CS 471, Intro to Artificial Intelligence
Fall 2022
Jose Renteria

Written Assignment 1

Due Wednesday, November 9, 2022

Q1. Campus Lay

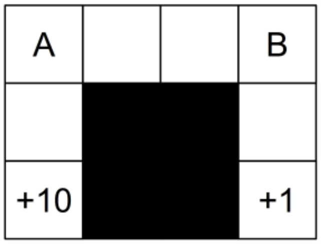


Figure 1: Grid world G

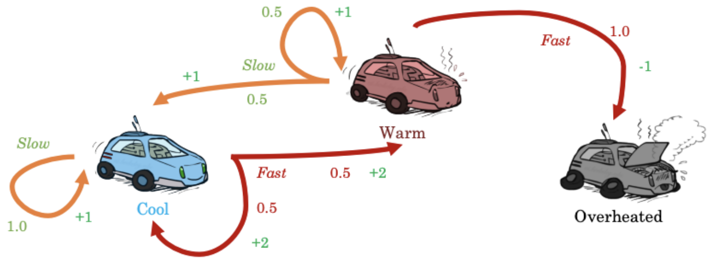
Q1.1

1. What is the smallest iteration k for which V_k (A) > 0? For this smallest iteration k, what is the value of V_k (A)?
   - The nearest reward is 10, which is 3 steps away, so our $k = 3$. Since our discount factor γ =1, there is no decay in the reward, so the propagated value is 10.
2. What is the smallest iteration k for which V_k (B) > 0? For this smallest iteration k, what is the value of V_k (B)?
   - The nearest reward is 1, which is 3 steps away, so our $k = 3$. Since our discount factor γ =1, there is no decay in the reward, so the propagated value is 1.
3. What is the smallest iteration k for which V_k(A) = V^*(A)? What is the value of V^*(A)?
   - Because our discount factor γ =1, the problem reduces to find the distance to the highest available reward, since there is no living reward. The highest possible reward is 10, which is at a distance of $k = 3$ steps.
4. What is the smallest iteration k for which V_k(B) = V^*(B)? What is the value of V^*(B)?
   - Because our discount factor $\gamma = 1$, the problem reduces to find the distance to the highest available reward, since there is no living reward. The highest available reward is 10, which is $k = 6$ steps away.

Q1.2 What is the smallest iteration k for which V_k(A) = V^* (A)? What is the value of V^* (A)?

- Since our discount factor γ = 1, and the only available rewards are in exit states, the optimal policy will favor the exit state containing the best possible reward. This will guarantee success, yielding an optimal value of 10 in state A. However, since the transition is non-deterministic, it's not guaranteed that this reward will be collected in precisely 3 steps. It could be any number steps k from [3, ∞). The values will only converge after an infinite number of iterations. Thus, our $k = \infty$, and the value of V^* (A) = 10.

Q2. MDPs - Policy Iteration



Q2.1

We begin with an initial policy of *Always Slow:*

	Cool	Warm	Overheated
π_0	*slow*	*slow*	–

Because terminal states have no outgoing actions, the policy can't assign a value to one. Hence, we can disregard *overheated* state and just assign $\forall$ i, $V^{\pi i}(s) = 0$ for any terminal state *s.* The next step runs a round of policy evaluation on π_0:

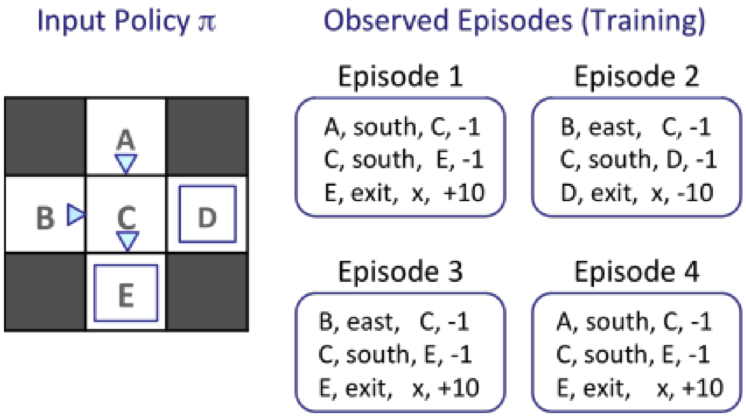
$V^{\pi 0}(cool) = 1 \ * [1 + 0.1 * V^{\pi 0}(cool)]$
$V^{\pi 0}(warm) = 0.1 * [1 + 0.1 * V^{\pi 0}(cool)] + 0.1 * [1 + 0.1 * V^{\pi 0}(warm)]$

