Deep Reinforcement Learning assignment

Lorenzo Zanolin

Results

RUN	LAYERS	EPISODES	BATCH SIZE	GAMMA	LEARNING RATE	DURATION after 100 ep (AVG)
1	3	600	128	0.99	10^{-4}	159.5
2	3	600	256	0.99	10^{-4}	403.8
3	3	600	256	0.99	10^{-3}	400.3
4	4	600	256	0.99	10^{-4}	338.7

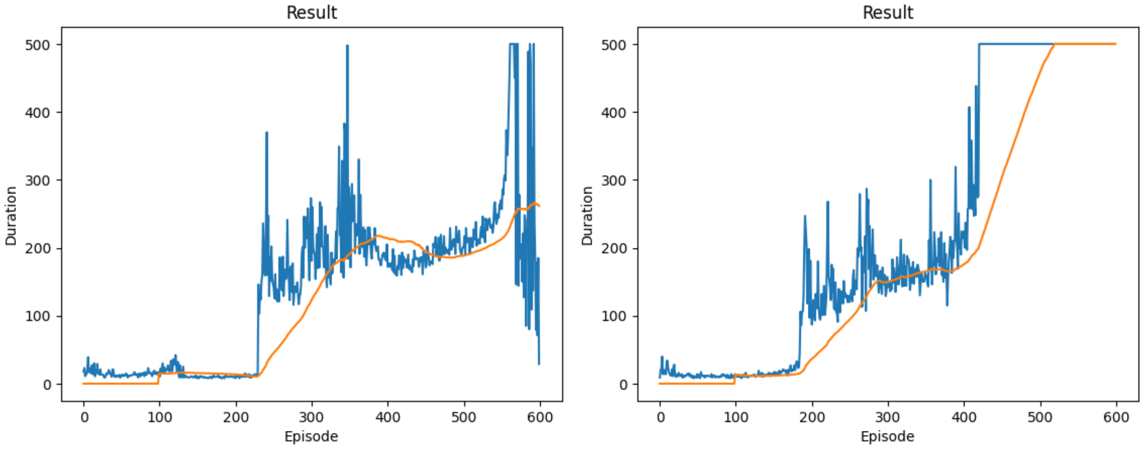

Figure 1: (a) First run. (b) Second run.

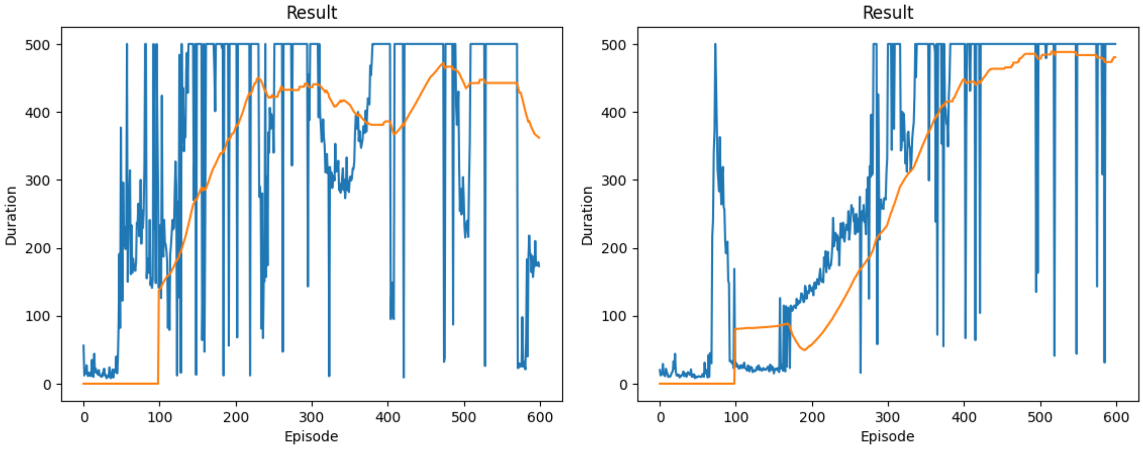

Figure 2: (a) Third run. (b) Fourth run.

Comments

The aim of the homework was to experiment the training of a Deep Q Learning (DQN) agent on the CartPole-v1 task from Gymnasium; the request was to also to keep track of how long the pole can be balanced after 100 episodes, thus we decided to calculate the *average* duration considering the episodes within $[100, 600]$.

The architecture used were the following:

- 3 layers: the standard one[1], with two ReLU's.
- 4 layers: added one intermediate layer, this time using 255 neurons per layer.

As we can see from the table above, the results were pretty much similar using a batch size of 256 and a $3-layers$ architecture; while adding a new layer seems to perform worse.To verify whether the collected data was representative and not randomic, other runs were done obtaining similar values; this means that for this specific task 3 layer architecture performs better.

For completeness, other tests were done changing the number of episodes and gamma but the results were not so interesting (seems pseudo-randomic), thus were not included in the above table.

References

[1] en. URL: `https://pytorch.org/tutorials/intermediate/reinforcement_q_learning.html`.