Published in Towards Data Science

Russo Alessio, PhD Student Follow

Jul 5, 2021 · 6 min read · Listen

An example of Reinforcement Learning exam — Rationale behind the questions (part 1)

What should you expect in the exam? My experience as a teacher

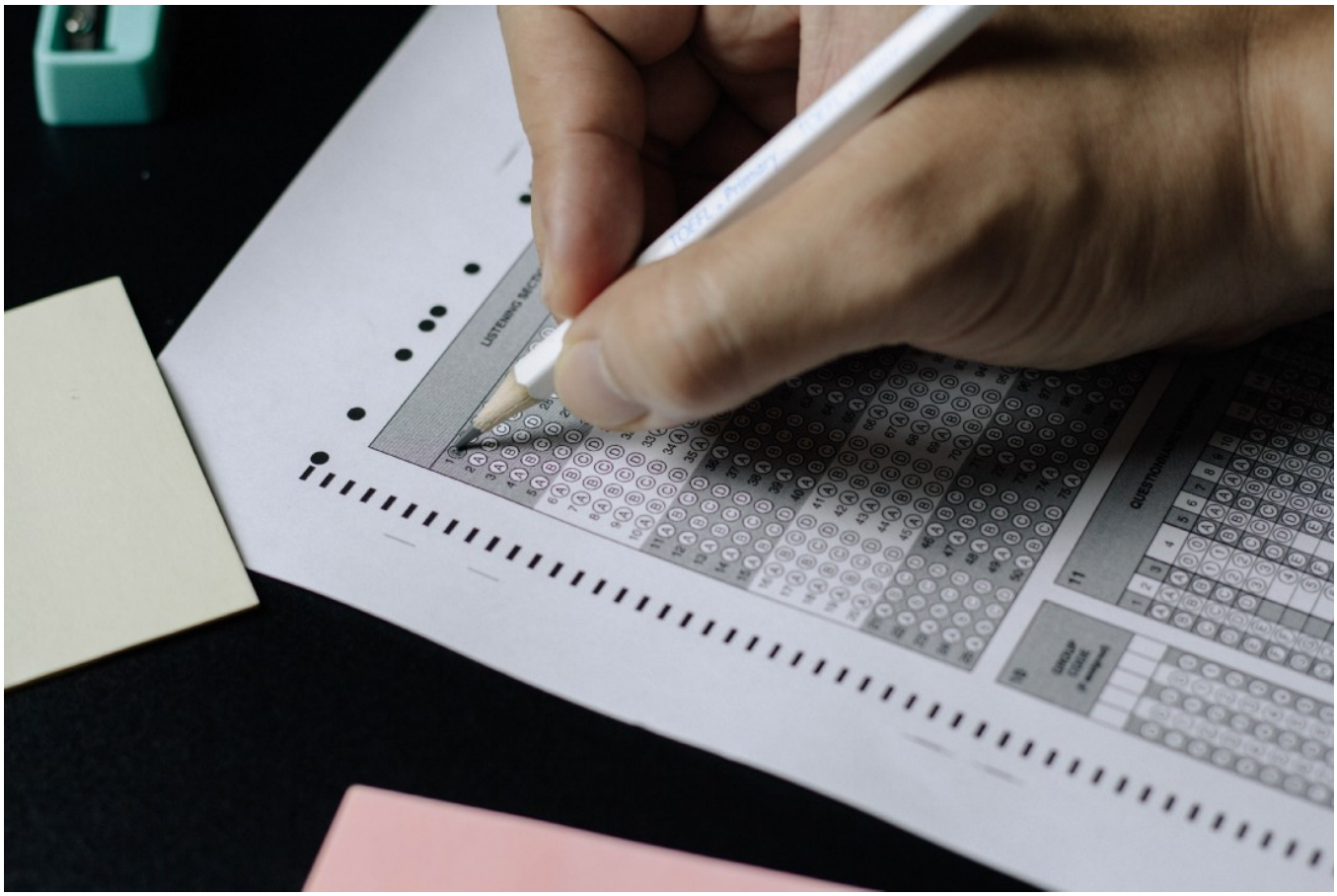

Photo by Nguyen Dang Hoang Nhu on Unsplash

- **Is there a way to prepare effectively for the exam?**
- **What are the most important topics to study in Reinforcement Learning?**
- **What are some possible questions?**

> In this series of articles, I will answer these questions and offer insight into how to tackle the test.

