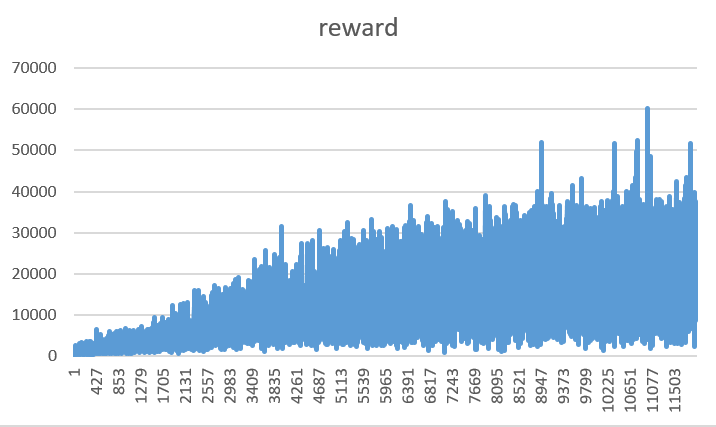
清華大學資工系

大二 學號: A053095 林柏淵

A plot shows episode rewards of at least 100,000 training episodes



 Explain the mechanism of TD(0):

要講TD(0)之前首先得先了解甚麼是TD，TD的反面就是MC(Monte Carlo)，MC最大的特點在於在所有state結束後以最終state的值對前面的state做update，而TD正好相反，TD不用用最後面的state來update而反之是用藥update的state後面幾個state來做update (ex:state2來update state1，或state5來update state1)而這就出現了另外一個表示法TD(λ)這代表的是λ越大，其跨越的state會越多，而我們可以視MC為一種TD(λ) 只是λ=end，而這邊主要是敘述TD(λ) λ=0時的情況，也就是state的update發生在隔壁，也就是(ex:state3 update state2，state6 update state5)

 **Describe how to train and use a V(state) network.**

V(state)主要是用random值後的數去估計他並且update他，他再行動選擇方面為再決定4個action(上下左右)後，將afterstate的值依各種有可能random到的地方做期望值估計，再將他跟reward相加就是做為選擇要做的action，再來是結束時的update部分，我們先將final state值做固定好讓update時有個基準可以校正，再來依序將random過的state從後到前依序update回去。

 **Describe how to train and use a V(after\_state) network.**

V(after\_state)跟V(state)不同在於，前者是用還沒random過的state做計算，後者是random過的，所以前者在forward 選擇action上比較輕鬆(因為不用估計random的期望值)只要比對四個action之間的估計值即可做出決定，然後再update方面也是針對after\_state做下手其做法就跟V(state)類似，只是將update對象由state改成afterstate。

 **Describe how the code work (the whole code)**

這個board中的是用long long type來存然後用16進位表示每一個位數代表一格數字，然後主要的class有

board:用來當作平台上面有數字並且提供函數來做上下左右移，並可以print出整個board。

Feature:用來更新tuple的parameter，估計board的值，。

State:用來更新state中所有的狀態，有一開始的狀態還會記錄action和action後的

reward及action後的state(還沒random)。

Learning:負責forward program和backwardtraining也是這次我們要主要修改的地方，另外還有儲存和讀取model的功能，還有將每1000場遊玩的狀況條列出來

* More I want to say:

說實在這次真的是另一種方面累，在於突然從pytorch轉換到C++上，雖然我對C++還算熟悉不過，在DL領域我還是第一次聽到可以用C++來train AI，而這次lab我學到最重要的東西就是，【你implement東西一定要有理論基礎】，沒理論基礎，trace code起來會非常痛苦，因為當output出錯時，你不知道你是code出問題還是演算法出問題，至少你懂理論基礎後code打起來也會輕鬆很多