Exercise 1

Jari Mattila - 35260T
ELEC-E8125 - Reinforcement Learning

October 10, 2021

Task 1 - 5 points

The training of the Cartpole with 200 timesteps per episode and 500 as the total number of averaged attempts is visualized in Figure 1.

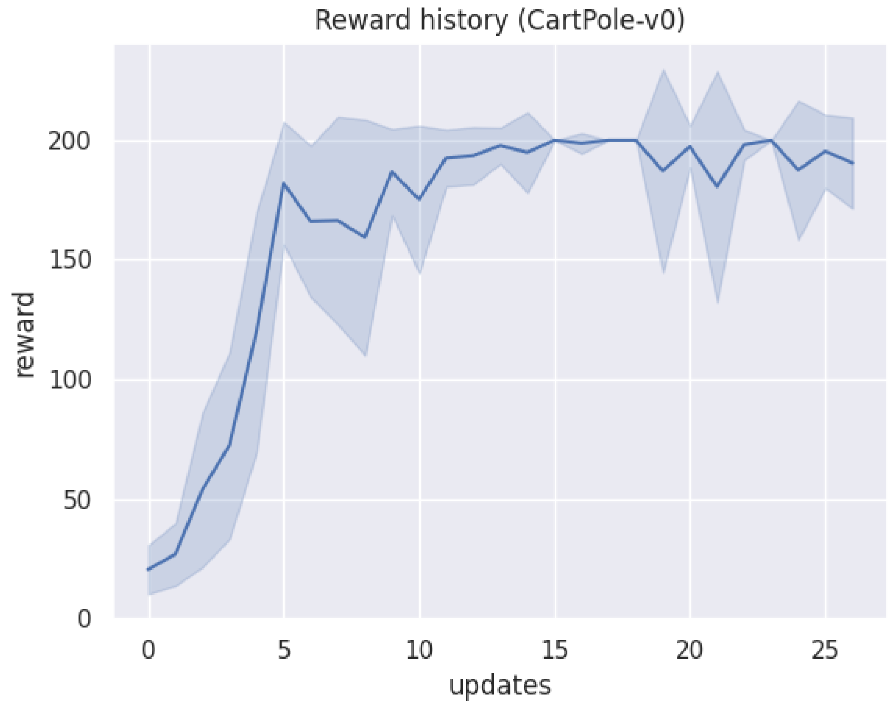

Figure 1: Cartpole training with 200 timesteps per episode.

Question 1.1 - 15 points

Can the same model, trained to balance for 200 timesteps, also balance the pole for 500 timesteps? Briefly justify your answer.

Running the balancing test with 500 timesteps gives the following result:

```
# of timesteps 500: Average test reward: 494.644 episode length: 494.644
```

Yes, the same model that is trained to balance for 200 timesteps can be used to balance the pole for 500 timesteps. You can see from Figure 1 that 200 timesteps (shown as the number of updates in the figure) for training is more than enough to learn the behavior.

Task 2 - 10 points

The average test rewards over three trials are as follows:

```
Average test reward: 497.818 episode length: 497.818
Average test reward: 489.306 episode length: 489.306
Average test reward: 497.082 episode length: 497.082
```

Question 1.2 - 15 points

Are the behavior and performance of the trained model the same every time? Why/why not? Analyze the causes briefly.

The behavior and performance of the trained models are not exactly the same every time because the training procedures are stochastic random processes. This is also illustrated by the average test rewards shown above. However, the overall difference between the average test results is rather small because the averaging is over 500 independent trials.

Question 2 - 10 points

Figure 2 shows the mean and standard deviation throughout 100 independent training procedures. You can notice that there is a large variance between the runs of the script.

Why is this the case? What are the implications of this stochasticity, when it comes to comparing reinforcement learning algorithms to each other? Please explain.

As shown in Figure 2, the performance difference between the individual training procedures is rather large that is due to the stochasticity of the training processes. This is also indicated by the large standard deviation (variance) between the individual processes. Comparing different RL algorithms to each other is possible using the averaged results. If the average results are computed over sufficiently many independent trials then the comparison of different algorithms should be valid. In Figure 2, the average of the reward curves is shown as an emphasized dark curve that would be easily comparable to other variations of the algorithm.

Task 3 - 20 points

1. Reach the goal point located in x=[1.0,1.0]

The code snippet below was used to reach the goal point in x=[1.0,1.0]. The idea is simply to give a reward =0 if the goal point is reached and otherwise =-1. The reward function of the Reacher training is shown in Figure 2.

```python
def get_reward(self, prev_state, action, next_state):
        # Cartesian position of the end-effector
        cartesian_pos = self.get_cartesian_pos(next_state)

        # Distance between the position of the end-effector and the goal point [1.0,1.0]
        terminal_distance = np.sqrt(np.sum((cartesian_pos - self.goal)**2))

        is_terminal = terminal_distance < self.termination_threshold

        # Reward is 0 if the goal is achieved - otherwise -1
        reward = -1
        if is_terminal:
            reward = 0

        return reward
```

