Assignment 2 -Model-free prediction and control

Reinforcement Learning, spring 2024

1. We study an environment with three states, $\mathcal{S} = \{A, B, C\}$, where C is a terminating state. The discount rate is $\gamma = 1$. A policy π is used to observe the following two episodes (states and rewards):

   $$\text{Episode 1: } A, 2, A, 4, B, -4, A, 4, B, -2, C(\text{terminate})$$
   $$\text{Episode 2: } B, -2, A, 4, B, -4, C(\text{terminate}).$$

   From this we want to estimate $v_\pi(A)$ and $v_\pi(B)$ ($v_\pi(C) = 0$ since it is a terminating state).

   What will $V(A)$ and $V(B)$ be after the two episodes

   (a) if we use first-visit Monte-Carlo?

   (b) if we use every-visit Monte-Carlo?

2. We study an environment with three states, $\mathcal{S} = \{A, B, C\}$, where C is a terminating state. A policy π is used to observe the following:

   $$S_0 = B, R_1 = -2, S_1 = A, R_2 = 4, S_2 = B.$$

   The discount rate is $\gamma = 1$.

   Initialization: $V(A) = V(B) = V(C) = 0$.

   We use TD(0) with constant step size $\alpha = 1$. What will $V(A)$ and $V(B)$ be after the updates?

3. The environment consists of three states $\mathcal{S} = \{\text{Room } 0, \text{Room } 1, \text{Room } 2\}$. Room 2 is a terminal state. The three rooms are in a corridor and the agent can take the action $\mathcal{A} = \{\text{Left}, \text{Right}\}$.

   Consider trying to learn a policy for this environment using Q-learning. We use the step size $\alpha = 1$ and the discount rate $\gamma = 1$.

   Initialization: $Q(s, a) = 0$ for all s and a except $Q(\text{Room } 1, \text{Right}) = 10$.

   (a) We start in $S = \text{Room } 0$ and choose action $A = \text{Right}$. The agent moves to $S' = \text{Room } 1$ and gets reward $R = -1$.
   The Q-values are updated. What is $Q(s, a)$ for all pairs now?

   (b) We continue from part (a). We are now in $S = \text{Room } 1$ and take action $A = \text{Left}$. The agent moves to $S' = \text{Room } 0$ and gets reward $R = -1$.
   The Q-values are updated. What is $Q(s, a)$ for all pairs now?

   (c) After the two steps above, what is the greedy policy respect to Q?

4. Use Q-learning to find the optimal policy for the `Taxi-v3` environment. This is an undiscounted problem, i.e. $\gamma = 1$. You can use $\alpha = 0.1$, $\varepsilon = 0.1$ and train on at least 10 000 episodes.

   Doing like this you should get an estimated Q-function such that (at least in most states) the greedy policy w.r.t Q is optimal. In the quizz the question will be e.g.:

   "Give the optimal action in state $s = 410$" for a few different states. So be sure that you have code ready to answer these types of questions. Since there is a risk that the Q-learning will find a policy that is not optimal in every possible state, you pass this part even if you only give the correct answer in 80% of the states asked for.

   Remember: When you have finished training your policy, you should use the *greedy* policy to answer the questions. If you use a ε-greedy with $\varepsilon > 0$ there is a chance that you return an action that is not greedy w.r.t Q.

   Tips: When you are done training your agent, it is also fun to use `test_policy` from Tinkering Notebook 3 to see your agent in action. This can also give you a feeling for if the agent seems to behave in an optimal way.