We can now run policy extraction with these values:

$$\pi_1(cool) = \text{argmax}\{slow\text{:} \ 1 * [1 + 0.1 * 2], \ fast : 0.1 * [2 + 0.1 * 2] + 0.1 * [2 + 0.1 * 2]\}$$
$$= \text{argmax} \ \{slow : 1.2, fast\text{:} \ 0.44\}$$
$$= slow$$
$$\pi_1(warm) = \text{argmax}\{slow : 0.1 \cdot [1{+}0.1 \cdot 2] {+}0.1 \cdot [1{+}0.1 \cdot 2], \ fast : 1 \cdot [-10{+}0.1 \cdot 0]\}$$
$$= \text{argmax}\{slow\text{:} \ 0.24, fast\text{:} \ {-10}\}$$
$$= slow$$

Running policy iteration for a second round yields $\pi_2(cool) = slow$ and $\pi_2(warm) = slow$. Since this is the same policy as π_1, we can conclude $\pi_1 = \pi_2 = \pi^*$. This shows the power of policy iteration, since after only two iterations, we've arrived at the optimal policy for our racecar MDP. This is more than we say for value iteration on the same MDP, which was several more iterations from convergence after two updates were applied.

Q3. Model-Based RL



Input Policy π

Observed Episodes (Training)

Episode 1

A, south, C, -1
C, south, E, -1
E, exit, x, +10

Episode 2

B, east, C, -1
C, south, D, -1
D, exit, x, -10

Episode 3

B, east, C, -1
C, south, E, -1
E, exit, x, +10

Episode 4

A, south, C, -1
C, south, E, -1
E, exit, x, +10

What model would be learned from the above observed episodes (transition/reward functions)?

We will evaluate all four observed episodes:

- T(A, South, C) = 1
    - The action south is taken twice from state A, and both times produces a result in state C. 2/2 = 1
- T(B, East, C) = 1.
    - The action east is taken twice from state B, and both times results in state C. 2/2 = 1.
- T(C, South, E) = 0.75
    - The action south is taken four times from C, and results in E three out of four times. ¾ = 0.75
- T(C, South, D) = 0.25
    - The action south is taken four times from C, and results in D only one out of these four times. ¼ = 0.25

Q4. RL - Direct Evaluation

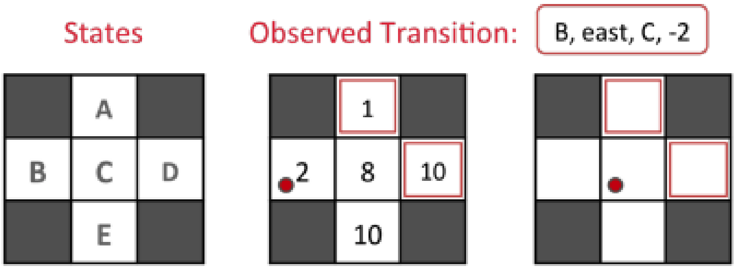
What are the estimates for $V^{\text{ˆ}\pi}$ (A), $V^{\text{ˆ}\pi}$ (B), $V^{\text{ˆ}\pi}$ (C), $V^{\text{ˆ}\pi}$ (D), and $V^{\text{ˆ}\pi}$ (E) as obtained by direct evaluation? Assume the discount factor $\gamma = 0.8$.

The estimated value of $V^{\text{ˆ}\pi}$ (*s*) is equivalent to the average value achieved beginning at state *s*.

$V^{\text{ˆ}\pi}$ (A)	Episodes 1 and 4 both start from A and result in utility 8.	$(8 + 8)/2 = 8$
$V^{\text{ˆ}\pi}$ (B)	Episodes 2 and 3 start from B, with episode 2 resulting in utility of -12, and episode 3 yielding utility 8.	$(8 - 12)/2 = -1$
$V^{\text{ˆ}\pi}$ (C)	C is visited in every episode, with remaining reward from C in episodes 1, 3, and 4 totaling 9. With remaining rewards in episode 2 totaling -11.	$(9 + 9 + 9 - 11) / 4 = 4$
$V^{\text{ˆ}\pi}$ (D)	State D only gets visited in episode 2, yielding a utility of -10.	-10
$V^{\text{ˆ}\pi}$ (E)	State E is visited in episodes 1, 3, and 4 and has remaining utility 10 in each state.	$(10 + 10 + 10) / 3 = 10$

Q5. RL - Temporal Difference Learning

Assuming γ = 0.8, α = 0.75, what are the value estimates of V^ π (A), V^ π (B), V^ π (C), V^ π (D), and V^ π (E) after the TD learning update?