Due to my experience teaching at KTH Royal Institute of Technology, I encountered different problems students might be experiencing, and I will mainly focus on my own experiences. Doctoral students at KTH are required to teach as part of their activities, and this is one of my favorite things to do as a doctoral student.

As **for those who are no longer students, the article may still be informative to gain new knowledge.**

This will hopefully be of help to many students.

Introduction to the exercise: Expected SARSA and On-policy vs Off-policy learning

A friend of yours is trying to control a small robot using Reinforcement Learning techniques. He is considering a stationary MDP with terminal state with discount factor λ. The MDP has finite state and action spaces, $\mathcal{S} = \{1, 2, \ldots, S\}$ and $\mathcal{A}$. He has tried to implement SARSA with an ε-greedy policy. The policy, denoted by μ_t, used to select actions at time t is ε-greedy with respect to the current Q values. Unfortunately, his algorithm, whose pseudo-code is presented below (Alg. 1), does not work. Perhaps you can help him?

Introduction to the exercise

Last semester (January 2021), according to my supervisor, I was quite creative in designing the exam question. **In the figure above, you can find an introduction to the**

> develop reflexivity and critical thinking skills as well as an understanding of reinforcement learning.

Thus, I designed the activity to evaluate the students' skills in the following areas:

1. **Critical thinking**
2. **Understanding of basic Reinforcement Learning concepts: on-policy, off-policy learning, and convergence**
3. **Ability to adapt an algorithm to different needs**

I will briefly describe how I assessed each of these three points.

First question — Critical thinking and understanding

To my students, I usually teach them how to think through their work. After graduation, I firmly believe that having this kind of skills is essential and necessary.

Part of the first exercise of the exam was aimed at testing theoretical knowledge and capability to evaluate algorithms. Essentially, this test evaluates the critical judgment of the student in assessing a piece of work.

The image below shows the pseudo-code of a faulty SARSA algorithm.

$\varepsilon = 0.1$

1: Initialize matrices $Q(s,a)$ and $n(s,a)$ to 0 for all (s,a)
2: **for** episodes $k = 1, 2, \ldots, N$ **do**
3: $t \leftarrow 1$
4: Initialize s_1 and choose a_1 according to a uniform distribution over the actions.
5: **while** Episode k is not finished **do**
6: Take action a_t: observe reward r_t and next state s_{t+1}
7: Choose a_{t+1} from s_{t+1} using μ_t: a ε-greedy policy with respect to Q
8: Compute target value: if the episode is terminal then $y_t = 0$ otherwise

$$y_t = r_t + \max_a Q(s_{t+1}, a)$$

9: $n(s_t, a_t) \leftarrow n(s_t, a_t) + 1$
10: Update Q function: $Q(s_{t+1}, a_{t+1}) \leftarrow Q(s_t, a_t) - \alpha_{n(s_t,a_t)}(y_t - Q(s_t, a_t))$
11: $t \leftarrow t + 1$
12: **end while**
13: **end for**

This is the first question I asked the student

> 1. Spot all the mistakes in the algorithm. Motivate why those are mistakes, and correct them.

The algorithm contains a number of errors.

The question itself may appear straightforward, but it is not. **First of all, I did not provide the total number of mistakes.** It made the exercise resemble a real situation in which you have to evaluate someone else's work. **Additionally, students need to be aware of basic reinforcement learning concepts in order to fix the errors (otherwise how can you fix them?)**

This exercise is worth 3 points (out of 10), and students often mistakenly believe that each error is worth 1 point, for a total of 3 points (the overall exam is worth 50 points, with 5 exercises, each worth 10 points).

Unfortunately, this is a naive way of thinking, and students should avoid thinking in those terms. A mistake may be worth more than 1 point, or less than 1 point.

But, what are the mistakes?

