CSE 150. Assignment 7 *Spring 2018*

Out: *Tue May 29*
Due: *Thu Jun 07*
Reading: *Sutton & Barto*, Chapters 1-4.

7.1 CAPE Survey

You should have received an email from CAPE asking you to evaluate this course. **Please complete the online survey if you have not already done so.** Your answers will affect future offerings of this course.

7.2 Two-state MDP

Consider the Markov decision process (MDP) with two states $s \in \{0, 1\}$, two actions $a \in \{\downarrow, \uparrow\}$, discount factor $\gamma = \frac{1}{2}$, and rewards and transition matrices as shown below:

s	$R(s)$
0	-1
1	2

s	s'	$P(s'\|s, a = \downarrow)$
0	0	$\frac{3}{4}$
0	1	$\frac{1}{4}$
1	0	$\frac{1}{4}$
1	1	$\frac{3}{4}$

s	s'	$P(s'\|s, a = \uparrow)$
0	0	$\frac{1}{2}$
0	1	$\frac{1}{2}$
1	0	$\frac{1}{2}$
1	1	$\frac{1}{2}$

(a) **Policy evaluation**

Consider the policy π that chooses the action shown in each state. For this policy, solve the linear system of Bellman equations (by hand) to compute the state-value function $V^\pi(s)$ for $s \in \{0, 1\}$. Your answers should complete the following table. (*Hint:* the missing entries are whole numbers.) **Show your work for full credit.**

s	$\pi(s)$	$V^\pi(s)$
0	$\downarrow$	
1	$\downarrow$	

$V^\pi_0 = R(0) + \frac{1}{2}[P(0|0,\downarrow)V^\pi(0) + P(1|0,\downarrow)V^\pi(1)) = -1 + \frac{1}{2}(\frac{3}{4}V^\pi(0) + \frac{1}{4}V^\pi(1))$

(b) **Policy improvement**

Compute the greedy policy $\pi'(s)$ with respect to the state-value function $V^\pi(s)$ from part (a). Your answers should complete the following table. **Show your work for full credit.**

s	$\pi(s)$	$\pi'(s)$
0	$\downarrow$	
1	$\downarrow$	

use max

$Q^\pi(s=0, a=\downarrow) = V^\pi(0)$
$Q^\pi(s=1, a=\downarrow) = V^\pi(1)$
$Q^\pi(s=0, a=\uparrow) = R(0) + \gamma[P(0|0,\uparrow)V^\pi(0) + P(1|0,\uparrow)V^\pi(1)]$
$Q^\pi(s=0, a=\uparrow)$:

7.3 Three-state MDP

Consider the Markov decision process (MDP) with three states $s \in \{1, 2, 3\}$, two actions $a \in \{\uparrow, \downarrow\}$, discount factor $\gamma = \frac{2}{3}$, and rewards and transition matrices as shown below:

s	$R(s)$
1	-15
2	30
3	-25

s	s'	$P(s'\|s, a = \uparrow)$
1	1	$\frac{3}{4}$
1	2	$\frac{1}{4}$
1	3	0
2	1	$\frac{1}{2}$
2	2	$\frac{1}{2}$
2	3	0
3	1	0
3	2	$\frac{3}{4}$
3	3	$\frac{1}{4}$

s	s'	$P(s'\|s, a = \downarrow)$
1	1	$\frac{1}{4}$
1	2	$\frac{3}{4}$
1	3	0
2	1	0
2	2	$\frac{1}{2}$
2	3	$\frac{1}{2}$
3	1	0
3	2	$\frac{1}{4}$
3	3	$\frac{3}{4}$

(a) **Policy evaluation**

Consider the policy π that chooses the action shown in each state. For this policy, solve the linear system of Bellman equations (by hand) to compute the state-value function $V^\pi(s)$ for $s \in \{1, 2, 3\}$. Your answers should complete the following table. (*Hint:* the missing entries are whole numbers.) **Show your work for full credit.**

$Q^\pi(1,\uparrow) = V^\pi(1) = R(1) + \gamma[P(1|1,\uparrow)V^\pi(1) + P(2|1,\uparrow)V^\pi(2) + P(3|1,\uparrow)V^\pi(3)]$

