1. the optimal trajectory:

$S_0 = 0 \quad a_0 = 2 \quad r_0 = 0$

$S_1 = 2 \quad a_1 = 1 \quad r_1 = 2$

$S_2 = 1 \quad a_2 = 2 \quad r_2 = 0$

$S_3 = 2 \quad a_3 = 1 \quad r_3 = 2$

$S_4 = 1 \quad a_4 = 0 \quad r_4 = 0.1$

Maximality - the maximal reward is $\overset{\text{the action:}}{2 \to 1}$. We start from 0, go to 1 and the we have to come back to 2, to do this action again. In the last action we'll just do the action with the maximal reward in this step.

Q - Learning

1. By representing the q function as $\mathbb{R}^{|A|}$ we can represent the q value for each action and choose easily the maximal one (We are maximizing per state, which mean that in each step we calculate only once the forward pass)

2. implemented

3. Not, because:

$$\nabla_w l(s;w) = 2\left(r + \gamma \max_{a'} q(s',a';w) - q(s,a;w)\right] \nabla_w \left(\underbrace{\max_{a'} q(s',a';w)}_{\text{also depend on } w\text{, and do not appear in equation 2}} - q(s,a;w)\right)$$

4. Yes.

$$\nabla_w l(s;w) = 2\left(r + \gamma \max_{a'} q(s',a';\bar{w}) - q(s,a;w)\right]\left[\underbrace{\nabla_w \max_{a'} q(s',a';\bar{w})}_{0} - \nabla_w q(s,a;w)\right]$$

5. if we choose small C, then we will update the learnable weights more often. Thus, on the one hand, we will tune our parameters toward more accurate weights. On the other hand, we can get close to the problem of question 3.

6. In supervised learning the examples are drawn i.i.d, while in our case D is examples that lead us to this certain state

Linear Approximation

1. (a) $W \leftarrow W + \alpha \left(r + \gamma \max_{a'} q(s', a'; w) - q(s, a; w)\right) \nabla_w q(s, a; w)$

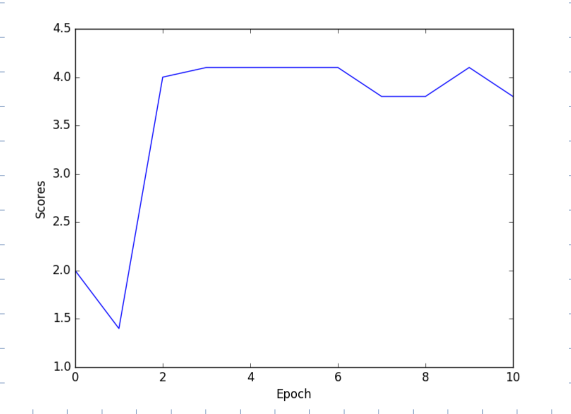
$\Rightarrow W \leftarrow W + \alpha\left(r + \max_{a' \in A} W^T \delta(s', a') - \underset{[W]_{s,a}}{\underbrace{W^T \delta(s, a)}}\right) \underset{\delta(s,a)}{\underbrace{\nabla_w [W]_{s,a}}}$

thus, we change only the coordinate of (s,a) and we get equation (1).

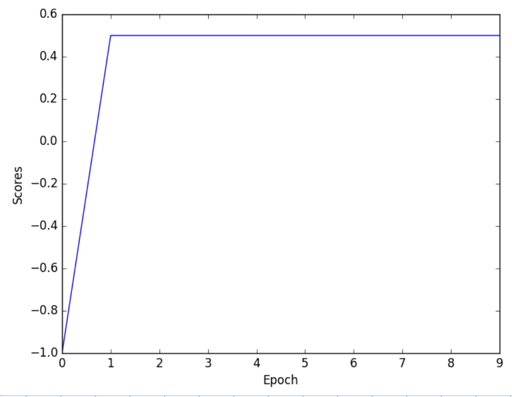
2. $\nabla_w W^T \delta(s, a) = \delta(s, a)$. The update rule is in q1.

3. done.

4. yes, we get to 4.1.



DQN



DQN on Atari

(a) No because the reward doesn't change between epochs.

(b)

n-step estimator

$$\text{Bias (1-step)} = \mathbb{E}\left[r_t + \gamma \hat{q}(s_{t+1}, a_{t+1}) \mid s_t, a_t\right] - Q^{\pi}(s_t, a_t) =$$

$$= \mathbb{E}\left[r_t + \gamma \hat{q}(s_{t+1}, a_{t+1}) \mid s_t, a_t\right] - \mathbb{E}\left[r_t + \sum_{i=t+1}^{\infty} \gamma^{i-t} r_i \mid s_t, a_t\right] =$$

$$\underset{\text{liniacity of expectation}}{\overset{=}{\downarrow}} \gamma \mathbb{E}\left[\hat{q}(s_{t+1}, a_{t+1}) \mid s_t, a_t\right] - \gamma \mathbb{E}\left[\sum_{i=t+1}^{\infty} \gamma^{i-t-1} r_i \mid s_t, a_t\right] =$$

$$= \gamma \left[\mathbb{E}\left[\hat{q}(s_{t+1}, a_{t+1}) - Q^{\pi}(s_{t+1}, a_{t+1})\right] \mid s_t, a_t\right] =$$

$$= \gamma \text{Bias}(s_{t+1}, a_{t+1} \mid s_t, a_t) \underset{\text{uniform bias + notation}}{\overset{}{\underset{\downarrow}{=}}} \gamma B$$

$$\text{Bias (n-steps)} = \mathbb{E}\left[r_t + \sum_{i=t+1}^{t+n-1} \gamma^{i-t} r_i + \gamma^n \hat{q}(s_{t+n}, a_{t+n} \mid s_t, a_t\right] - Q^{\pi}(s_t, a_t) =$$

$$= \mathbb{E}\left[r_t + \sum_{i=t+1}^{t+n-1} \gamma^{i-t} r_i + \gamma^n \hat{q}(s_{t+n}, a_{t+n} \mid s_t, a_t\right] - \mathbb{E}\left[r_t + \sum_{i=t+1}^{\infty} \gamma^{i-t} r_i \mid s_t, a_t\right] =$$

$$= \gamma^n \mathbb{E}\left[\hat{q}(s_{t+n}, a_{t+n}) \mid s_t, a_t\right] - \gamma^n \mathbb{E}\left[\sum_{i=t+n}^{\infty} \gamma^{i-t-n} r_i \mid s_t, a_t\right] =$$

$$= \gamma^n \mathbb{E}\left(\hat{q}(s_{t+n}, a_{t+n}) - Q^{\pi}(s_{t+n}, a_{t+n})\right) \mid s_t, a_t\right] =$$

$$= \gamma^n B$$

$$\gamma < 1 \Rightarrow \gamma^n < \gamma \Rightarrow \text{Bias}(1\text{step}) > \text{Bias}(n\text{-step})$$