顏廷宇
資工四 B03902052
2017年12月16日

ADLxMLDS HW3

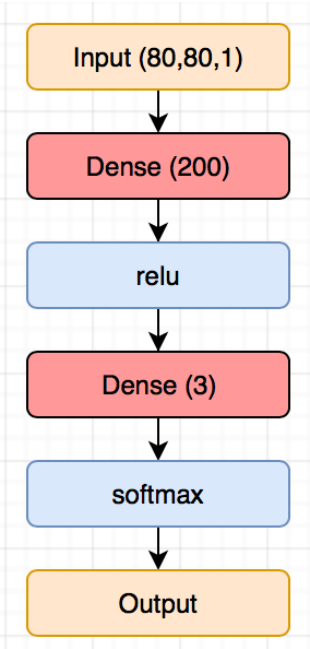
1. Basic Performance

A, Policy Gradient Model :

* 訓練流程 :

在每個動作後，將得的的 observation 轉成 (80,80,1) 的格式，model 用這個 observation 得到下一步動作的機率分佈，用機率最高的動作來得到下一個 observation，並把 (observation, action, reward)都存起來，直到這次的遊戲結束。一場遊戲結束後，會先將存起來的遊戲紀錄做 discount 和 normalize，再用這些記錄去算 gradient 以更新 model。

* 模型架構 :



* 參數 :

LEARNING_RATE = 0.0001

Gamma = 0.99

RMSProp, decay=0.90

B, DQN model :

* 訓練流程 :

先讓model隨機的玩一段時間，並把這段時間的遊戲紀錄 (observation, action , reward, done) 存在 replay_memory 中。 之後，在邊玩邊從 replay_memory 中隨機抽取紀錄來計算 gradient，更新 target network 和 online network。

* 模型架構 :

Conv2d, out_channels=32, kernel_size=8, stride=4, activation=relu

Conv2d, out_channels=64, kernel_size=4, stride=2, activation=relu

Conv2d, out_channels=64, kernel_size=3, stride=1, activation=relu

Dense, neurons=512, activation=relu

Dense, neurons=3

* 參數 :

