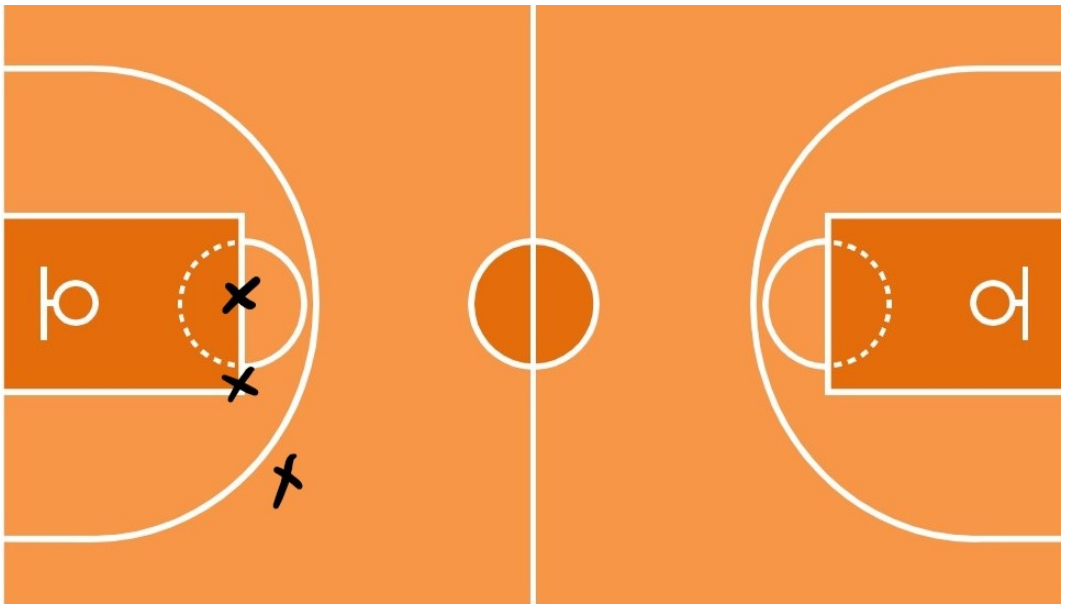
Exercises - Chapter 3

Victoria Nagorski

Exercise 3.1

Devise three example tasks of your own that fit into the MDP framework, identify for each its states, actions, and rewards. Make the three examples as different from each other as possible. The framework is abstract and flexible and can be applied in many different ways. Stretch its limits in some way in at least one of your examples.

1. Shooting a basketball (high-level)



Shooting a basketball can be grossly simplified to 3 different types of shots: free-throws (which require you to be fouled first, but it is a strategy to draw fouls to be put on the line), inside the three-point line, and then three-pointers. These three different shots bring different rewards. The three-pointer, being the hardest shot brings the greatest reward if made. However, if the shot is missed, the other team can retrieve the ball- giving a negative consequence for every missed shot.

$$\mathcal{A} = \left\{ \begin{array}{c} \text{Free-Throw} \\ \text{Inside the Three-Point Line} \\ \text{Three-Pointer} \end{array} \right\}, \mathcal{R} = \left\{ \begin{array}{c} \text{Free-Throw } (+1) \\ \text{Inside the Key } (+2) \\ \text{Three-Pointer } (+3) \\ \text{Missed Shots } (-1) \end{array} \right\},$$

$$\mathcal{S} = \left\{ \begin{array}{c} \text{Inside 3-point line} \\ \text{Outside 3-point line} \\ \text{Foul line} \end{array} \right\}$$

2. Taking a picture on a DSLR

$$\mathcal{A} = \left\{ \begin{matrix} \text{Underexpose } +1 \\ \text{Underexpose } +2 \\ \text{Meter} \\ \text{Overexpose } +1 \\ \text{Overexpose } +2 \end{matrix} \right\}, \mathcal{R} = \left\{ \begin{matrix} \text{Editable Photo}(+1) \\ \text{Perfect Exposure } (+2) \\ \text{Unusable Photo } (-2) \end{matrix} \right\},$$

$$\mathcal{S} = \left\{ \begin{matrix} \text{Cloudy} \\ \text{Bright (harsh light)} \\ \text{Golden Hours (sunrise and sunset)} \end{matrix} \right\}$$

3. Settler's of Catan

For those who have never played Settler's of Catan, it is a board game where multiple players have to interact with each-other to build roads, settlements, and cities to win the game. A dice is rolled and players with a matching hexagonal tile number get to collect resources for the tile. These resources can be used to build assets to gain reward for an eventual win. While the roads do not bring immediate reward (outside of longest road), they have to be built to maintain the ability to make more settlements and cities.

$$\mathcal{A} = \left\{ \begin{matrix} \text{Trade} \\ \text{Build Road} \\ \text{Build Settlement} \\ \text{Build City} \end{matrix} \right\}, \mathcal{R} = \left\{ \begin{matrix} (+1)/\text{Settlement} \\ (+2)/\text{City} \\ \text{"Longest Road"} \ (+2) \\ \text{Development Card (Victory Point)} \ (+1) \end{matrix} \right\},$$

$$\mathcal{S} = \left\{ \begin{matrix} \text{Turn} \\ \text{Not Turn} \end{matrix} \right\}$$

Exercise 3.2

Is the MDP framework adequate to usefully represent all goal-directed learning tasks? Can you think of any clear exceptions?

No, not all learning tasks can use the MDP framework adequately. For example: large states, scenarios that depend on past actions, scenario that cannot be broke into time intervals, and unknown rewards and environment.

