學號：B03705049 系級： 資管四 姓名：廖寬璿

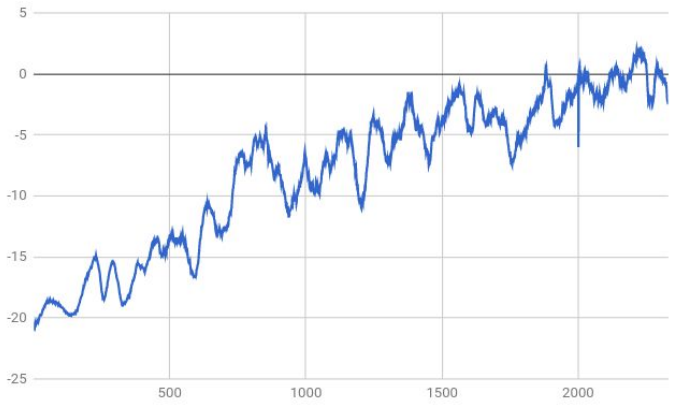
Policy Gradient

Model

Model是2層conv2d接2層dense
gamma : 0.997, optimizer:Adam, learning rate: 0.001
Loss是categorical crossentropy
每結束一個episode update一次
discount的reward在每球reset

Learning Curve

這是最後的結果, 因為model不是很穩定train到一半就會掉下去所以是拿同樣的code在2000episode的checkpoint來繼續train的



DQN

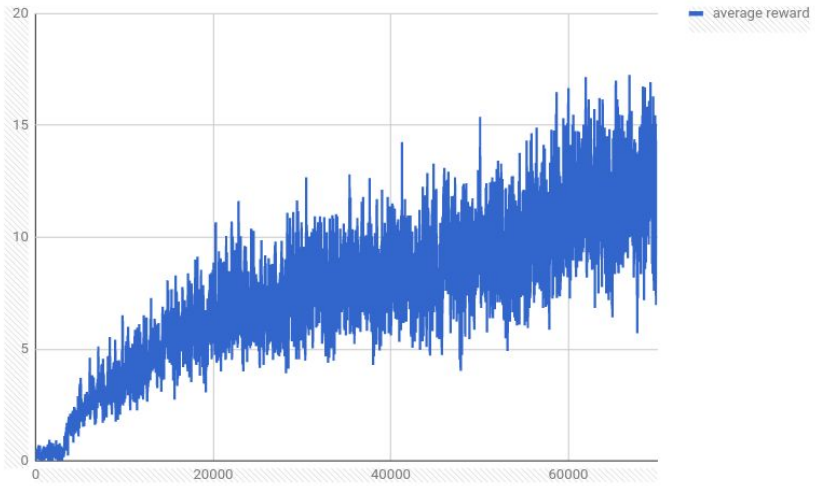
Model

以下的實驗用的 baseline 是兩層的 (conv2d + maxpool) 接一層 dense
Epsilon 固定 0.01, gamma 0.99, memory size 50000
loss: huber loss, optimizer: Adam, learning rate: 0.001
每結束一個episode 就update online network
每100episode update target network

交上去的模型是 Double DQN
跟上面一樣只是update 的target Q value是用online predict出來最大值選target 的output

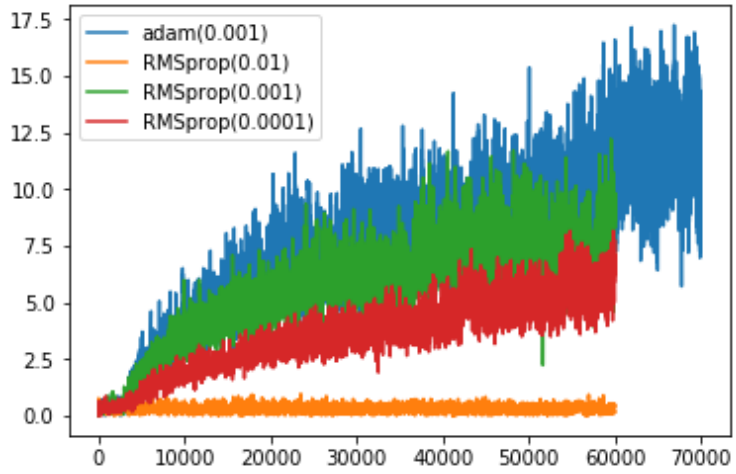
Learning Curve

以下是baseline 的 learning curve



Hyperparameter Search

我選來調的是 optimizer/learning rate
原因是看到大家都在說要用RMSprop但握自己一開始在寫的後RMSprop一開始train的結果都不太樂觀, 同時助教也有講說不要用adam但我卻是adam train出來的



用的模型跟上面baseline一樣只改了optimizer/learning rate
我的實驗結果是adam還是比RMSprop好, 但我在實驗跑到一半的時候才看到社團裡的流言說助教用的pytorch預設值跟我用的keras的不一樣, 同學也有說lkeras的rho 從0.9 -> 0.99 會好很多

Improvements

Double DQN

改變target Q values的選擇方法 從 max(target_Q) -> target_Q[argmax(online_Q)]
目標是減少因未local minimum造成的error, 所以把 選擇 跟 用的值分開算