HW3 Report

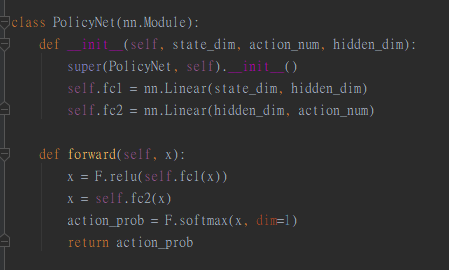
姓名：黃浩恩

學號：f05921120

Coworker：賴棹沅、謝忱、嚴聲揚

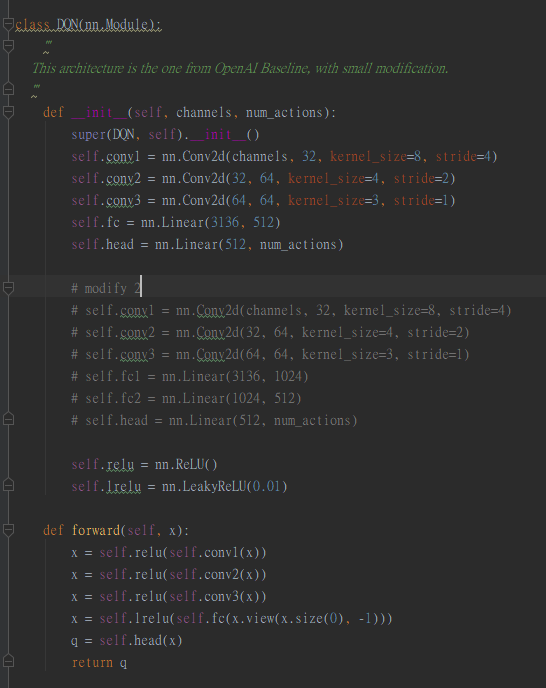
1. Basic Performance (6%)
   1. Describe your Policy Gradient & DQN model (1% + 1%)
      1. Policy Gradient

Policy Gradient的model，是透過NN將state當作輸入，action當作輸出，使NN直接訓練目前的state對應的action，不分析Q值，透過Categorical的distributions直接輸出行為，並學習調透過reward的反應調高或降低目前選擇該action的機率。

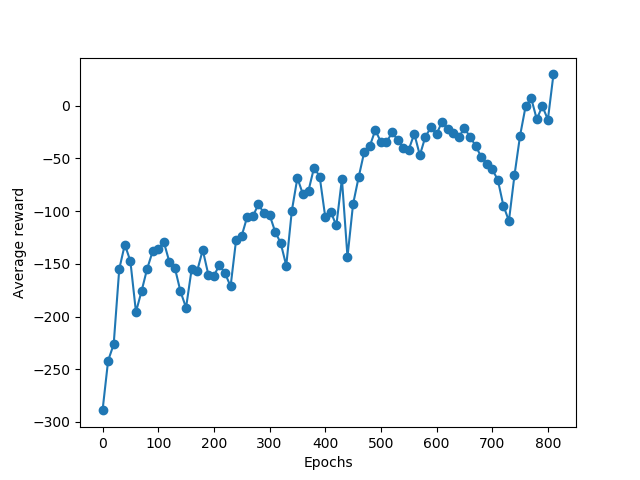


* + 1. DQN model

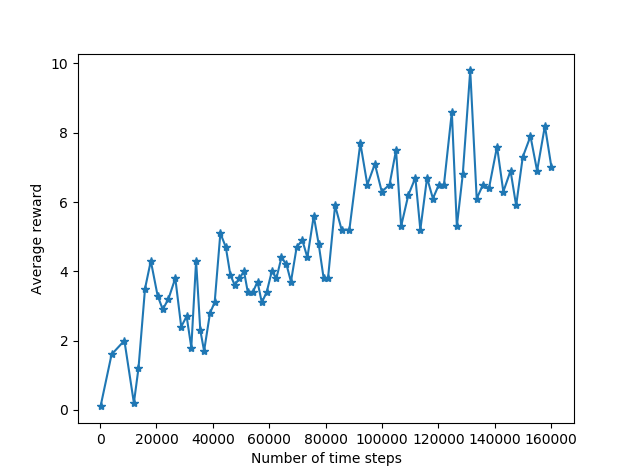
DQN一開始會先有個隨機取樣機率Epsilon，透過該機率可以選擇隨機選擇任意動作或是進入NN。隨機選擇動作是因為讓NN透過ReplayMemory歷史資料學到不同state下各動作的Q值。倘若進入NN，DQN model的NN架構事先透過CNN將state資料採樣，後再將輸出與action連結。得到每個action的Q值後，再透過max取出最大Q值的動作，整個運作過程就是Value base的運作方式。其中因為Q-learning是一種off-policy的離線學習，可以學習online與過往歷史經歷，透過ReplayMemory將過去玩過的歷史資料存取下來，隨機抽取來學習打斷資料連續相關性的問題(target\_net)。



