(1) Test Environment

(state, action, next state) = reward

(0,2,2) + (2,1,1) + (1,0,0) + (0,2,2) + (2,1,1) = 0.0 + 3.0 + 0.1 + 0.0 + 3.0 = 6.1

This is the maximum reward that can be obtained in a single trajectory.

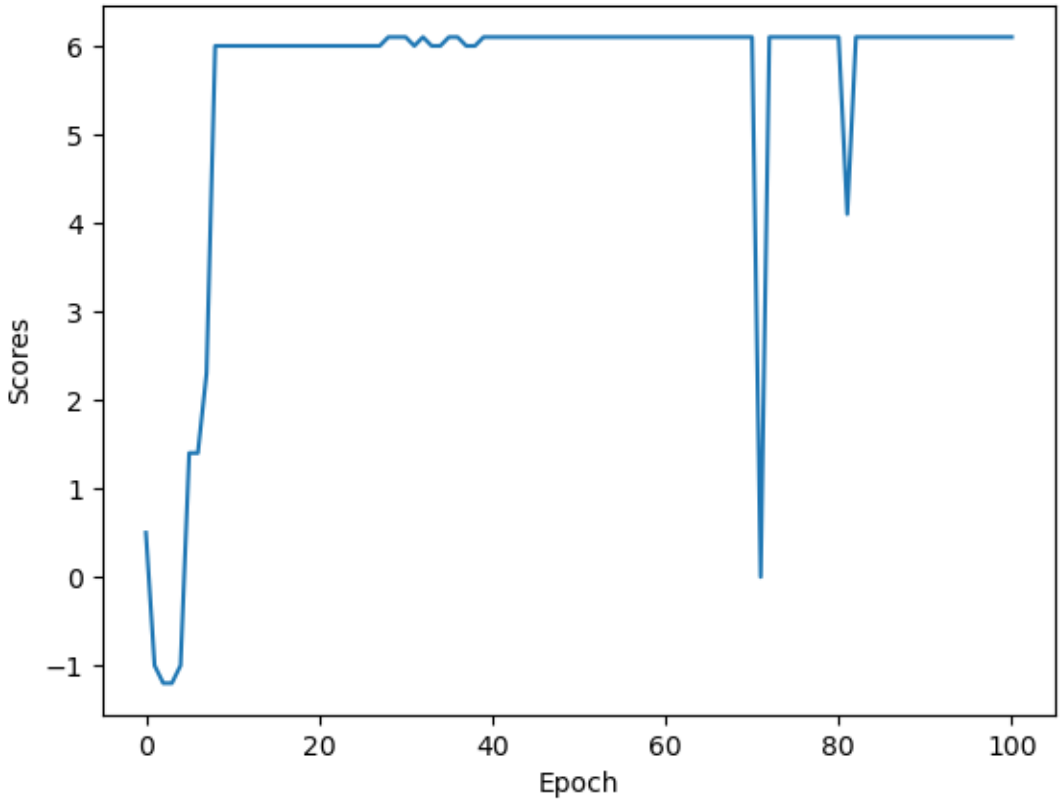
(4) Linear Approximation

a) Let δ(s, a) be $w_{s,a}$ and consider the case where δ(s, a) = 1. From this, we can get

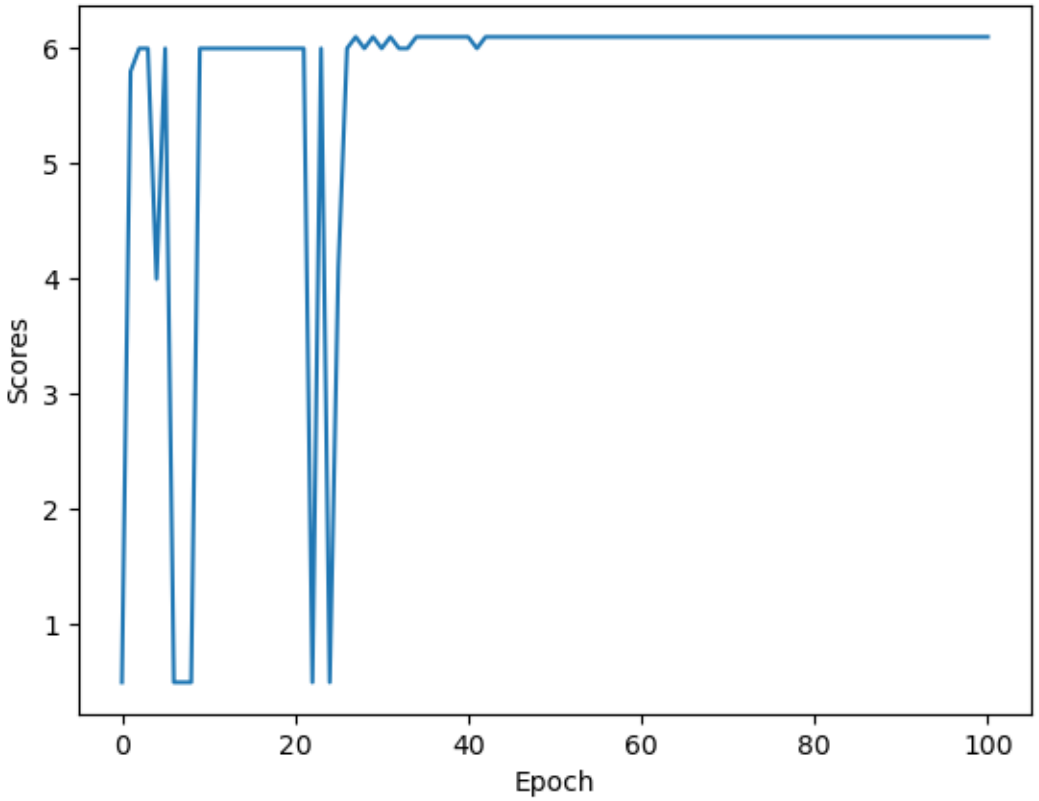
$w_{s,a} = w_{s,a} + \alpha (r + \gamma \max w_{s', a'} - w_{s,a}) \nabla w \, w_{s,a}$

