CITY UNIVERSITY LONDON

EST 1894

INM707 Deep Reinforcement Learning
Coursework

Introduction

On completing this coursework, **you should be able to**:

1. Represent reinforcement learning problems in terms of transition functions and reward functions.
2. Solve reinforcement learning problems using the Q-learning algorithm.
3. Implement and evaluate advanced reinforcement learning algorithms (e.g., Deep Q-Networks).

Deliverables and submission

You must submit the following documentation:

1. A **Written Report**, which must include
   a. All the **tasks**, team and individual (up to 10 pages),
   b. A summary of your **contribution** to the team tasks (1 page),
   c. Your (team and individual) **code**.
2. Your **code** as a **zip** file. You must also share a **GitHub link** and grant access to the lecturers.
3. An **Oral Presentation** (15 minutes).

The **Written Report** must be a **.pdf** file. You must **not** use **compression**. The code must be in **Python**. The **.pdf** file (1) and the **zip** file (2) should be submitted through Moodle **before 5pm on the 24th of April**.

The **Oral Presentation** (3) must be submitted through Moodle **before the 24th of May 5pm**. **Both members of the team must participate actively in the presentations.**

- For students in the MSc in Data Science and the MSc in Computer Games Technology, the oral presentation must be submitted as a **video and your slides**.
- For the MSc in Artificial Intelligence, you must submit your slides only. You will then give a **live presentation** via Zoom (slots will be announced).

Please note that the submission system is automatic, so **late submissions are not accepted**. Do not leave final submission to the last minute.

Grading of the Coursework

- **The Written Report and the Oral Presentation are each marked out of 100 marks.**
- **To pass the module you must obtain 50% in each component – Written Report and Oral Presentation. You must resit only the component you fail.**
- **The Written Report counts for up to 70% of the Total Mark of the module; the Oral Presentation up to 30%.**

Grading of the Written Report

Solutions will be graded according to their correctness, including accuracy of the representation and numerical exactitude. Scope of the domain/task and of the solution provided will also contribute to your final grade. Grading will be structured in three levels, Basic, Advanced, and Extra, as follows:

Basic, up to 50 marks

1. 10 marks: Define an environment and the problem to be solved
2. 05 marks: Define a state transition function and the reward function
3. 05 marks: Set up the Q-learning parameters (gamma, alpha) and policy
4. 10 marks: Run the Q-learning algorithm and represent its performance
5. 10 marks: Repeat the experiment with different parameter values, and policies
6. 10 marks: Analyze the results quantitatively and qualitatively

Advanced, up to 50 marks

7. 10 marks: Implement DQN with two improvements, presented during the Lectures/Labs or from the literature (if from the literature, please consult first with Michael Garcia-Ortiz). Motivate your choice and your expectations for choosing a particular improvement. Apply it in an environment that justifies the use of Deep Reinforcement Learning
8. 10 marks: Analyse the results quantitatively and qualitatively
9. 20 marks: Apply the RL algorithm of your choice (from rllib) to one of the Atari Learning Environment. Briefly present the algorithm and justify your choice
10. 10 marks: Analyse the results quantitatively and qualitatively

Extras, up to 10 marks

11. Implementation of PPO or SAC

Given that some techniques do not scale up well from Basic environments, you are free to choose Basic and Advanced tasks that are unrelated. You can also decide to choose related Basic and Advanced tasks in order to highlight the limitations of simpler algorithms, that might work on Basic tasks and not on Advanced tasks, hence justifying the need for more complex algorithms. The Basic and Advanced tasks will be evaluated separately.

Grading of the Presentation

In the Presentation you need to explain in technical detail your work, how you approached each task and how you tackled the challenges you faced, and present and comment on the results. The Presentation will be marked according to its scope and completeness, and clarity of exposition.

Teamwork

- **You can work in teams (of two) for tasks 1-8 (team tasks); tasks 9 and 10 tasks are individual (individual tasks).**
- **You must contribute to all the team tasks given to you. Distributing labour is 'collision', a form of academic misconduct.**
- Should you prefer to work on your own entirely, this is also acceptable (although this will not give you extra marks).
- Each member of the team should submit the deliverables individually. Name the files with the code of the module and the surnames of the team **(e.g., "INM716-Alonso-Garcia.pdf")** and state again the members of the team in the first page of the written report and in the oral presentation.
- Obviously, the deliverables in the Written Report and the code that refer to the team tasks must be the same in the submissions of both members of the team. The submissions will only differ in the individual tasks and on the contribution of individuals to the team tasks.
- Teams can ***only be formed from the same MSc programme***, that is, a team cannot consist of a mix of students from Artificial Intelligence/Data Science/Computer Games Technology.
- Remember that students in the MSc in Artificial Intelligence can only team up with the same partners in two modules per term.

Notes on marking and coding

- The written report must be submitted as a non-compressed **.pdf** file. **Any other format will give you 0 marks.**
- Please structure your coursework in terms of the grading scheme, so that your marks are transparent and our feedback useful.
- Sub-tasks and thus their marks are not un-related.
- The same grading applies whether you select a problem from the literature, or if you choose your own domain and task.

- Code will not be marked. However, the code must be well-structured and commented, and runnable. Instructions must be clear.
- You must use **Python**. You are encouraged to present your code as a **Jupyter Notebook**, as this will facilitate the evaluation.
- The code must be yours. **You can use external code but please see the Plagiarism note at the end of this document.** Obviously, the more code from others you use the less marks on the corresponding tasks you will receive.

Feedback

In the labs, we will check your progress and *formative feedback* will be given to each group.

You will receive your marks and *evaluative feedback* in due time –within 3 weeks of submission.

Additional material

You can find samples of previous years' courseworks in Moodle, under ***CourseworkSampleX*.**

Extenuating Circumstances

- If you are not able to submit your coursework on time or attend the presentation for unforeseen medical reasons or personal reasons beyond your control you should contact the PG Office asap and fill an Extenuating Circumstances form. Strong evidence in the form of, for instance, medical certificates or legal statements will have to be produced.
- You must submit/present unless you hear from the EC Panel that your request has been warranted.
- Remember that lecturers cannot warrant extensions.
- ECs are individual. You must discuss with your partner in the team how your circumstances may affect the team tasks of the CW before applying for an EC.

Plagiarism

If you copy the work of others (either that of another team or of a third party), with or without their permission, you will score no marks and further disciplinary action will be taken against all members of the team. Same applies if you allow others to copy your work. You can use pre-existing code, but **you must acknowledge it and state explicitly the source** –if you don't, you will be referred to the Academic Misconduct Panel.