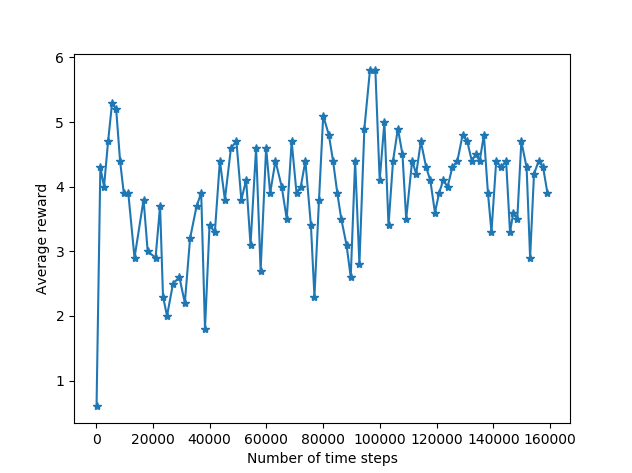
* 1. Plot the learning curve to show the performance of your Policy Gradient on LunarLander (2%)



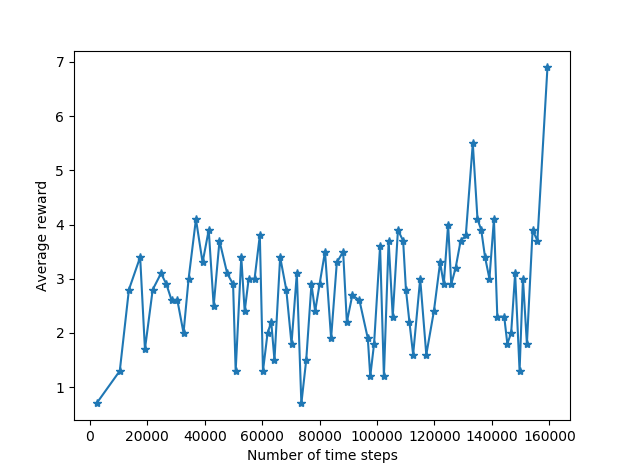
* 1. Plot the learning curve to show the performance of your DQN on Assualt (2%)



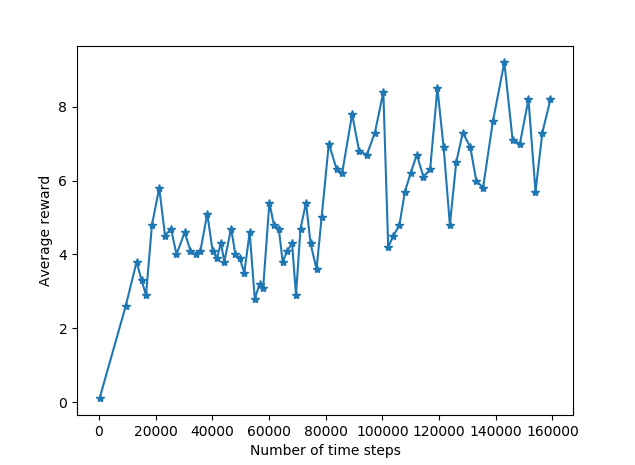
1. Experimenting with DQN hyperparameters (2%)
   1. Plot all four learning curves in the same figure (1%)
      1. 更改GAMMA