> Most of the mistakes require the student to pay attention to the details, while the rest are simple mathematical errors (plus sign instead of minus sign, etc...).

$$\text{learning rate } \alpha_n = 1/\log(n+1);$$

The first mistake is about the learning rate

- **A first mistake involves the Robbins-Monro conditions, one of the cornerstones of reinforcement learning.** Stochastic approximation algorithms are used to learn from data in reinforcement learning. **The following two conditions are necessary for the convergence of stochastic approximation schemes: $\sum \alpha(t) = \infty$ and $\sum \alpha^2(t) < \infty$.**
- **In the exercise, the latter requirement is not satisfied. One can simply choose a different learning rate, such as 1/n.**

$$y_t = r_t + \max_a Q(s_{t+1}, a)$$

Three mistakes are about the target value

- **A second major mistake is the computation of the target value y(t). SARSA is an on-policy learning method.**
- **This means that the target value should be computed according to the action taken by the behavior policy** (in other words, if you took an action x, you should use the same action x to compute the target value), **and not using the max operating as in Q learning!**
- **Another mistake is the missing discount factor!**

- The only minor (really minor) mistake is that when the episode is terminal (i.e., we have reached the final state), the target value should equal the reward only (y(t)=r).

$$\text{Update } Q \text{ function: } Q(s_{t+1}, a_{t+1}) \leftarrow Q(s_t, a_t) - \alpha_{n(s_t,a_t)}(y_t - Q(s_t, a_t))$$

Two mistakes are about the update of the Q values

- Last, but not least, **the computation of the Q values also contains some minor errors.**
- **We update the values of the current state and action pairs, not the successive ones! It should be Q(s(t), a(t)) on the left hand-side, and not Q(s(t+1), a(t+1)).**
- Moreover, there is a minus sign in front of α that should be a plus.

Second and third questions: critical thinking, understanding, and adaptation

Your friend came across a variant of SARSA, which is defined through a sequence of policies $(\pi_t)_{t\geq 1}$, and consists in just changing, in Algorithm 1 **after corrections**, the way the target is computed. The target becomes:

$$y_t = r_t + \lambda \sum_a \pi_t(a|s_{t+1}) Q(s_{t+1}, a),$$

where $\pi_t(a|s)$ denotes the probability that a is selected in state s under π_t.

2. What sequence of policies $(\pi_t)_{t\geq 1}$ should you choose so that the corresponding variant of SARSA is on-policy? This variant is called Expected SARSA. [1 pt]
3. Consider an off-policy variant of SARSA corresponding to a stationary policy $\pi = \pi_t$ for all t. Under this algorithm, do the Q values converge? If so, what are the limiting Q values? Justify your answer. [1 pt]

Second and third questions of the exercise

The second and third questions in the exercise gave me a bit of freedom to experiment with the algorithm.

First, I change the way I calculate the target value. To shuffle things a bit, I introduce a new element, the policy π. **Using this method, I can assess their understanding of a basic reinforcement learning concept, off-policy vs on-policy learning.**

> Per se, the questions are not difficult. Nevertheless, the introduction of unexpected elements in an exam can have a huge mental impact on students. The majority of them may feel nervous as a result of this change, which may affect their performance.
>
> However, as long as the change does not require complex answers, I feel that the students should be able to handle it.

You can answer question 2 in one line:

- **In question (2) this policy π is like a free parameter. To answer this question, the student needs to understand the difference between on-policy and off-policy learning.**
- **In addition, the notation may be intimidating for the student. Not all students can handle mathematical notation at this level, despite being an engineering school.** In my opinion, this reflects a lack of mathematical knowledge, a result of the industrial need to have data scientists/data engineers who can tackle problems that do not require mathematical modeling.
- **The answer to question (2) is plain and simple:** π should be simply the behavior policy, i.e., the policy you use to take an action. (therefore the policy μ).

Question 3 also is a bit harder, but not much, and requires 1–2 lines of answer.


- **Therefore**, given a sufficiently exploring behavior policy, **the Q values will converge to the Q values of the policy π, and not the policy μ.**

Conclusions and next articles

Photo by Angelina Litvin on Unsplash

This is the first article of a series where I will describe some of the most common questions you can find in Reinforcement Learning tests.

In this article, I showed some simple, but tricky questions, I proposed in the last exam. Over the next few articles, I will show more exercises and discuss other reinforcement

I hope this article has inspired you in the way you solve exercises, so that it may be helpful for your upcoming exa or future studies!