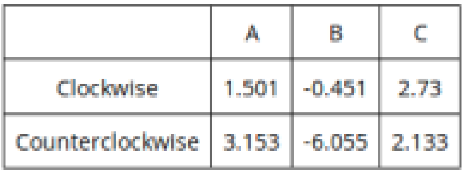


$V^{\pi}(s) \leftarrow (1 - \alpha)V^{\pi}(s) + \alpha[R(s, \pi(s), s') + \gamma V^{\pi}(s')]$

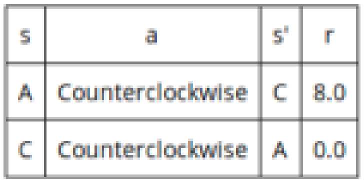
(1 - 0.75)(2)+0.75(-2+8)+(0.8)(8)

$V^{\wedge\pi}(A)$ = (1 - 0.75)(1)+0.75(1+8)+(0.8)(8) = 13.4
$V^{\wedge\pi}(B)$ = (1 - 0.75)(2)+0.75(-2+8)+(0.8)(8) = 11.4
$V^{\wedge\pi}(C)$ = (1 - 0.75)(8)+0.75(8+8)+(0.8)(8) = 20.4
$V^{\wedge\pi}(D)$ = (1 - 0.75)(10)+0.75(10+8)+(0.8)(8) = 22.4
$V^{\wedge\pi}(E)$ = (1 - 0.75)(10)+0.75(10+8)+(0.8)(8) = 22.4

Q6. RL - Model-Free Reinforcement Learning



	A	B	C
Clockwise	1.501	-0.451	2.73
Counterclockwise	3.153	-6.055	2.133



s	a	s'	r
A	Counterclockwise	C	8.0
C	Counterclockwise	A	0.0

Provide the Q-values for all pairs of (state, action) after both samples have been accounted for.

- Q(A, clockwise)
    - Ans: 1.501
- Q(A, counter-clockwise)
    - Ans: 6.259
- Q(B, clockwise)
    - Ans: -0.451
- Q(B, counter-clockwise
    - Ans: -6.055
- Q(C, clockwise)
    - Ans: 2.71
- Q(C, counter-clockwise)
    - Ans: 2.63125

Q7. RL - Feature-based Representation

Consider the following feature based representation of the Q-function: $Q(s, a) = w_1 f_1(s, a) + w_2 f_2(s, a)$ with:

- $f_1(s, a)$ = 1/ (Manhattan distance to nearest dot after having executed action a in state s)
- $f_2(s, a)$ =(Manhattan distance to nearest ghost after having executed action a in state s)

$w_1 = 3$, $w_2 = 8$

Q7.1

There are 2 actions available:

WEST

- $f_1(s, west) = 1$
- $f_2(s, west) = 3$
- $Q(s, west) = 3*1 + 8*3 = 27$

SOUTH

- $f_1(s, south) = 1$
- $f_2(s, south) = 1$
- $Q(s, south) = 3*1 + 8*1 = 11$

Q7.2

Provide the values of $Q(s', west)$ and $Q(s', east)$ What is the sample value?

$Q(s', west) = 11$

$Q(s', west) = 11$

Sample value = $[r + \gamma\ max_{a'} Q(s', a')] = 9 + (0.8)(11) = 17.8$

Q7.3

Difference = 17.8 - $Q(s, west)$ = 17.8 - 27

Now provide the update to the weights. Let $a = 0.5$

$w_1 = w_1 + \alpha \cdot difference \cdot f_1(s, West) = 3 + 0.5 * -9.2 * 1 = –1.6$

$w_2 = w_2 + \alpha \cdot difference \cdot f_2(s, West) = 8 + 0.5 * -9.2 * 3 = -5.8$