t + \lambda V_{\theta_t}(s_{t+1}) - V_{\theta_t}(s_t)$. Show that

$$\hat{A}_t^{(k)} = \delta_t + \lambda \delta_{t+1} + \cdots + \lambda^{k-1} \delta_{t+k-1}.$$

[2 pts]

(5.5) Consider a factor $\alpha \in [0,1)$. Prove that the discounted sum of advantage function estimators (GAE) can be written as a discounted sum of TD-errors, i.e., prove that [2 pts]

$$(1-\alpha) \sum_{n=1}^{\infty} \alpha^{n-1} \hat{A}_t^{(n)} = \sum_{n=0}^{\infty} (\alpha\lambda)^n \delta_{t+n}.$$