Homework 2 Template

Use this template to record your answers for Homework 2. Add your answers using LaTeXand then save your document as a PDF to upload to Gradescope. You are required to use this template to submit your answers. **You should not alter this template in any way** other than to insert your solutions. You must submit all 6 pages of this template to Gradescope. Do not remove the instructions page(s). Altering this template or including your solutions outside of the provided boxes can result in your assignment being graded incorrectly. You may lose points if you do not follow these instructions.

Instructions to upload code have been provided in the handout.

Instructions for Specific Problem Types

On this homework, you must fill in the blank for each problem; please make sure your final answer is fully included in the given space. **Do not change the size of the box provided.** For short answer questions you should **not** include your work in your solution. Only provide an explanation or proof if specifically asked. Otherwise, your assignment may not be graded correctly, and points may be deducted from your assignment.

Fill in the blank: What is the course number?

10-703

Problem 0: Collaborators

Enter your team's names and Andrew IDs in the boxes below. If you do not do this, you may lose points on your assignment.

Name 1:	Boxiang Fu	Andrew ID 1:	boxiangf
Name 2:		Andrew ID 2:	
Name 3:		Andrew ID 3:	

Problem 1: DQN (15 pts)

1.1 DQN plot (15 pts)

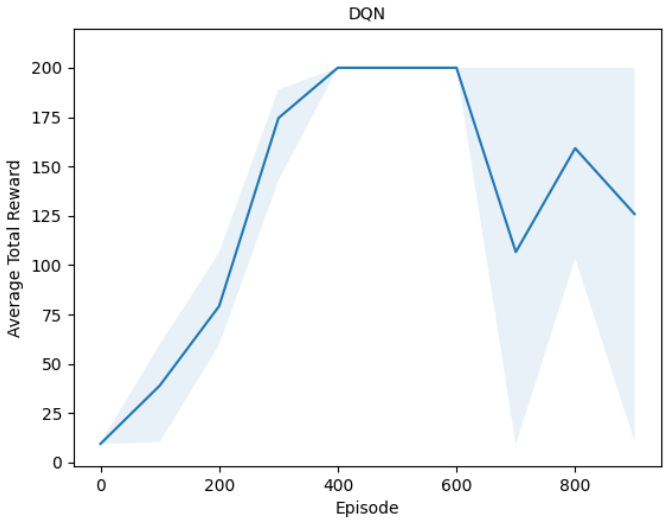

Figure 1: DQN with default hyper-parameters

Problem 2: Double DQN (21 pts)

2.1: Double DQN plot (10 pts)

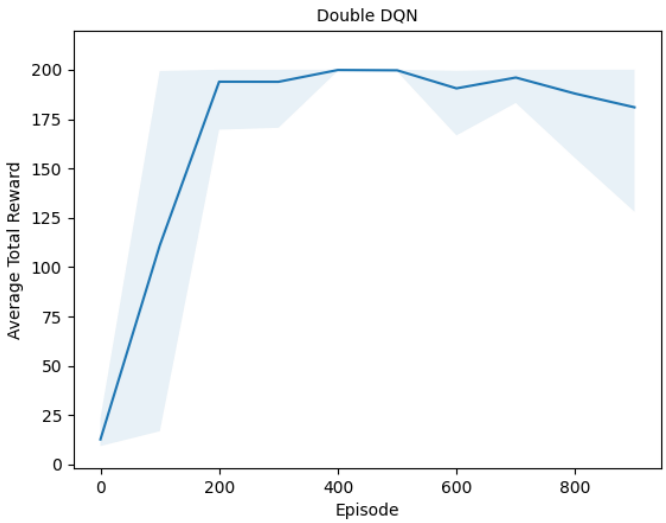

Figure 2: Double DQN with default hyper-parameters

2.2 DQN vs. Policy Gradient Algorithms (5 pts)

In general, Double DQN outperforms the REINFORCE/A2C algorithms by obtaining high rewards (200) using less episodes (approx. 100 compared to on average 200+ for REINFORCE/A2C). Double DQN achieves this since it is an off-policy algorithm, and can reuse old environment interactions stored in its replay buffer, increasing its sampling efficiency. In contrast, on-policy algorithms from Homework 1 cannot use the same trajectory once it is consumed. Thus, it requires more episodes to interact with the environment using policy roll-outs. Additionally, the environment (CartPole-v1) has a relatively small action/state space with relatively simple reward mechanisms (i.e. keep the pole up). Such problems are relatively simple to solve by choosing actions using only state information (e.g. good actions can be determined using Newtonian mechanics based on the current state). It doesn't require stochastic exploration used in REINFORCE/A2C, which injects additional variance to the actions chosen and slows down learning the optimal policy.
Reference: https://medium.datadriveninvestor.com/which-reinforcement-learning-rl-algorithm-to-use-where-when-and-in-what-scenario-e3e7617fb0b1

2.3 Pros and Cons of Policy gradient methods (6 pts)

A setting where A2C would be preferable to DQN is robot arm manipulation using continuous torque/force outputs. Such a setting is a continuous action space in a high-dimensional state space. Using DQN, we would need to discretize the action space. This is not optimal since small action differences can have differing results for manipulation. An A2C method would be much more suited as we can directly optimize the policy parameters using on-policy rollouts (i.e. controlling the robot arm using teleoperation).

A setting where DQN would be preferable to A2C is playing Atari using discrete control inputs (i.e. left/right/up/down). Such a setting is a discrete action space in a low-dimensional state space. Using DQN is more sample efficient as DQN is off-policy and we can use replay buffers to train on old environment interactions. An A2C approach would work, but there will need to be more rollouts since it is on-policy. Thus, it would take more epochs to converge.

Feedback

Feedback: You can help the course staff improve the course for future semesters by providing feedback. You will receive a point of you provide actionable feedback. What was the most confusing part of this homework, and what would have made it less confusing?

The most confusing part of this homework is that Algorithm 1 in Problem 1 did not mention a target network, but Problem 2 the target network was mentioned for the standard DQN algorithm. From the problem description, I thought the y_j values are calculated from the target network. It was not until I read Question @90 on Piazza that I understood that we should be using only one network. It would be less confusing if Algorithm 1 explicitly mentioned that there is no target network and how y_j values are based on a single network.

Collaboration: Detail the work division amongst your group below.

I did not join a group for this assignment as I had a weekend to spare to work on the assignment.

Time Spent: How many hours did you spend working on this assignment? Your answer will not affect your grade. Please average your answer over all the members of your team.

Alone	12
With teammates	0
With other classmates	0
At office hours	0