

SIT251 – Artificial and Computational Intelligence



Project: Investigating Reinforcement Learning

Submission Deadline: 9am Monday 1st October, 2018

Overview

Within SIT215 you have been learning about a range of problems that can be solved using techniques from artificial and computational intelligence. This study has included coverage of both models and algorithms suitable for AI and CI solutions. A particular limitation of all of the solutions that we have considered is that they are designed by hand, or rely on the problem being formulated as an optimisation task.

In this project you are going to explore an advanced technique for solving many interesting and challenging real world problems. One in which an agent *learns* a solution to a problem through interaction with the environment, and through perception of a reinforcement, or feedback signal. This field is called, naturally, *reinforcement learning* (RL). RL can also be seen as an online method for solving Markov Decision Problems – as opposed to the offline methods of policy iteration, value iteration or dynamic programming, presented in lectures (in week 9 & 10).

This project will require you to undertake self-directed study and learning of RL solution methods, building upon topics and content covered in the first 10 weeks of this course. While this might seem daunting (not being told how to solve the problem), you've been practicing this approach throughout the unit in the group-based PBL tasks, and so this is your chance to demonstrate individually what you've learned about problem solving methodology.

Learning Objectives

This project addresses ULO2 and ULO3 for this unit:

- Design and implement software artefacts to demonstrate effectiveness and efficiency of solutions for intelligent systems development
- Apply theoretical concepts and models to explain and communicate the design of intelligent systems

Specifically, these are addressed through achievement of the following task-specific learning objectives:

- Demonstrate ability to work with and extend software systems and frameworks for RL
- Describe and model RL problems using specific concepts and models
- Implement, evaluate and analyse the performance of different solutions on a range of RL problems

- Effectively communicate the process and outcomes of your research and development project

Preparatory Learning Activities

In order to complete this assessment task you will need to have first developed an understanding of a range of topics covered in this unit in weeks 1 to 10. Given the assessment deadline, this may require you to complete independent study of these topics prior to their presentation in lectures. The topics that you will need to be familiar with are:

- **Bayesian AI** (working with probabilistic representations of uncertainty)
- **State Space Search** (understanding state space representations of systems)
- **Normative Decision Theory** (definitions of rational action, utility, intertemporal utility, payoff/reward)
- **Markov Decision Problems** (representing sequential decision problems for agents acting in complex domains, reward processes and finite horizon decision problems, optimal policies)
- **Dynamic Programming** (optimal solutions to sequential decision problems under specified constraints)

Ultimately you will be able to complete this assessment task without a sound theoretical grounding in each of these areas. However, having some knowledge of these areas and understanding of how they inter-relate will make it far easier to understand learning materials on reinforcement learning, and far easier to explain and describe your investigations and outcomes in this project. Our advice is that you use this project as a basis for further study of these underlying areas, to assist in integrating the knowledge covered in this unit into a meaningful ‘whole’, which supports completing this assessment task.

Task Requirements

This project will require you to use the OpenAI Gym environment for experimenting with reinforcement learning tasks. You should start by reviewing the website for the Gym: <https://gym.openai.com/>. There are links to documentation and software downloads.

To complete this project, you need to complete the following requirements and sub-tasks.

1. Read the relevant documentation for installing AI Gym, starting with <https://gym.openai.com/docs/>.
2. Read and complete the following tutorial: <https://www.learndatasci.com/tutorials/reinforcement-q-learning-scratch-python-openai-gym/>, ensuring that you can reproduce all steps discussed.
3. Write a brief report (2-3 pages at most) on the Taxi problem, including a mathematical description of the reinforcement learning problem and the Q-learning algorithm for its solution. To do this, you may want to refer to a good textbook on reinforcement learning. A good starting point is the “bible” of RL: “*Reinforcement Learning: An Introduction*”, by Sutton & Barto. You can find this book online as a free PDF download. There’s even a 2nd edition draft completed just this year. In your report you should contrast the quality of solution of a random policy versus the “optimal” policy obtained by Q-learning.

4. Complete the following tutorial to explore the Cart-Pole environment in the Gym:
<http://kvfrans.com/simple-algorithms-for-solving-cartpole/>. In this case, implement a random policy and Q-learning. It's not essential that you attempt the policy gradient method, but you might like to try it.
5. Extend your report to cover briefly the Cart-Pole problem, highlighting any differences with the Taxi problem. Compare performance of Q-learning on both of these problems, presenting evidence (such as graphs) to support your evaluation.

If you've gotten to this point and created a good report that details what you've learned, you've met the minimum requirements for this assessment task. Assuming a reasonable quality of report and evidence, you can expect to earn a credit grade. Continue on to achieve a higher grade.

6. **[Distinction]** Select another environment from the OpenAI Gym, and implement Q-learning for this environment. Extend your report to describe this new environment, including a mathematical model. Evaluate performance of Q-learning on this model, and identify any significant outcomes or limitations of this approach on this new problem, compared to previous problems. Attempt to explain any difference or limitations.
7. **[High Distinction]** Implement Temporal Difference learning on the new environment you completed for step 6, as well as one of the Taxi problem, or the Cart-Pole problem. Contrast the performance of TD learning and Q-learning in your report, providing evidence such as graphs and performance data.

