(CS 533) Intelligent and Decision Making

Homework 4:

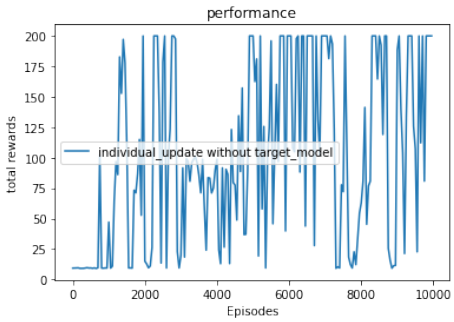
Distributed Deep Q-Learning (DQN)

A-young Kang (Student ID: 933-610-350)

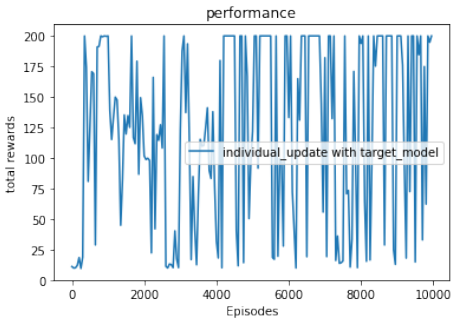
Part 1. Non-distributed DQN

1. Summary of observations

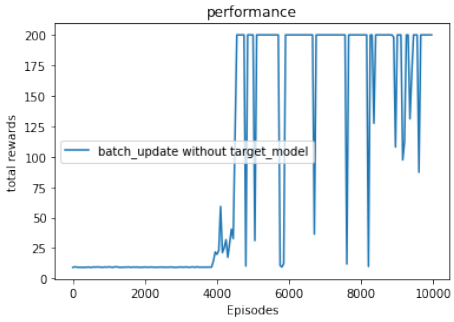
Figure 1 (a) and (b) show the learning curves *without* a replay buffer. Since it doesn't have a replay buffer, the update is based on the highly correlated sequence of examples. Thus, the total reward in both cases does not converge to some point. When DQN has a target network, it stabilizes learning by making the targets more stable. Therefore, the learning curve in (b) reaches the total reward of 200 earlier than the one in (a). Figure 1 (c) and (d) show learning curves *with* a replay buffer. Although there are some fluctuations, the total reward converges to 200 after around 5,000 and 4,000 episodes, respectively. We can also observe full-DQN has a more stable learning curve than DQN without a target network in (c).



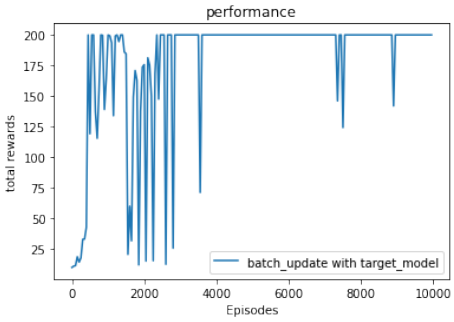
(a) DQN **without** a replay buffer and **without** a target network



(b) DQN **without** a replay buffer and **with** a target network



(c) DQN **with** a replay buffer and **without** a target network



(d) DQN **with** a replay buffer and **with** a target network

Figure 1. Learning Curves (Non-distributed)

2. Parameters used

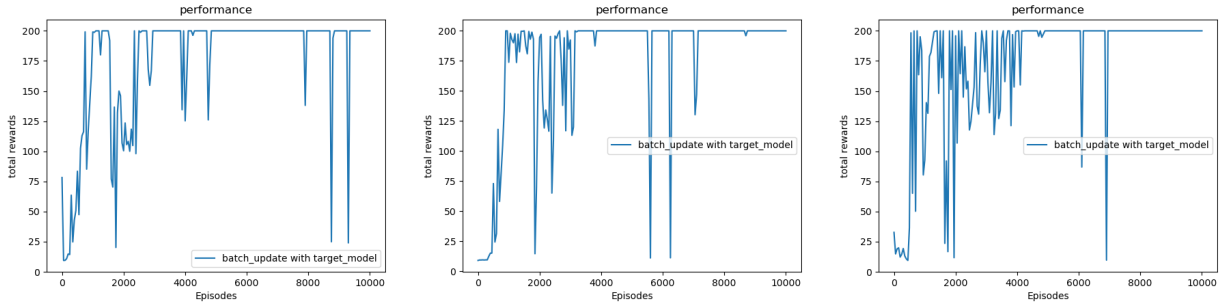
'epsilon_decay_steps' : 100000, 'final_epsilon' : 0.1, 'batch_size' : 1, 'update_steps' : 1, 'memory_size' : 1, 'beta' : 0.99, 'model_replace_freq' : 2000, 'learning_rate' : 0.0003 'use_target_model': False	'epsilon_decay_steps' : 100000, 'final_epsilon' : 0.1, 'batch_size' : 1, 'update_steps' : 1, 'memory_size' : 1, 'beta' : 0.99, 'model_replace_freq' : 2000, 'learning_rate' : 0.0003, 'use_target_model': True
(a) DQN **without** a replay buffer and **without** a target network	(b) DQN **without** a replay buffer and **with** a target network
'epsilon_decay_steps' : 100000, 'final_epsilon' : 0.1, 'batch_size' : 32, 'update_steps' : 10, 'memory_size' : 2000, 'beta' : 0.99, 'model_replace_freq' : 2000, 'learning_rate' : 0.0003, 'use_target_model': False	'epsilon_decay_steps' : 100000, 'final_epsilon' : 0.1, 'batch_size' : 32, 'update_steps' : 10, 'memory_size' : 2000, 'beta' : 0.99, 'model_replace_freq' : 2000, 'learning_rate' : 0.0003, 'use_target_model': True
(c) DQN **with** a replay buffer and **without** a target network	(d) DQN **with** a replay buffer and **with** a target network

Table 1. Parameters used

Part 2. Distributed DQN

1. Summary of observations

Figure 2 shows the learning curves with different collector workers. The total reward converges to 200 after around 4,000 training episodes as in the non-distributed case. As the number of collection workers increases, the learning time decreases. However, as the number of collector workers increases from 8 to 12, the decrease in learning time is not as significant as from 4 to 8. Since we always have the same number of evaluators in all cases, just increasing the number of collection workers would not have a large impact on decreasing learning time.



(a) Collector workers = 4 (b) Collector workers = 8 (c) Collector workers = 12

Figure 2. Learning curves versus the number of collectors

# of workers	Learning time (sec)
Collector workers = 4 **Evaluator workers = 4**	6206.70823931694
Collector workers = 8 **Evaluator workers = 4**	3868.8533148765564
Collector workers = 12 **Evaluator workers = 4**	3392.695325613022

Table 2. Learning time