Which is the same as equation 1 when $w_{s,a}$ is replaced with Q(s,a).

b)



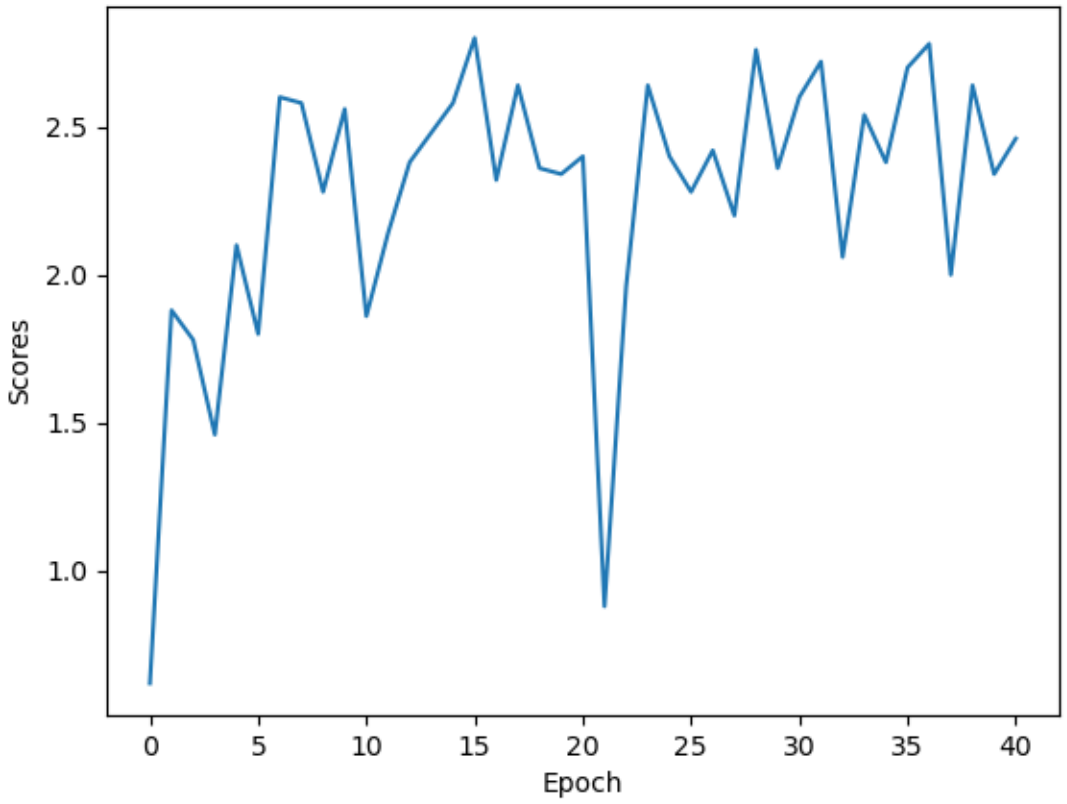
(5) Implementing DeepMind’s DQN



This reached the same final performance as linear approximation. They both took a similar amount of time to run.

(6) DQN on MinAtar

a)



b)

c)

The linear approximation model is not complex enough to match the performance of convolutional networks when analyzing images.

d)

DQN uses off-policy evaluation and uses epsilon-greedy for exploration. It is not guaranteed to improve monotonically.