$V^\pi(2) = R(2)$

$Q^\pi(3,\downarrow) = V^\pi(3) = R(3) + \gamma[P(1|3,\downarrow)V^\pi(1) + P(2|3,\downarrow)V^\pi(2) + P(3|3,\downarrow)V^\pi(3)]$

s	$\pi(s)$	$V^\pi(s)$
1	$\uparrow$	
2	$\uparrow$	
3	$\downarrow$	

(b) **Policy improvement**

Compute the greedy policy $\pi'(s)$ with respect to the state-value function $V^\pi(s)$ from part (a). Your answers should complete the following table. **Show your work for full credit.**

s	$\pi(s)$	$\pi'(s)$
1	$\uparrow$	
2	$\uparrow$	
3	$\downarrow$	

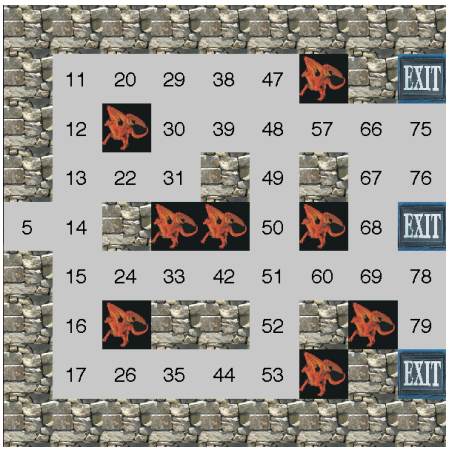
7.4 Value iteration

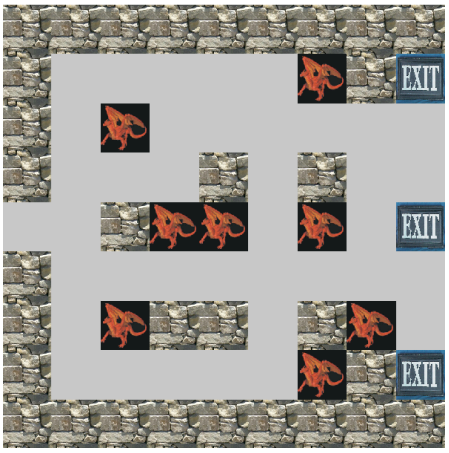
In this problem, you will use value iteration to find the optimal policy of the MDP demonstrated in class. This MDP has $|\mathcal{S}| = 81$ states and $|\mathcal{A}| = 4$ actions, and discount factor $\gamma = 0.99$. Download the ASCII files on the course web site that store the transition matrices and reward function for this MDP. The transition matrices are stored in a sparse format, listing only the row and column indices with non-zero values; if loaded correctly, the rows of these matrices should sum to one.

(a) Compute the optimal state value function $V^*(s)$ using the method of *value iteration*. Print out a list of the non-zero values of $V^*(s)$. Compare your answer to the numbered maze shown below. The correct value function will have positive values at all the numbered squares and negative values at the all squares with dragons.

(b) Compute the optimal policy $\pi^*(s)$ from your answer in part (a). Interpret the four actions in this MDP as moves to the WEST, NORTH, EAST, and SOUTH. Fill in the correspondingly numbered squares of the maze with arrows that point in the directions prescribed by the optimal policy. Turn in a copy of your solution for the optimal policy, as visualized in this way.

(c) **Turn in your source code along with your answers to the above questions.**





7.5 Approximate policy evaluation

Consider an MDP with transition matrices $P(s'|s,a)$ and reward function $R(s)$. In class, we showed that the state value function $V^\pi(s)$ for a fixed policy π satisfies the Bellman equation:

$$V^\pi(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi(s)) V^\pi(s').$$

The above defines a set of n linear equations, where n is the number of states in the MDP. If n is not too large, these equations can be solved *exactly* by standard methods.

If n is prohibitively large, however, the standard methods—which scale as $O(n^3)$—may not be applicable. In this case, one needs an approach that *approximately* solves these linear equations with whatever computational resources are available. One such method is to solve these equations *by iteration*. In this problem, you will analyze how quickly this approach converges to the exact solution. (*Hint:* the analysis is very similar to the proof of convergence for value iteration, though in this case it is even simpler.)

Let $V_k(s)$ denote our approximation to the state value function $V^\pi(s)$ at the kth iteration of the algorithm. Thus $V_0(s)$ is our initial approximation, and to start the algorithm, we adopt a very simple initialization, setting $V_0(s) = 0$ for all states s. Then the update rule at the k^{th} iteration, for all states $s \in \{1, 2, \ldots, n\}$, is given by:

$$V_{k+1}(s) \;\leftarrow\; R(s) + \gamma \sum_{s'} P(s'|s,\pi(s)) V_k(s').$$

Use this update rule to derive an upper bound on the error

$$\Delta_k = \max_s |V_k(s) - V^\pi(s)|$$

after k iterations of the update rule. Your result should show that the error Δ_k decays exponentially fast in the number of iterations, k, and hence that $\lim_{k\to\infty} V_k(s) = V^\pi(s)$ for all states s.