Name: Alfred
Student No.: 2020-12788

Exercise 6

Data Science & Reinforcement Learning

Spring 2023

1. Give pseudocode for semi-gradient one-step Expected Sarsa for control. You can build on the semi-gradient Sarsa code for this question.

Episodic Semi-gradient Sarsa for Estimating $\hat{q} \approx q_*$

Input: a differentiable action-value function parameterization $\hat{q} : \mathcal{S} \times \mathcal{A} \times \mathbf{R}^d \to \mathbf{R}$
Algorithm parameters: step size $\alpha > 0$, small $\varepsilon > 0$
Initialize value-function weights $\mathbf{w} \in \mathbf{R}^d$ arbitrarily (e.g., $\mathbf{w} = \mathbf{0}$)

Loop for each episode:
  $S, A \leftarrow$ initial state and action of episode (e.g., ε-greedy)
  Loop for each step of episode:
    Take action A, observe R, S'
    If S' is terminal:
      $\mathbf{w} \leftarrow \mathbf{w} + \alpha[R - \hat{q}(S, A, \mathbf{w})]\nabla\hat{q}(S, A, \mathbf{w})$
      Go to next episode
    Choose A' as a function of $\hat{q}(S', \cdot, \mathbf{w})$ (e.g., ε-greedy)
    $\mathbf{w} \leftarrow \mathbf{w} + \alpha[R + \gamma\hat{q}(S', A', \mathbf{w}) - \hat{q}(S, A, \mathbf{w})]\nabla\hat{q}(S, A, \mathbf{w})$
    $S \leftarrow S'$
    $A \leftarrow A'$

a policy π

all again

for (S,A) pairs from ep.
  Loop for every S, A:
    take action A, observe R, S'
    If S' terminal:
      $w \leftarrow w + \alpha[R + \gamma \sum_{a'} \pi(a'|s', w) - \hat{q}(s, a, w)] \nabla \hat{q}(s,a,w)$
    go to next (S,A) pair

2. Show that tabular methods studied earlier in the course are a special case of linear function approximation. What would the feature vectors be?

$\nabla(\hat{s}_w, \vec{w}) = \vec{x}(s_k) \cdot \vec{w} = \sum_{i=1}^{n} x_i(s_k) w_i$

$= \sum_{i=1}^{n} \delta_{ik} w_i = w_k$

$\vec{x}(s_k) \sim [0, \cdots, 1, \cdots, 0]$
k-th position. only

$\hat{v}(s, w) = w^T x(s) = \sum_{i=1}^{d} w_i x_i(s)$ $w_n \cdot 1 = w_n$

(S,A) vectors are ind feature vectors one-hot-encoding, defining as vectores some length.

$w \leftarrow w + \alpha[R + \gamma \hat{q}(s', A', w) - \hat{q}(s, A, w)]$
$\nabla \hat{q}(s,A,w)$

because
$q(s,a) \leftarrow q(s,a) + \alpha[R + \gamma \hat{q}(s', A', w) - \hat{q}(s,A,w)]$
$\cdot q(s,A,w)$ $\nabla w = 1$

special case of linear function approx

3. For control with function approximation, we did not explicitly consider or give pseudocode for any Monte Carlo methods. Why is it reasonable to NOT consider Monte Carlo methods for function approximation?

- Because is too expensive computationally since the learning should go to terminal for any random policy. TD or SARSA would converge online faster.

- MC is variation/type of general TD's, won't be that different than n-step SARSA but n=T (terminal).

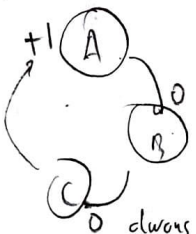
- since MC updates value function as opp. of trgt-value but TD does from current value-estimation. TD has less going deterministically around the ring. A reward of +1 is received upon arrival in **A** and otherwise as reducing variance the reward is 0. What are the differential values of the three states, as defined below?

but MC doesn't.

$\Sigma \gamma^t [\mathbb{E}_\pi [R_{t+1}(A)] = 0 + 0\gamma + 1\gamma^2 + \dots + 1\gamma^n = \frac{\gamma^2}{1-\gamma^3}$

$\mu_\pi(A) = \mu_\pi(B) = \mu_\pi(C)$ $\quad \mu_\pi(s) = \frac{1}{3}$.

$r(\pi) = \frac{1}{3}(0+0+1) = \frac{1}{3}$ $\quad v_\pi(s) := \lim_{\gamma \to 1} \lim_{h \to \infty} \sum_{t=0}^{h} \gamma^t (\mathbb{E}_\pi[R_{t+1} \mid S_0 = s] - r(\pi))$



always

$\pi(a|s)\, p(s'|s,a) = 1$ or 0 (determ.)

$V_\pi(s_{=A}) = \lim_{\gamma \to 1} \left(\frac{1}{1-\gamma^3}\right)\left(-\frac{1}{3} - \frac{1}{3}\gamma + \frac{2}{3}\gamma^2\right) = -\frac{1}{3}$

$V_\pi(s=B) = \lim_{\gamma \to 1} \left(\frac{1}{1-\gamma^3}\right)\left(-\frac{1}{3} + \frac{2}{3}\gamma - \frac{1}{3}\gamma^2\right) = 0.$

$V_\pi(s=C) = \lim_{\gamma \to 1} \left(\frac{1}{1-\gamma^3}\right)\left(\frac{2}{3} - \frac{1}{3}\gamma - \frac{1}{3}\gamma^2\right) = \frac{1}{3}$

$v(\pi) = \mu_\pi(s)$

$= \sum_s \mu_\pi(s) \sum_a \pi(a|s)$

$= \sum_s \mu_\pi(s) \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a)\, r$

5. How would you use optimistic initial values for Sarsa with linear function approximation in episodic MDP with the episode length H? Assume that your value function is parametrized with weights $\mathbf{w} \in \mathbb{R}^4$, all the feature vectors $\phi(s,a) \in \mathbb{R}^4$ are upper-bounded in L2 norm, i.e., $\|\phi(s,a)\|_2 \le 1$ for all (s,a), and the all rewards are bounded above by 1, i.e., $R(s,a) \le 1$ for all (s,a).

$w = \begin{bmatrix} \\ \end{bmatrix}_{4\times 1}$ $\quad \phi(s,a) = \begin{bmatrix} \\ \end{bmatrix}_{4\times 1}$ $\quad \langle \phi, \phi \rangle \le 1$

- Initialize value-function weights $w \in \mathbb{R}^4$ to optimistic values such as 1. to explore more initially.

- if other values, all of them should be less than

Or yet: $R(s,a) + \gamma w^T \phi(s',a')$

$1 + r + \dots + r^{H-1} \le \frac{1-r^H}{1-r}$ Since $G_{t:t+T}$ won't exceed $1 + r + \dots + r^{H-1}$

$= \frac{1-r^H}{1-r}$ that's a consideration too

But if n-step SARSA $\to \frac{1-r^n}{1-r} + r^n \|w\|_{\text{initial}}$

can be used too.