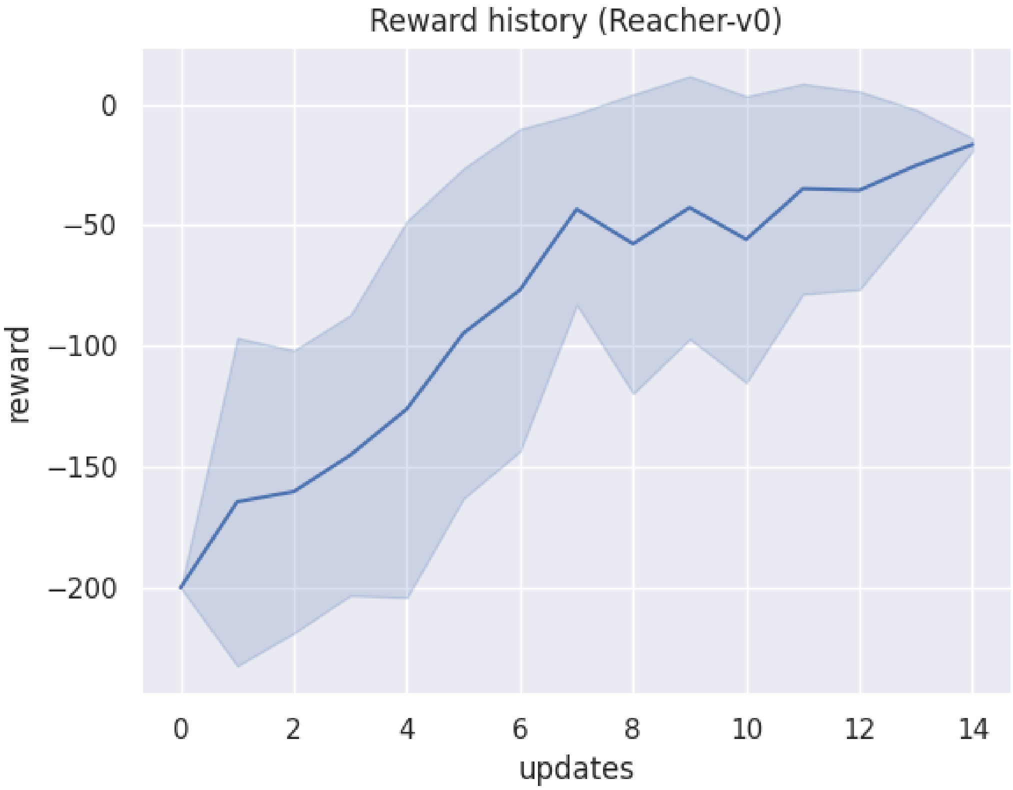

Figure 2: Reacher training with 200 timesteps per episode.

Task 4 - 10 points

Now, let us visualize the reward function for the first behavior (reaching the goal [1,1]). Plot the values of the first reward function from Task 3 and the learned best action as a function of the state (the joint positions).

The reward function and the learned best actions are illustrated in Figure 3.

Question 3

Analyze the plots in Task 4.

Question 3.1 - 5 points

Where are the highest and lowest reward achieved?

The highest rewards (=0.0) are achieved in the light spots of Figure 3 (see the Reward plot on the left). As an example, one such spot area is $j_1 \approx 3.14$ and $j_2 \approx [-1.88, -1.26]$. The lowest rewards (=-1.0) in Figure 3 are achieved in the black area.

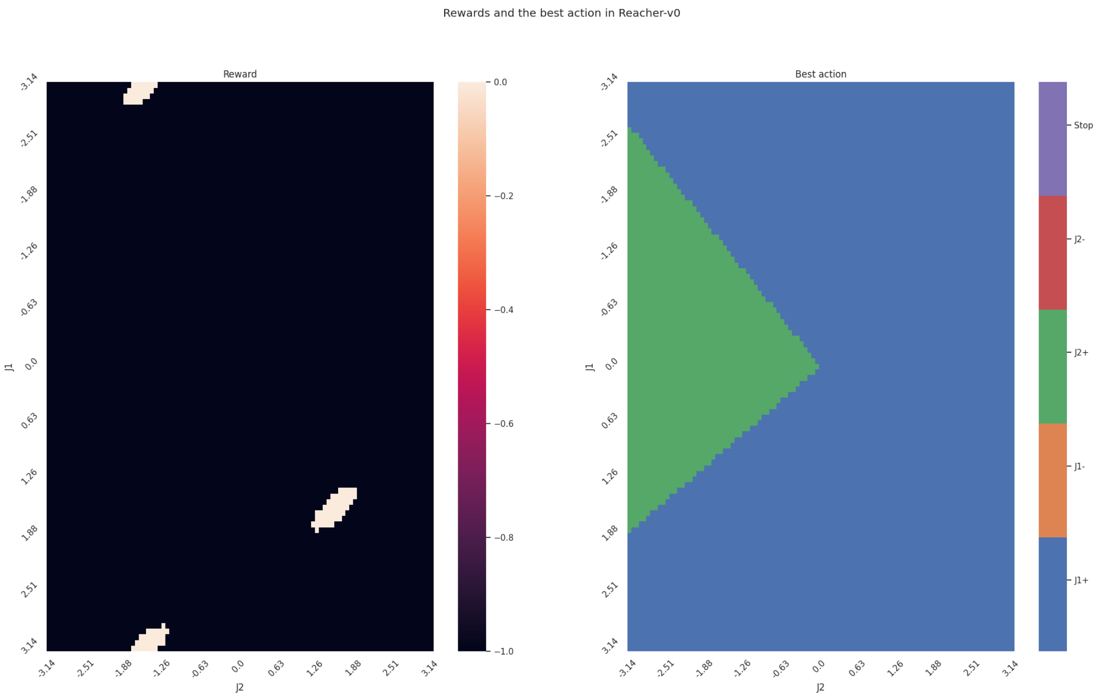

Figure 3: The learned best action as a function of the state (the joint positions).

Question 3.2 - 10 points

Did the policy learn to reach the goal from every possible state (manipulator configuration)? Why/why not?

No, the policy didn't learn to reach the goal from every possible state (manipulator configuration). It is quite obvious that only some manipulator configurations are possible for reaching the goal point [1,1], see e.g. the light spots of Figure 3 (the Reward plot on the left).