Applied Deep Learning Assignment 5

資工所 碩一 江東峻 R05922027

December 26, 2016

1 Problem Description

使用 Deep Q-Network 模型來學習 Atari 中的打磚塊遊戲 (breakout)，這是一個 reinforcement learngin task，目標是盡量達到高分。

2 DQN Functions

2.1 Experience replay

為了避免 training data 之間有相依性，在 train 時，會從之前的歷史紀錄中 sample 一些 data(uniformly, 或依照 reward 的權重)，這樣可以打亂 data 之間的相依性，讓每個 batch 趨近於 i.i.d.. 另外，replay 也可以讓一些比較有價值的紀錄重複被 train 到。

2.2 Target network

$$L(w) = \mathbb{E}[(r + \gamma max_{a'} Q(s', a', w^-) - Q(s, a, w))^2] \tag{1}$$

Eq. 1 是 DQN 的 objective function，target network 負責預測經過某個 action 後，能獲得的分數，在 implement 時，其實是一個 CNN classifier，input 是 4 個 screen，output 是 4 種 action 中機率最高的那個。
在 training 時，因為在 Q-network 和 Q-learning target 都有 $Q()$，所以 Q 的值有可能是極大或者極小 (oscillation)，因此，在 train Q-network 時，會固定 Q-learning target 的參數 (w^-)，這樣也可以將兩邊的 Q() 的 gradients 計算分開，在 update 數個 $Q()$ 後，再更新 Q-leanring target 的參數。

2.3 Epsilon greedy

Epsilon greedy 是讓 model 有極小的機率 (0.05) 會 random 決定一個 action(exploration)，讓 model 有機會找到新的決定。在 training 時，epsilon 會從 1.0 降到 0.1，一開始要積極的嘗試各種 actions(exploration rate 大)，後面要降到夠低，讓 model 較能夠基於自己的決定做 training(否則全部隨機的話很難玩到高分)。

2.4 Clip reward

為了避免 $Q()$ 的 value 過大 (too sensitive)，會把 reward 值 clip 在 (-1, 1)，讓 Q-network 不要 update 太多，可以避免 oscillation 的問題，這樣也能讓 Q-network 和 Q-learning target 不會差異太多 (因為在 training 的時候，Q-network 和 Q-learning target 是分開 update 的，但理論上應該要越像越好。)

3 DQN for Two Actions

把 action 變成兩兩的 pair(不重複)，所以如果有 4 種 action，就會變成 C_2^4(按兩個按鍵) +4(只按一個按鍵) = 10 種可能性當作新的 action set，如果選到兩個 action 相同的 action pair，那就只做一次 action。如 Fig. 1 中，除了方向鍵以外，也把方向鍵 + 功能鍵的組合也加入 action set。

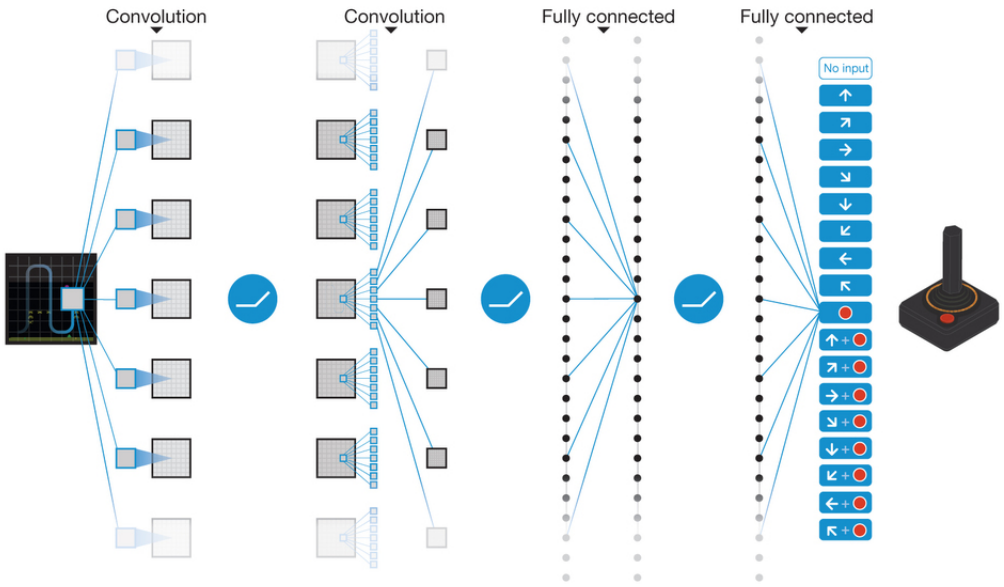

Figure 1: Q-network for multi-actions in Mnih, Volodymyr, et al. Nature 15

4 Experiment

在實驗中，我們跑 20 次打磚塊遊戲，每一次遊戲，都從 (0, 10000) 中隨機 sample 出一組 init_seed 以及 init_rand，最後紀錄 20 次遊戲的平均及標準差。

game	avg. score	std. score	min. score	max. score	time
breakout	257	77	140	324	397

Table 1: Experiment result

References

[1] gtoubassi, *A TensorFlow based implementation of the DeepMind Atari playing Deep Q LearningÄgent*

[2] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra and Martin Riedmiller, *Playing Atari with Deep Reinforcement Learning*, arXiv:1312.5602 (2013).

[3] Mnih, Volodymyr, et al., *Human-level control through deep reinforcement learning*, Nature 518, 529–533 (26 February 2015) doi:10.1038/nature14236.