Exercise 6

Jari Mattila - 35260T
ELEC-E8125 - Reinforcement Learning

November 1, 2021

Task 1

Revisit the policy gradient solution for the continuous Cartpole from Exercise 5 with learned sigma and implement the actor-critic algorithm. In the initial setup, perform TD(0) updates at the end of each episode.

The training performance plots for each of the tasks (Tasks 1, 2 and 3).

Source files: cartpole.py, agent_task1a.py, agent_task1b.py, agent_task1c.py

Question 1

What is the relationship between actor-critic and REINFORCE with baseline?

Question 2

How can the value of advantage be intuitively interpreted?

Task 2

Update the actor-critic code to perform TD(0) updates every 50 timesteps, instead of updating your network at the end of each episode. Make sure to handle the end of each episode correctly (as in the previous exercises, the value of the terminal state is 0).

The training performance plots for each of the tasks (Tasks 1, 2 and 3).

Source files: cartpole.py, agent_task1a.py, agent_task1b.py, agent_task1c.py

Task 3

Update your code to use parallel data collection. Start up with the parallel_cartpole.py script. This code makes use of the parallel wrapper for the environment, which can be found in parallel_env.py. This code instantiates processes worker processes, with envs_per_process threads each. After adapting your code, run it with at least 20 parallel environments and report the results. You can use Aalto computational resources (Maari, Paniikki) if your computer runs sluggish.

The training performance plots for each of the tasks (Tasks 1, 2 and 3).

Source files: cartpole.py, agent_task1a.py, agent_task1b.py, agent_task1c.py

Question 3

How is parallel data collection different from the parallelism in multiple_cartpoles.py script we've seen in Exercises 1 and 5? Can it replace multiple runs of the training algorithm for comparing RL algorithms? **Explain your answer.**

Question 4

Figure 1 shows the training performance for all three actor-critic variants and the REINFORCE algorithm from the last lecture. In terms of initial performance, REINFORCE seems to completely outperform all tested A2C flavours on Cartpole, despite being a simpler algorithm. **Why is it so? Explain your answer.**

Question 5.1

How do actor-critic methods compare to REINFORCE in terms of bias and variance of the policy gradient estimation? **Explain your answer.**

Question 5.2

How could the bias-variance tradeoff in actor-critic be controlled?

Question 6

What are the advantages of policy gradient and actor-critic methods compared to action-value methods such as Q-learning? **Explain your answer.**

1 Question 1

If you add a figure, you can refer to it using Figure. 1.

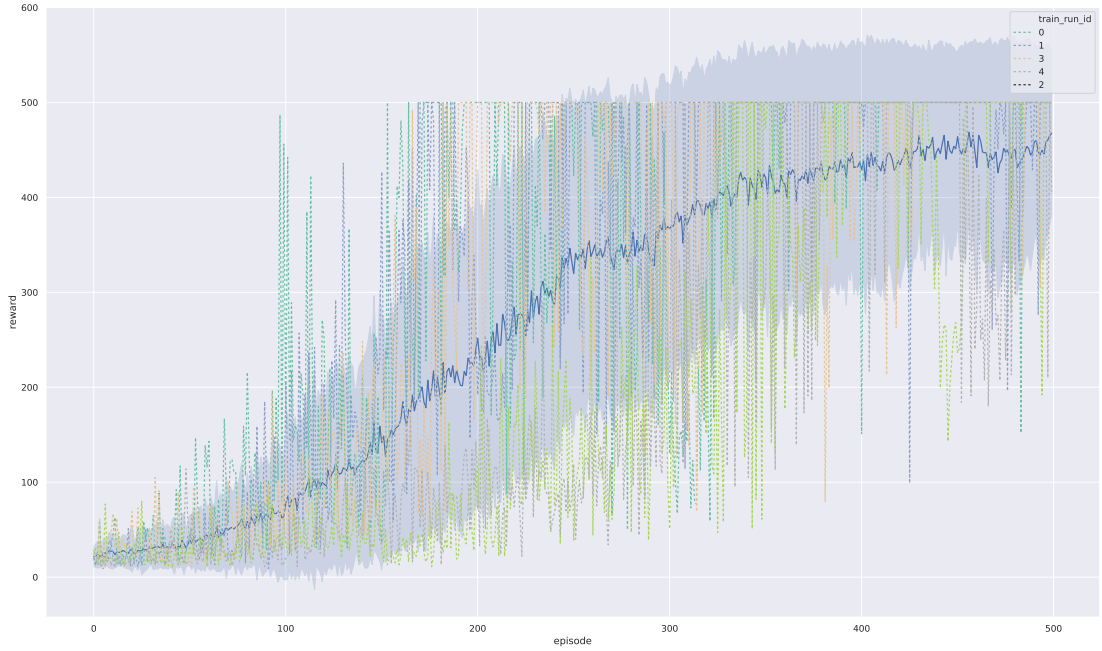

Figure 1: This is a sample figure.