Replay Memory Size = 10000

Learning Start = 10000

Target Network Update Frequency = 1000

Online Network Update Frequency = 4

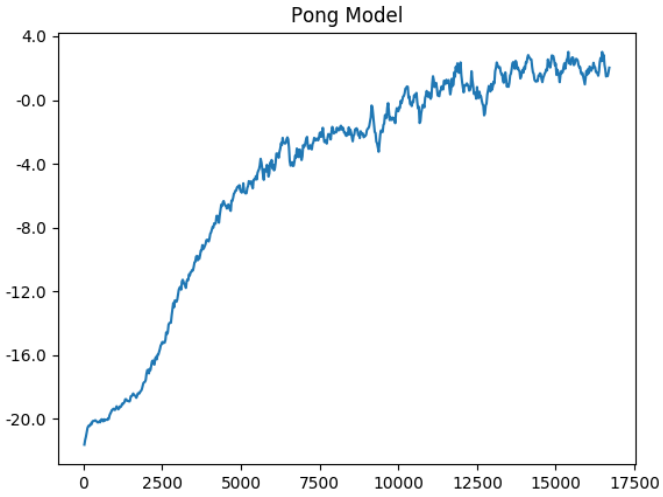
Gamma = 0.99

Batch size = 32

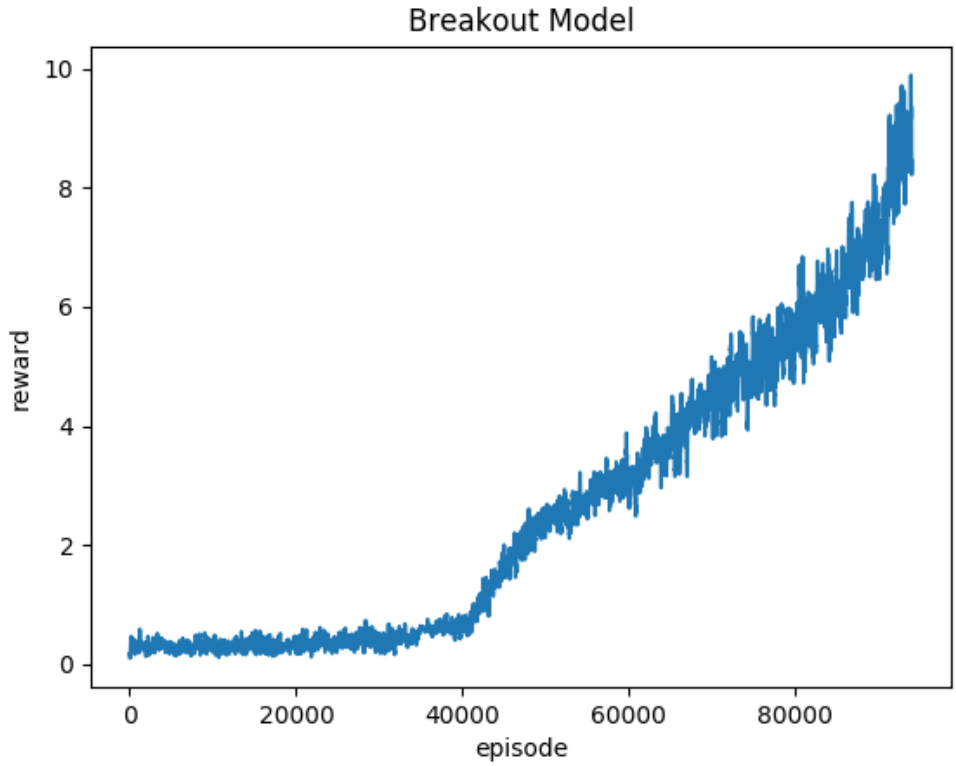
LEARNING_RATE = 0.0001

RMSProp, decay=0.90

C, Learning Curve of Policy Gradient Model on Pong :

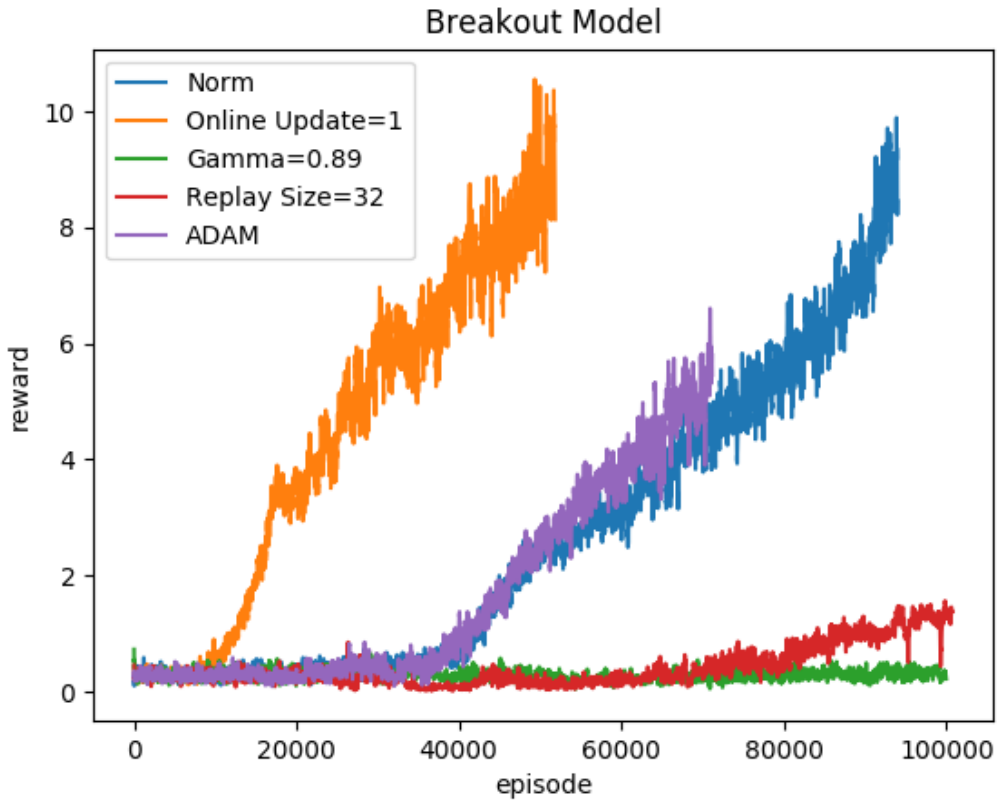


D, Learning Curve of DQN Model on Breakout :



2. Experimenting with DQN hyperparameters

A, Learning Curve of different DQN Model on Breakout :



B, Why I choose this hyperparameter and how it effect the results :

* Online Network Update Frequency = 1:

選擇這個參數的原因，是想看出，增加 online network 更新頻率，能否讓網路學得更快。從表中可以看出，增加更新頻率的確可以讓學習的速度增加。

* Gamma = 0.89 :

選擇這個參數的原因，是想看出不同的gamma 值，網路能不能學的起來。從表中可以看出，較小的 gamma 值會讓後面的 reward 變太小，導致網路學不起來。

* Replay Memory Size = 32 :

選擇這個參數的原因，是想看出 replay memory 對模型的影響。從表中可以看出，減少 replay memory後，學習的多樣性降低，導致網路學不起來。

* Optimizer = Adam :

選擇這個參數的原因，是想了解不同 optimizer 對模型的影響。從表中可以看出，換成 Adam後，學習的成效其實是差不多的，甚至有稍微好一點的趨勢。

3. Bonus

* Implement other advanced RL method : A3C on Pong

A3C 是由一個 global network 和數個 worker network 組成的模型，每個 worker 會先從 global network 中複製網路過來，經過一場的遊戲後，算出 gradient 來更新 global network ，接下來再回去複製 global network，一直做這個循環。而 network 中，會用 Q-learning 的 value 來更新 Policy。

A3C 有多個 worker 來增加 network 的多樣性，加速 model 更新的速度，並在 network 中結合了 policy gradient 和 Q-learning 的方式，讓他表現比原本的方式好。