COMP0037 2021/22 Coursework 2

COMP0037 Teaching Team

24th March 2022

Overview

- Coursework 02 Release Data: Thursday, 24th March 2022; Latest Revision 25th March 2022
- **Assignment Due Date: 19th April 2022 (16:00 UK time)**
- Weighting: 60% of the module total
- Final Submission Format: Each group submits *three* things:
    1. A zip file (which contains the source code implemented to tackle this coursework). The name of the zip file must be of the form `COMP0037_CW2_GROUP_g.zip`, where `g` is the letter code of your group.
    2. A report in PDF format. It will be named `COMP0037_CW2_GROUP_g.pdf`.
    3. A video which describes your answers. You may use a variety of video formats, but the video must be playable by VLC. It will be named `COMP0037_CW2_GROUP_g.[ext]`.

The total marks for each question are written in the form [X marks]. Approximately 67% of the total coursework marks are from your analysis and 33% from your coding of the algorithms. Therefore, please take time and care with your writing and analysis of solutions.

For the submitted code, you must use the notation variable names and code provided in the lectures. If you do not use these, we will consider the work a potential plagiarism case and investigate further.

You will need to implement additional routines to support the functionality required in these questions. The code to implement this is available on Moodle. It is based upon but extends and refactors the code provided in the labs. The general areas requiring adjustment will be shown using comments.

We expect you to take care in writing your report. For example, figures and graphs need to have captions and be numbered and labelled. Equations captured by a screenshot, from the slides or elsewhere, and pasted into the document will not be accepted. When we ask you to write some code, provide an explanation in your report of what you wrote.

The final mark will be computed by summing all the marks awarded on individual questions together and dividing by the mark total.

Neither the video nor the code will be independently marked, but both must be provided before the submission deadline. Both will be checked.

Use Case

The use case extends the scenario you encountered in Coursework 01. A robot has been tasked with automatically cleaning various terminals in an airport. Airports are generally very busy places and require constant cleaning.

In Coursework 01 you used policy and value iteration to develop policies for the trajectory for the robot. In this coursework you will experiment with Temporal Difference Learning (TDL) algorithms.

Given both the length of coursework 01 and the delay in releasing this coursework, the tasking for this coursework is much shorter.

Questions

Two methods for model-free evaluation and control have been proposed: Monte Carlo methods, and Temporal Difference (TD) Learning.

1. a. What are Monte Carlo approaches? How are they potentially better than the policy and value iteration algorithms described so far?

[10 marks]

b. Describe how Monte Carlo methods can be used to evaluate the performance of a policy $(v_\pi(s))$ in an episodic system. What is the difference between first visit and every visit algorithms? Your description should include an explanation of the equations and algorithms used.

[8 marks]

c. How can you use the Monte Carlo prediction of the policy to estimate the policy itself? Why is it not used in practice?

[4 marks]

d. Describe another approach for using the MC approach to estimate the policy. Your description should include common techniques to address issues if the algorithms become stuck.

[10 marks]

e. What are the main disadvantages of MC algorithms?

[2 marks]

[Total 34 marks]

2. a. Describe what Temporal Difference (TD) methods are. What are their potential advantages over both Monte Carlo methods and policy and value iteration?

[8 marks]

b. Describe the TD(0) algorithm for policy prediction. Why, in general, does this compute a different solution from the Monte Carlo estimate?

[10 marks]

c. Define and explain the difference between *on-policy* and *off-policy* algorithms. What are the advantages and disadvantages of each?

[6 marks]

d. Describe the Sarsa and Q-learning algorithms and identify if they are on or off policy.

[12 marks]

[Total 36 marks]

3. The TD learning algorithms will now be applied to learn policies for the robot to move in some small environments in the airport.

   a. Modify the `SarsaLearner` class to implement Sarsa. Explain your code. Run the scripts `q1_ab_pi.py` and `q1_a_sarsa.py` to compare the performance of policy iteration and Sarsa on a 3×3 problem. Present what value Sarsa converges to. Discuss how long it takes to achieve this, and whether it converges to the same solution as policy iteration.

   Hint: The algorithm might take a very long time to converge.

   [12 marks]

   b. Modify the `QLearner` class to implement the Q-learning algorithm. Explain your code. Run the scripts `q1_ab_pi.py` and `q1_b_ql.py` to compare the performance of policy iteration and Q learning on a 3×3 problem. Present what value Q learning converges to. Discuss how long it takes to achieve this, and whether it converges to the same solution as policy iteration.

   Hint: The algorithm might take a very long time to converge.

   [12 marks]

   One suggested explanation for the behaviour of the Sarsa and Q-Learning algorithms is that they quickly estimate a policy which will achieve the terminal state, but the computed policy is not very optimal. However, the algorithm will continue to iterate until an optimal solution is obtained.

   c. Modify `SarsaLearner` and `QLearner` to store the returns from each episode and explain your code.

   Plot graphs of the average returns from each algorithm over the total number of episodes and discuss any trends you observe. Do they support or refute the proposed explanation?

   [10 marks]

d. Several approaches for improving the performance of Q-learning algorithms have been proposed. Choose two techniques you believe to be most relevant and justify your choice. Describe, but do not implement, the two algorithms you have chosen.
[12 marks]

[Total 46 marks]

[Coursework Total 116 marks]

Video Submission

Please note that the video is not separately marked or graded. However, you *must* submit a video if your coursework mark is to count. The reason for providing a video is to help reduce the prevalence of plagiarism. We are not looking to mark / penalise presentation skills.

In your video, you will discuss your solution for each part you attempted for the coursework above. Each video will feature all the members of the team. Each member of the team will be expected to speak for at least a minute, and the overall video should run between 3 and 10 minutes. (Please note that there is no assumption that longer is better. Rather, producing a very compact short video can take a very long amount of time, which is not the intent of this task.)

The video should include the following:

- An introduction to the team. Each team member must introduce themselves on camera.
- For each part of the coursework attempted, explain your solution. You do not need to produced a polished presentation. It is sufficient to highlight parts of the report or code using a mouse cursor and to discuss the presented text. Simply reading out the text of the solution is not sufficient. Instead, you should discuss your solution and identify things such as where the challenges lie.
- For each part of the coursework not attempted, provide a brief explanation of why this was the case.

Recordings carried out using Zoom, Teams, etc. are more than sufficient. One idea might be to arrange a group call in which each person in turn shares their screen and explains one of the questions answered. If you can show videos of algorithms running in real-time this is even more effective, but it is not mandatory.

There are a variety of ways to encode videos including the standard formats used in Zoom or Teams. As noted, the requirement is that the video must be playable by VLC 3.0.5 or later (https://www.videolan.org/vlc/releases/3.0.5.html). (VLC is supported on Linux, Mac and Windows. It is typically plays a much wider range of media than, say, the Windows Media Player or the QuickTime Player.) Note that moodle has a maximum file upload size of 160MB.

Getting Help

You are encouraged to use the assignment discussion forum to help when you get stuck. Please check the forum regularly and try and answer each other's questions. Notifications should now be set up, and we will be checking the forum as often as we can and will be answering any questions that have remained unanswered. Please note that if you have questions about coursework, please post them to the forum directly rather than emailing me. The reason is that all students will be able to see the reply.