Q2. [24 pts] MDPs and RL: Mini-Grids

The following problems take place in various scenarios of the gridworld MDP (as in Project 3). In all cases, A is the start state and double-rectangle states are exit states. From an exit state, the only action available is *Exit*, which results in the listed reward and ends the game (by moving into a terminal state X, not shown).

From non-exit states, the agent can choose either *Left* or *Right* actions, which move the agent in the corresponding direction. There are no living rewards; the only non-zero rewards come from exiting the grid.

Throughout this problem, assume that value iteration begins with initial values $V_0(s) = 0$ for all states s.

First, consider the following mini-grid. For now, the discount is $\gamma = 1$ and legal movement actions will always succeed (and so the state transition function is deterministic).

+1	A			+10

(a) [1 pt] What is the optimal value $V^*(A)$?

10

Since the discount $\gamma = 1$ and there are no rewards for any action other than exiting, a policy that simply heads to the right exit state and exits will accrue reward 10. This is the optimal policy, since the only alternative reward if 1, and so the optimal value function has value 10.

(b) [1 pt] When running value iteration, remember that we start with $V_0(s) = 0$ for all s. What is the first iteration k for which $V_k(A)$ will be non-zero?

2

The first reward is accrued when the agent does the following actions (state transitons) in sequence: Left, Exit. Since two state transitions are necessary before any possible reward, two iterations are necessary for the value function to become non-zero.

(c) [1 pt] What will $V_k(A)$ be when it is first non-zero?

1

As explained above, the first non-zero value function value will come from exiting out of the left exit cell, which accrues reward 1.

(d) [1 pt] After how many iterations k will we have $V_k(A) = V^*(A)$? If they will never become equal, write *never*.

4

The value function will equal the optimal value function when it discovers this sequence of state transitions: Right, Right, Right, Exit. This will obviously happen in 4 iterations.

Now the situation is as before, but the discount γ is less than 1.

(e) [2 pts] If $\gamma = 0.5$, what is the optimal value $V^*(A)$?

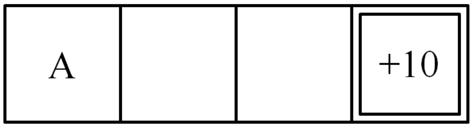
The optimal policy from A is Right, Right, Right, Exit. The rewards accrued by these state transitions are: 0, 0, 0, 10. The discount values are $\gamma^0, \gamma^1, \gamma^2, \gamma^3$, which is 1, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$. Therefore, $V^*(A) = 0 + 0 + 0 + \frac{10}{8}$.

(f) [2 pts] For what range of values γ of the discount will it be optimal to go *Right* from A? Remember that $0 \leq \gamma \leq 1$. Write *all* or *none* if all or no legal values of γ have this property.

The best reward accrued with the policy of going left is $\gamma^1 * 1$. The best reward accrued with the policy of going right is $\gamma^3 * 10$. We therefore have the inequality $10\gamma^3 \geq \gamma$, which simplifies to $\gamma \geq \sqrt{1/10}$. The final answer is $1/\sqrt{10} \leq \gamma \leq 1$

Let's kick it up a notch! The *Left* and *Right* movement actions are now stochastic and fail with probability f. When an action fails, the agent moves *up* or *down* with probability $f/2$ each. When there is no square to move *up* or *down* into (as in the one-dimensional case), the agent stays in place. The *Exit* action does not fail.

For the following mini-grid, the failure probability is $f = 0.5$. The discount is back to $\gamma = 1$.



A			+10

(g) [1 pt] What is the optimal value $V^*(A)$?

10. Same reasoning as for the previous problem.

(h) [1 pt] When running value iteration, what is the smallest value of k for which $V_k(A)$ will be non-zero?

4. Same reasoning as for the previous problem, but now the only reward-accruing sequence of actions is Right, Right, Right, Exit.

