Assignment 1 - MDPs and Dynamic Programming

Reinforcement Learning, spring 2025

Before you start with the quizz that corresponds to this assignment, it is a good idea to prepare by solving the problems in this pdf.

1. Solve Exercise 3.4 in the textbook (page 53).

2. Consider the `GridWorld-v0` environment studied in Tinkering Notebook 2 with discount rate $\gamma = 1$. The environment is also described in Example 4.1 in the textbook.

   Let $\pi(a|s)$ be a uniformly random policy (in all states all actions have the same probability). The state-value function $v_\pi(s)$ for this policy is given in Figure 4.1 (lower left) on page 77 of the textbook. Given a state s and action a, make sure that you understand how to compute $q_\pi(s,a)$ for this environment.

   Note: For this question you do not need to write any code since $v_\pi(s)$ is given in Figure 4.1. You are recommended to do this by hand, as it is also a way to train for the exam!

   In the quizz you will be given some s and a, and then have to compute $q_\pi(s,a)$.

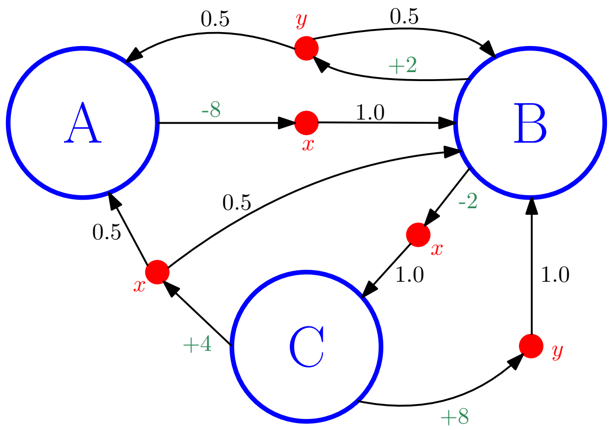
   Hint: One way to check that you are doing your computations correctly is as follows. Take e.g. state $s = 10$ and compute $q_\pi(10,a)$ for all actions (left, down, right, up). In the last row of Figure 4.1 you can see the greedy policy w.r.t to $v_\pi(s)$. This is just

   $$\pi'(s) = \arg\max_a q_\pi(s,a).$$

   You can now check that you in e.g. state $s = 10$ maximize $q_\pi(10,a)$ with either action down or right.

   Also, you can check that you get $q_\pi(1, \text{down}) = -19$.

3. In this problem we consider the MDP shown in Figure 3.1. It has three state, $\mathcal{S} = \{A, B, C\}$. In state B and C there are two possible actions called x and y. In state A only the action x is available. The discount rate is $\gamma = 0.5$, and we consider a uniform random $\pi(a|s)$ that in in each state picks between all possible actions with equal probability.



Figur 3.1: An MDP

   (a) It can be shown that $v_\pi(B) = 0.356$. What is $v_\pi(A)$ and $v_\pi(C)$?

   (b) Given $v_\pi(s)$ from part (a), find the policy that is greedy with respect to v_π.

(c) Assume that we start with an initial value function $v_1(A) = v_1(B) = v_1(C) = 2$. Perform one iteration with synchronous policy evaluation (do *not* use the in-place version!). What will $v_2(s)$ be?

4. Consider the `FrozenLake8x8-v1` environment. It is similar to the `FrozenLake-v1` that was studied in Tinkering Notebook 2, but it consist of an 8×8 grid and thus have 64 states.

   Write a code that find an optimal policy $\pi_*(s)$ and the corresponding value function $v_*(s)$.

   In the quizz on you will be asked for example "Which of these are optimal actions in state $s = 26$?" or "What is $v_*(26)$?". So make sure that you can easily run code that can answer these types of questions for different states.

   Hint: You can check that your code seems to be working by ensuring that you get to correct answer to the following:

   - For the optimal policy $v_*(26) = 0.80$ (rounded to two decimals).
   - In $s = 26$ the optimal action is 0 (left).