Submission Components & Due Dates

This is an individual assessment task and as such, each student will complete their own project and submission components. To be eligible for assessment in this task you must submit the following artefacts to the relevant submission folder on the Unit Site no later than the given deadline:

- 1) The report detailing your models, experiments and outcomes of your reinforcement learning problems and solutions. Your report should provide adequate information to evidence your learning against the objectives stated above, and in line with the assessment rubric provided.
- 2) All code developed or used in this project. Your code must include appropriate documentation (internal comments are sufficient) that explains what the code does. You should also provide instructions on how to execute the code (for example, in a README file).

You may assume that the assessment team has access to OpenAI Gym and can execute your code. If you rely on any third party libraries or applications that are required to run your solution, you need to provide those, or make them accessible to the assessor (e.g., by providing a link to a download site, and instructions on how to install and use the library in your solution).

Assignment Marking

This assignment will be marked on the following scale

| Level | Grade |
|-----------------------------------|--------|
| Does Not Meet Minimum Standards | N |
| Meets Minimum Standards | P or C |
| Exceeds Minimum Standards | D |
| Greatly Exceeds Minimum Standards | HD |

A numeric mark will be assigned based on the assessor's determination as to where within the relevant grade category the standard or work sits. A rubric will be provided on the Unit Site, under the Resources>Assessment folder, to indicate the criteria upon which your submission components will be assessed and the standards that will be applied for these criteria. Please contact the teaching team if you have any concerns or questions regarding how you will be assessed.

Penalties

In accordance with Faculty assessment policies, late submissions to the submission folder will incur a penalty of 5% of the total available marks per day, up to five days total, after which the score for this part of the task is 0. Such penalties will be deducted from the awarded numeric mark to determine the final grade for this task.

Submissions will not be accepted or marked more than five days after the final submission deadline, except in cases where an extension has been approved **prior to the deadline**.

Getting Help and Support

Students are encouraged to support each other to discuss the tasks, as well as to assist in overcoming problems in understanding the concepts, models and algorithms relevant to RL problems and solutions. Getting feedback from peers will certainly improve your understanding in this project, and help others to build their understanding. Note however that as this is an individual assessment task, and all development work and report writing must principally be the work of the student being assessed. Where you are asked to replicate the work of others (e.g., completing the tutorials and reproducing the code and results of others), ensure that you accurately and appropriately reference the source work. Academic penalties for collusion and plagiarism are severe and students are urged to seek guidance and advice from the teaching team if they have any concerns about how to complete this task appropriately.

Programming & Software Help

The School of IT runs a Learning Help Hub, which is accessible Monday to Friday, both on campus and in the cloud. The Help Hub staff can assist with issues surrounding programming, and installation of software (such as setting up Python), but they are NOT going to help you to complete the assignment. They cannot tell you how to do this project, nor tell you if you're doing it correctly. They are not teaching staff. You should contact the Help Hub in the first instance to deal with any programming or software related issues.

The Help Hub runs as follows:

Monday: 2pm-6pm – Burwood (T1.05) & Bb Collaborate (all students)
Tuesday: 2pm-6pm – Burwood (T1.05) & Bb Collaborate (all students)
Tuesday: 6pm-8pm – Bb Collaborate (Cloud students only)
Wednesday: 2pm-6pm – Waurin Ponds (Ka5.213) & Bb Collaborate (all students)
Thursday: 2pm-6pm – Burwood (T1.05) & Bb Collaborate (all students)
Thursday: 6pm-8pm – Bb Collaborate (Cloud students only)
Friday: 2pm-6pm – Burwood (T1.05) & Bb Collaborate (all students)

Mathematics & Algorithm Help

The teaching team are here to support you in this task, in particular to assist you to develop an understanding of reinforcement learning models and solution algorithms. We are also here to help you learn the underlying knowledge, as described in the Preparatory Learning Activities section. If you are having trouble understanding this material, or this task, please make contact with us. The best way to do this is by asking questions in the Project Discussion forum on the Unit Site. Answers to your question will also help other students, who are undoubtedly having the same kinds of problems as you.

Beyond this you may seek assistance from the teaching team during practical classes, or Bb Collaborate sessions. Additional Bb Collaborate sessions dedicated to this project will be run in weeks 9, 10 and 11. Details of these will be provided on the Unit Site.

Report Writing Help

While the teaching team are also happy to provide advice and guidance on writing your report, the University also provides support services for students. In particular, the Writing Mentors team offer great assistance for students completing written assessment tasks – especially report writing. Visit <http://www.deakin.edu.au/students/studying/study-support/writing-mentors> for more information.

Feedback

Students will receive verbal, written or recorded audio feedback on their project submission as part of their assessment. Due to the timing of assessment and scheduling of exams by DSA, it cannot be guaranteed that this feedback will be provided before the unit exam. Where a student requires specific feedback prior to the exam, they should contact the Unit Chair, allowing sufficient time prior to the exam for this feedback to be provided.

Students are actively encouraged to seek formative feedback from peers and teaching staff, on their work completed before the submission deadline, to ensure they are on track with this task. Feedback may be obtained during weekly scheduled practical classes upon request. Talk to us and we'll support you!