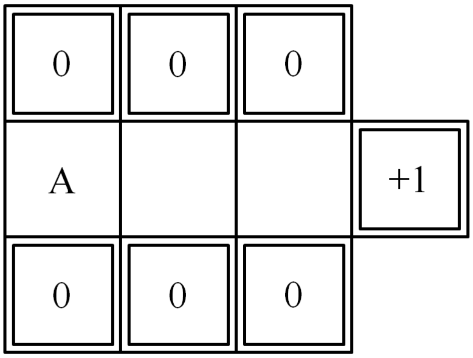
(i) [1 pt] What will $V_k(A)$ be when it is first non-zero?

10/8. Although $\gamma = 1$, the probability that the agent successfully completes the sequence of actions that leads to a reward at $k = 4$ (Right, Right, Right, Exit) is only $\frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} = \frac{1}{8}$, as at each non-Exit step it has only a $\frac{1}{2}$ probability of success.

(j) [1 pt] After how many iterations k will we have $V_k(A) = V^*(A)$? If they will never become equal, write *never*.

Never. There is always only a $\frac{1}{2}$ probability of success on any movement action, so while V_k will asymptotically approach V^*, it won't ever equal it. Consider the square right next to the exit, which we'll call C: $V_{k+1}(C) = \frac{1}{2}10 + \frac{1}{2}V_k(C)$.

Now consider the following mini-grid. Again, the failure probability is $f = 0.5$ and $\gamma = 1$. Remember that failure results in a shift *up* or *down*, and that the only action available from the double-walled exit states is *Exit*.



0	0	0	
A			+1
0	0	0	

(k) [1 pt] What is the optimal value $V^*(A)$?

1/8. Same reasoning as for the previous problem. Note that the exit node value is now only 1, not 10.

(l) [1 pt] When running value iteration, what is the smallest value of k for which $V_k(A)$ will be non-zero?

4

(m) [1 pt] What will $V_k(A)$ be when it is first non-zero?

1/8

(n) [1 pt] After how many iterations k will we have $V_k(A) = V^*(A)$? If they will never become equal, write *never*.

4. This problem is different from the previous one, in that a state transition never fails by looping to the same state. Here, a movement action may fail, but that always moves the agent into an absorbing state.

Finally, consider the following mini-grid (rewards shown on left, state names shown on right).

+4	A	+16

L	A	R

In this scenario, the discount is $\gamma = 1$. The failure probability is actually $f = 0$, but, now we do not actually know the details of the MDP, so we use reinforcement learning to compute various values. We observe the following transition sequence (recall that state X is the end-of-game absorbing state):

s	a	s'	r
A	*Right*	R	0
R	*Exit*	X	16
A	*Left*	L	0
L	*Exit*	X	4
A	*Right*	R	0
R	*Exit*	X	16
A	*Left*	L	0
L	*Exit*	X	4

(o) [2 pts] After this sequence of transitions, if we use a learning rate of $\alpha = 0.5$, what would Q-learning learn for the Q-value of (A, *Right*)? Remember that $Q(s, a)$ is initialized with 0 for all (s, a).

4. How do you get the max? Here's an example:

The sample sequence: $(A, Right, R, 0)$.

$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + \alpha(r + \gamma \max_{a'}(s', a'))$.

$Q(A, right) \leftarrow (1 - \alpha)Q(A, right) + \alpha(r + \gamma \max_{a'}(R, a'))$.

But since there is only one exit action from R, then:

$Q(A, right) \leftarrow (1 - \alpha)Q(A, right) + \alpha(r + \gamma Q(R, Exit))$.

Note that this MDP is very small – you will finish the game in two moves (assuming you have to move from A).

(p) [2 pts] If these transitions repeated many times and learning rates were appropriately small for convergence, what would Q-learning converge to for the Q-value of (A, *Right*)?

16. Q-learning converges to the optimal Q-value function, if the states are fully explored and the convergence rate is set correctly.