Question 1(20%)

Please see the 11 line in the following algorithm, what is the corresponding concept to the Q-table method? Why it can't be coded like Q-table method?

Q-table method 是 tabular method，適用於環境簡單、狀態數量不多，通常代理人拜訪所有狀態的次數夠多，狀態價值函數或動作價值函數估測通常會收斂，而使用表格解法不影響週遭的價值函數值。

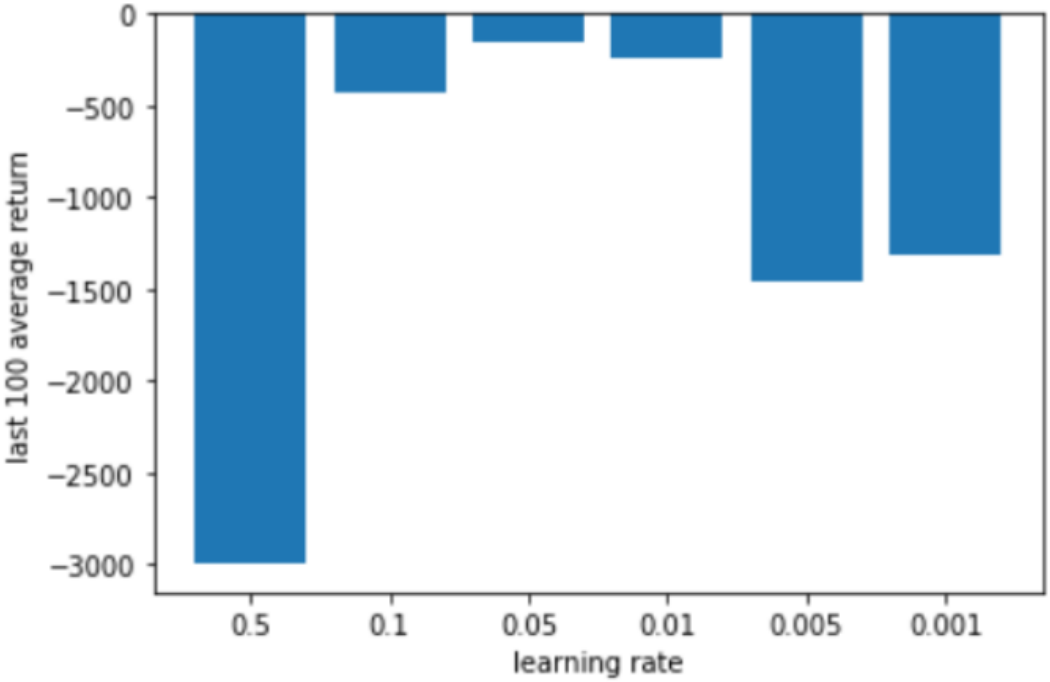
但在現實環境中往往相當複雜，且狀態數量龐大，Q-table 難以明確定義，代理人甚至無法通過有限次的狀態拜訪保證演算法的收斂性，因此通常會使用 Deep Q-network 這類 Approximation method 來處理問題。但此法在更新狀態動作價值部分便不能像 tabular method 一樣直接更新，由於狀態及動作價值無法明確定義。

使用近似解法時，須將狀態價值函數或動作價值函數做參數化，因此在演算法(6.8) 11 line 必須使用 tile coding 或 ANN 將特徵抽樣出來再透過梯度法更新，更新參數等同於更新狀態或動作價值函數，因為是做參數的更新而不是表格解法中點的更新，所以鄰近的狀態或動作價值函數也會有一定程度的更新。

Experiment 1 (20%)

In Sarsa or Q-learning set learning rate to 0.9~0.1 usually work well, but in approximation method set learning rate to 0.1 is too big, please test the learning rate(self.alpha) in these numbers [0.5 0.1 0.05 0.01 0.005 0.001], and make a graph, x-axis is learning rate y-axis is average of last 100 episode returns, you can break your simulation and set average return to -500,000,when an episode have return lower than -500,000.

比較不同的 learning rate 造成的 returns，可以觀察到 alpha 為 0.5 時，得到的平均 reward 值非常的差，若學習率高則參數更新步長太大，會使參數難以收斂到最佳值，同樣地假如學習率太低則更新步長太小，也難以收斂。根據模擬圖示顯示，最佳 learning rate 為 0.05。



Experiment 2 (60%)

Example code have 2 tiling, please make the 3 tiling version, the average of last 100 episode returns must higher than -300, plot your result like example.

最後 100 個 episode 平均 returns 大小的部分，雖然在多次的模擬中會有幾次仍小於-300，大部分都能維持在-200～-300 左右，最後平均 reward= -237.44 小於 -300。