

# ADL HW5 Report

電信二  
r04942056  
余朗祺

這次作業我主要是參照 <https://github.com/devsisters/DQN-tensorflow> 來實作。故基於這個檔案內容來回答以下問題。

- **Explain why DQN algorithm need these function and how you implement them:**

- **Experience replay (1%)**

讓 training data 中相互的關聯性、correlation 降低，更“乾淨”。實作上，我們有一個 class ReplayMemory 專門用來儲存 observe 過的 state, reward, action 等相關資訊，以便之後拿出來用。

- **Target network (1%)**

避免 training 產生“oscillation”，也減少 DQN 網路和 target 間不必要的 correlation. 這部分其實就跟定義一樣，不要每一步都更新 Q-network 參數，固定一定步數後再更新即可。

- **epsilon greedy (1%)**

在選擇 action 時，有  $\epsilon$  的機率會採取隨機的 action,  $(1-\epsilon)$  機率會照目前 policy 選最好 action, 增加 DQN 探索更好 policy 的機會。這個定義相當 trivial，很好實做。

- **clip reward (1%)**

避免 update 時使用到太大的 Q value, 概念類似 SGD 中的 gradient clipping, 讓學習過程更穩定。實作上就和 gradient clipping 一樣，檢查 Q-value 值，若太大/小就改用一個邊界值， $[-1, 1]$  之類的。

- **If a game can perform two actions at the same time, how will you solve this problem using DQN algorithm ?(there's no absolute answer) (2%)**

利用 experience replay, 在當下先採取其中一種 action, 之後 replay 時可以再採取另外一種 action. 這個方法還可以推廣到同時有 3 個以上 actions 的遊戲。