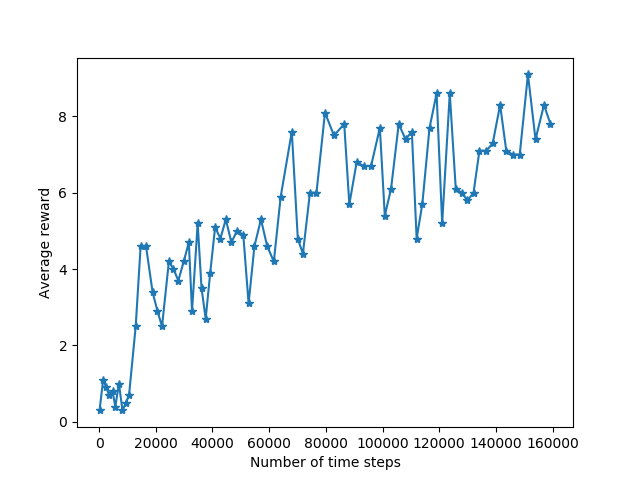
* + 1. 將replay memory從10000調低為100



* + 1. 將DQN神經架構做更改，增加一層Linear



* + 1. 提升target\_update\_freq與batch\_size使分數提升較快

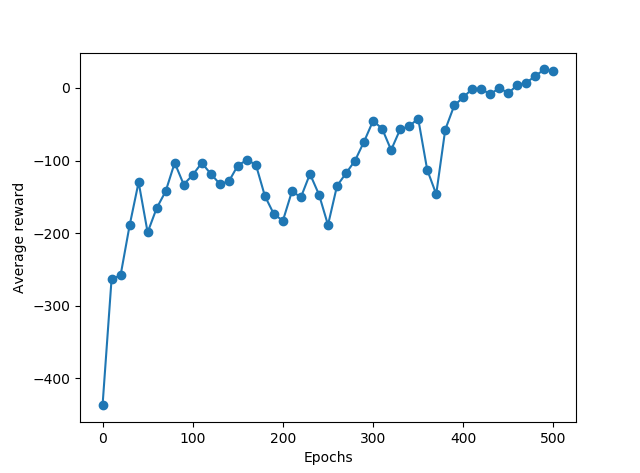


* 1. Explain why you choose this hyperparameter and how it affect the results (0.5% + 0.5%)
     1. 更改GAMMA，使得原本的GAMMA=0.99變為GAMMA=0.75，讓學習歷史資料的衰減率增加。由圖可見，學習的平均獎勵以0.99優於0.75，表示歷史學習之關係度仍為重要。
     2. 將replay memory從10000調低為100，由圖可見，效果非常不好，原因可能為存在memory的資料，即使隨機選取的資料關聯度依然太高，無法從更多不同之state與action中學到資訊。
     3. 將DQN神經架構做更改，增加一層Linear於CNN接至Linear後，來觀察原先助教提供的架構之神經元個數的差距跳太大與較緩和的下降的比較。由圖可見，平均分比起原先得來的高，可能是透過該層Linear擷取了更多的資訊，在將state mapping 到action上做得更加完整。
     4. 提升target\_update\_freq，在參數調整上從原本的1000調降至200，使200步就更新一次，並再將batch\_size提升一倍，從32調至64，使訓練速度上升速度增快，再訓練過程中，以上3種更改方法皆耗時約20分鐘，但此次僅18分鐘就完成。由圖可知，進步效率較快，於最後的平均分數也與原本的差不多，效果不錯。

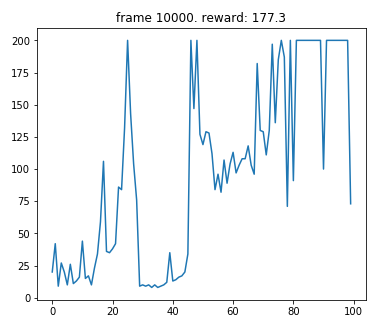
1. Improvements to Policy Gradient & DQN / Other RL methods (2% + 2%)
   1. describe why they can improve the performance (1%)

**Actor-Critic是一個結合value-base與policy-base的架構。針對policy-base部分，以每回合結束來更新一次，但是value-base的學習卻可以很快速的更新，除了可以針對連續動作來做訓練、學習外，亦可以像一個眼睛，看著action的動作、reward的好壞，來進行超脫訓練的學習。**

* 1. Plot the graph to compare results with and without improvement (1%)
     1. **將LunarLander-v2用A2C來做improvements，可以看到如果照著作業的目標，當avg\_reward > 50就停止的話，則原先的PG需要花上約800個step才能達到目標，但是使用A2C的話，僅需要500個step就可以達到目標。可見訓練速度幾乎快上PG要上1.6倍。**



* + 1. 傳統的DQN普遍會過高估計Action的Q值，而且估計誤差會隨Action的個數增加而增加。如果高估不是均勻的，則會導致某個次優的Action高估的Q值超過了最優Action的Q值，永遠無法找到最優的策略。在許多基於視覺的感知的DRL任務中，不同的狀態動作對的值函數是不同的，但是在某些狀態下，值函數的大小與動作無關。於” CartPole-v0”遊戲中的結果如下。



他的improvement之DDQN結果如下：

