NAME: ... KEY ....
CSCI E-89c Deep Reinforcement Learning
Part I of Assignment 9

Please consider a Markov Decision Process (MDP) with $\mathcal{S} = \{s^A, s^B, s^C\}$.

Given a particular state $s \in \mathcal{S}$, the agent is allowed to either try staying there or switching to any of the other states. Let's denote an intention to move to state s^A by a^A, to state s^B by a^B, and to state s^C by a^C. The agent does not know transition probabilities, including the distributions of rewards. There is, however, some evidence that the agent gets rewards only at the entrance to s^C; and transition MDP probabilities to/from s^A appear to be same (or nearly same) as to/from s^B.

Suppose the agent chooses policy $\pi(a^A|s) = 0.05$, $\pi(a^B|s) = 0.05$, $\pi(a^C|s) = 0.90$ for all $s \in \{s^A, s^B, s^C\}$. Because of the apparent symmetry between s^A and s^B, it makes sense to assume that $v_\pi(s^A) \approx v_\pi(s^B)$ and approximate the state-values as follows:

$$v_\pi(s) \approx \hat{v}(s, \mathbf{w}) = w_1 \cdot \mathbb{1}_{(s=s_A)} + w_1 \cdot \mathbb{1}_{(s=s_B)} + w_2 \cdot \mathbb{1}_{(s=s_C)}.$$

Please notice that $\hat{v}(s^A, \mathbf{w}) = \hat{v}(s^B, \mathbf{w})$ for any choice of weights.

Assume that the agent runs the following algorithm with $\alpha = 0.1$ and $m = 2$ for estimating v_π:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha \sum_{t=mk}^{m(k+1)-1} \left[R_{t+1} + \gamma \hat{v}(S_{t+1}, \mathbf{w}_k) - \hat{v}(S_t, \mathbf{w}_k)\right] \nabla \hat{v}(S_t, \mathbf{w}_k), \; k = 0, 1, 2, \ldots$$

This algorithm is a modification of the Semi-gradient 1-step Temporal-Difference (TD) with the model now being trained in mini-batches of size m. Please use $\gamma = 0.9$ and zero weights for $k = 0$.

If the agent observes the following sequence of states, actions, and rewards:

$$\begin{aligned}
&S_0 = s^A, A_0 = a^C, R_1 = 20, \\
&S_1 = s^C, A_1 = a^B, R_2 = 0, \\
&S_2 = s^B, A_2 = a^C, R_3 = 20, \\
&S_3 = s^C, A_3 = a^C, R_4 = 20, \\
&S_4 = s^C, A_4 = a^B, R_5 = 20, \\
&S_5 = s^C, A_5 = a^C, R_6 = 0, \\
&S_6 = s^B,
\end{aligned}$$

find (a) weights $\mathbf{w}_k$ and (b) corresponding approximations $\hat{v}(s, \mathbf{w}_k)$ for iteration step $k = 1, 2, 3$. Specifically, please fill the tables in below:

SOLUTION:

The gradient is

$$\begin{aligned}
\nabla \hat{v}(S_t, \mathbf{w}_t) =& \nabla\{w_1 \cdot \mathbb{1}_{(S_t=s_A)} + w_1 \cdot \mathbb{1}_{(S_t=s_B)} + w_2 \cdot \mathbb{1}_{(S_t=s_C)}\} \\
=& \left(\mathbb{1}_{(S_t=s_A)} + \mathbb{1}_{(S_t=s_B)}, \mathbb{1}_{(S_t=s_C)}\right)^T,
\end{aligned}$$

then the mini-batch gradient descent algorithm becomes

$$\begin{bmatrix} w_{k+1,1} \\ w_{k+1,2} \end{bmatrix} \doteq \begin{bmatrix} w_{k,1} \\ w_{k,2} \end{bmatrix} + \alpha \sum_{t=mk}^{m(k+1)-1} \left[\underbrace{R_{t+1} + \gamma \hat{v}(S_{t+1}, \mathbf{w}_k)}_{\approx v_\pi(S_t)} - \hat{v}(S_t, \mathbf{w}_k) \right] \begin{bmatrix} \mathbb{1}_{(S_t=s_A)} + \mathbb{1}_{(S_t=s_B)} \\ \mathbb{1}_{(S_t=s_C)} \end{bmatrix},$$

where $k = 0, 1, 2, \ldots$

Therefore
for $k = 0$:

$$\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} \doteq \begin{bmatrix} w_{0,1} \\ w_{0,2} \end{bmatrix} + \alpha \sum_{t=0}^{1} \left[\underbrace{R_{t+1} + \gamma \hat{v}(S_{t+1}, \mathbf{w}_0)}_{\approx v_\pi(S_t)} - \hat{v}(S_t, \mathbf{w}_0) \right] \begin{bmatrix} \mathbb{1}_{(S_t=s_A)} + \mathbb{1}_{(S_t=s_B)} \\ \mathbb{1}_{(S_t=s_C)} \end{bmatrix};$$

for $k = 1$:

$$\begin{bmatrix} w_{2,1} \\ w_{2,2} \end{bmatrix} \doteq \begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} + \alpha \sum_{t=2}^{3} \left[\underbrace{R_{t+1} + \gamma \hat{v}(S_{t+1}, \mathbf{w}_1)}_{\approx v_\pi(S_t)} - \hat{v}(S_t, \mathbf{w}_1) \right] \begin{bmatrix} \mathbb{1}_{(S_t=s_A)} + \mathbb{1}_{(S_t=s_B)} \\ \mathbb{1}_{(S_t=s_C)} \end{bmatrix};$$

for $k = 2$:

$$\begin{bmatrix} w_{3,1} \\ w_{3,2} \end{bmatrix} \doteq \begin{bmatrix} w_{2,1} \\ w_{2,2} \end{bmatrix} + \alpha \sum_{t=4}^{5} \left[\underbrace{R_{t+1} + \gamma \hat{v}(S_{t+1}, \mathbf{w}_2)}_{\approx v_\pi(S_t)} - \hat{v}(S_t, \mathbf{w}_2) \right] \begin{bmatrix} \mathbb{1}_{(S_t=s_A)} + \mathbb{1}_{(S_t=s_B)} \\ \mathbb{1}_{(S_t=s_C)} \end{bmatrix}.$$

Then

(a) weights $\mathbf{w}_k = (w_{1,k}, w_{2,k})^T$:

	$k=0$	$k=1$	$k=2$	$k=3$
$w_{1,k}$	0	2	3.8	3.8
$w_{2,k}$	0	0	2	4.122

(b) approximations $\hat{v}(s, \mathbf{w}_k)$:

	$k=0$	$k=1$	$k=2$	$k=3$
$\hat{v}(s^A, \mathbf{w}_k)$	0	2	3.8	3.8
$\hat{v}(s^B, \mathbf{w}_k)$	0	2	3.8	3.8
$\hat{v}(s^C, \mathbf{w}_k)$	0	0	2	4.122