Deep Learning Lab: Exercise 3 (RL)

Neeratyoy Mallik (4774378)

June 6, 2019

1 Imitation Learning

1.1 Hyperparameters

- Architecture[1]
  - # of convolution layers: 2
  - # of fully connected layers: 2
- Batch size: *64*
- Learning Rate: *1e-4*
- Optimizer: *Adam*

1.2 Training

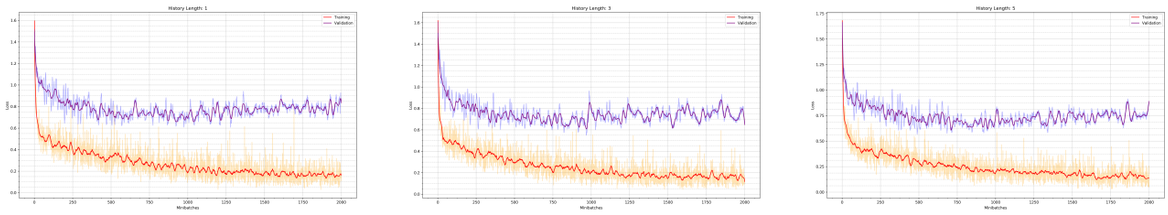

Figure 1: Loss over minibatches

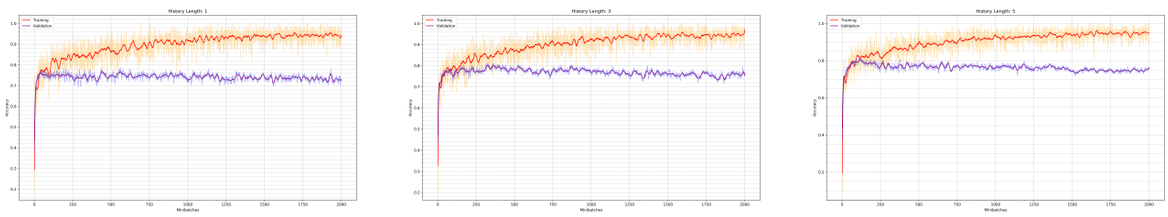

Figure 2: Accuracy over minibatches

1.3 Testing

Metric	Expert	History:1	History:3	History:5
Mean	847.51	442.37	461.56	583.79
Std. Dev.	19.49	89.09	84.16	101.27

Over 15 episodes with max time-steps of 1000[2]

Metric	Expert	History:1	History:3	History:5
Mean	847.51	734.85	674.32	844.04
Std. Dev.	19.49	197.76	145.31	40.92

Over 15 episodes with max time-steps of 2000[2]

1.4 Inference

The agent learns to mimic the behaviour of the expert and thus effectively clones the expert's biases too. Which can be seen in the lack of acceleration and the left-right jitters to try and stay within the track. However, given enough time-steps, the agent does well to collect all the track tiles.

[1] Exact architecture from the Atari paper with an additional FC layer of 256 hidden nodes
[2] Expert score is based on data collected till environment terminates episode (done=True)

2 DQN

2.1 Hyperparameters

- Architecture[1] (for Car Racing)
  - # of convolution layers: 2
  - # of fully connected layers: 1
- Batch size: *8* for cart pole; *16* for car racing
- Learning Rate: *1e-4*
- Optimizer: *Adam*
- ϵ=0.1 (Linearly decayed from 1 till 10% of total episodes and kept constant subsequently)
- Frames skipped: 5 (for Car Racing)

2.2 Cart Pole

Exploration comprised of decaying ϵ linearly from 1 to 0.1 from episode 1 to episode 25.

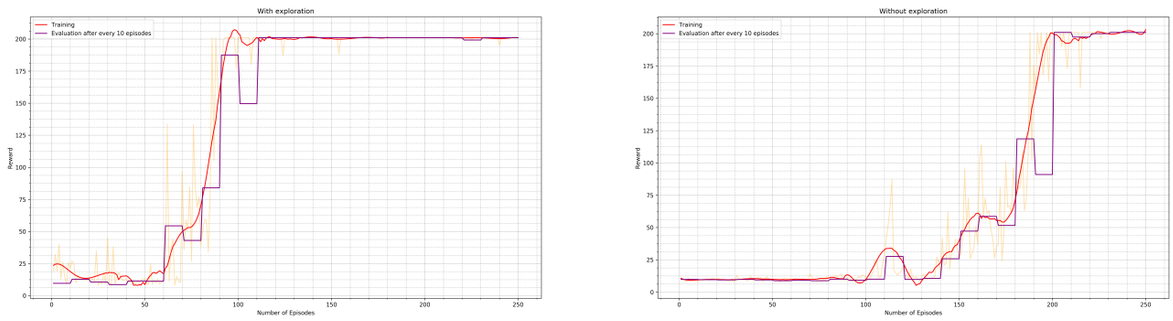

Figure 3: Training and Evaluation reward curves for Cart Pole during training phase

For 100 test episodes:

- Model with exploration: Mean of *200.69*[2] and Standard Deviation of *1.25*
- Model without exploration: Mean of *201* and Standard Deviation of *0*

2.3 Car Racing

Exploration comprised of decaying ϵ linearly from 1 to 0.1 from episode 1 to episode 1000.
While the maximum time-steps was linearly increased from 200 to 1000 from episode 1 to 2000.

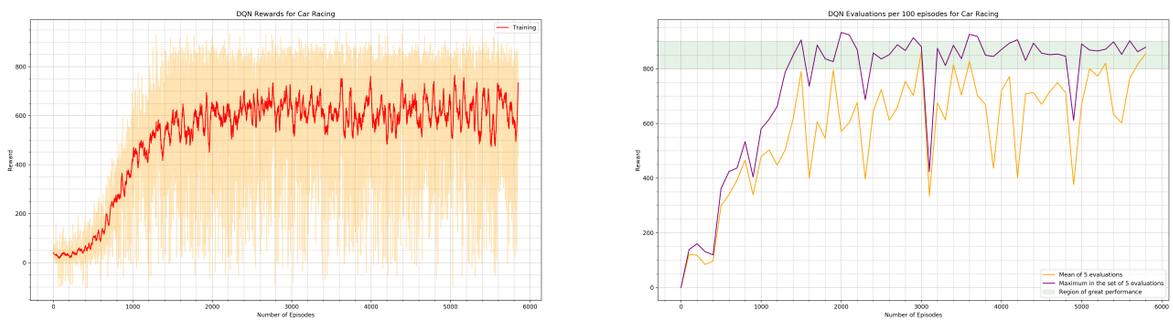

Figure 4: Reward curves for Car Racing

# of Episodes	$\|Left\| = \|Right\| = 1$	$\|Left\| = \|Right\| = 0.5$
1500	(680.32, 105.85)	(677.25, 129.28)
3000	(739.98, 86.96)	(664.78, 188.99)
4100	(738.87, 56.3)	(691.02, 74.74)
5600	(724.9, 145.67)	(**767.7**, 52.54)

Evaluation (Mean, Std.Dev) over 15 episodes for models after completion of given bumber of episodes. The two columns represent the maximum magnitude of left-right actions used during evaluation.

2.4 Inference

The agent appears to have learnt that it has to go around the track as fast as possible to get more rewards. For the same, it effectively never uses BRAKE and uses left-right actions to induce braking. As a result, the motion is often jittery and despite learning how to navigate the track, the agent overshoots sharp hair pin turns. However, as long as the the track is visible in the frame, the agent shows it can often recover.

[1] Exact architecture from the Atari paper
[2] Slight bug in step counter made each episode of 201 time-steps instead of 200