Exercise 3.3

Consider the problem of driving. You could define the actions in terms of the accelerator, steering wheel, and break, that is, where your body meets the machine. Or you could define them farther out- say, where the rubber meets the road, considering your actions to be tire torques. Or you could define them father in- say, where your brain meets your body, the actions being muscle twitches to control your limbs. Or you could go to a really high level and say that your actions are your choices of where to drive. What is the right level, the right place to draw the line between agent and environment? On what basis is one location of the line to be preferred over another? Is there any fundamental reason for preferring one location over another, or is it a free choice?

It largely depends on what problem you are trying to solve.

- accelerator, steering wheel, and brake: Handling autonomous driving on the road with the other vehicles on the road.
- tire torques: You would not normally care about this on a day to day basis with driving. You might care about this when off-roading in the desert, and are worried about getting stuck in the sand. Or, driving on California roads in the rain after it has not rained in a long time.
- muscle twitches: This ones I am not sure its usefulness. Might be the same level of accelerator, etc. Might be more useful on training someone/something to drive.
- where to drive: Mail or gym? Both? Which path is the most time efficient for doing both tasks?

Exercise 3.4

Give a table analogous to that in Example 3.3, but for $p(s', r|s, a)$. *It should have columns for* s, a, s', r, *and* $p(s', r|s, a)$, *and a row for every 4-tuple for which* $p(s', r|s, a) > 0$.

s	s'	a	$p(s'\|s,a)$	$r(s,a,s')$
high	high	search	α	$r_{\texttt{search}}$
high	low	search	$1-\alpha$	$r_{\texttt{search}}$
low	high	search	$1-\beta$	-3
low	low	search	β	$r_{\texttt{search}}$
high	high	wait	1	$r_{\texttt{wait}}$
high	low	wait	0	$r_{\texttt{wait}}$
low	high	wait	0	$r_{\texttt{wait}}$
low	low	wait	1	$r_{\texttt{wait}}$
low	high	recharge	1	0
low	low	recharge	0	$0.$

Removing all rows where probability is 0 due to the problem statement. If $(s, a, s') \rightarrow (low, search, high)$, the robot lost charge and had to be recharged.

s	a	s'	r	$p(s', r\|s, a)$
high	search	high	r_{can}	$(0.5)(1-\alpha)$
high	search	high	0	$(0.5)\alpha$
high	search	low	r_{can}	$(0.5)(1-\alpha)$
high	search	low	0	$(0.5)\alpha$
low	search	high	r_{drain}	β
low	search	low	r_{can}	$(1-(0.5)\beta)$
low	search	low	0	$(1-(0.5)\beta)$
high	wait	high	r_{can}	$(1-\xi)$
high	wait	high	0	ξ
low	wait	low	r_{can}	$(1-(0.5)\gamma)$
low	wait	low	r_{drain}	$(1-(0.5)\gamma)$
low	wait	low	0	γ
low	recharge	high	0	1

Exercise 3.5

The equations in Section 3.1 are the continuing case and need to be modified (very slightly) to apply to episodic tasks. Show that you know the modifications needed by giving the modified version of (3.3)

Equation 3.3:

$$\sum_{s' \in \mathcal{S}, r \in \mathcal{R}} p(s', r|s, a)$$

For an episodic case, there are non-terminal states, $\mathcal{S}$, and then terminal states, $\mathcal{S}^+$. The variable T is also defined as the time of termination of the episodic episode. Rewrite:

$$\sum_{\substack{s' \in \mathcal{S};\, t<T \\ s' \in \mathcal{S}^+;\, t=T}} \sum_{r} p(s', r|s, a)$$

Exercise 3.6

Suppose you treated pole-balancing as an episodic task but also used discounting, with all rewards zero except for -1 upon failure. What then would the return be at each time? How does this return differ from that in the discounted, continuing formulation of this task?

Starting with the following equation:

$$G = R_{t+1} + \gamma R_{t+2} + ... + \gamma^{k-1} R_{t+k}, \text{ where } \gamma < 1$$

At each time interval that the agent manages to balance the pole, the reward would be 0. At failure, the reward is -1. Since γ is less than 1 and the exponent is growing as time goes on, the negative reward is decaying as time goes one. This means the longer the agent manages to balance the pole, the smaller the negative reward will become. Consequently, discounting in episodic tasking vs continuous would not differ. The only reason it might is if you have the agent go back to balancing the bar after failure. Episodic would restart at $t = 0$ while the continuous would not restart the time. As $t \to \infty$, the negative reward would go to 0.

Exercise 3.7

Image that you are designing a robot to run a maze. You decide to give it a reward of +1 for escaping from the maze and a reward of zero at all other times. The task seems to break down naturally into episodes- the successive runs through the maze- so you decide to treat it as an episodic task, where the goal is to maximize expected total reward (3.7). After running the learning agent for a while, you find that it is showing no improvement in escaping from the maze. What is going wrong? Have you effectively communicated to the agent what you want it to achieve?

The agent might not be improving because it has no incentive to do better. It only receives a reward other than 0 only when it escapes. Meaning, no matter how long it takes, it will always receive the same reward. A -1 reward for every second in the maze would encourage the agent to solve it faster.

Exercise 3.8

Suppose $\gamma = 0.5$ *and the following sequence of rewards is received* $R_1 = -1$, $R_2 = 2$, $R_3 = 6$, $R_4 = 3$, *and* $R_5 = 2$, *with* $T = 5$. *What are* G_0, G_1, ..., G_5? *Hint: Work backwards.*

$G_5 = 0,$ since after terminal time no additional reward is received

$$G_4 = 2$$

$$G_3 = 3 + 2(0.5) = 4$$

$$G_2 = 6 + 3(0.5) + 2(0.5)^2 = 8$$

$$G_1 = 2 + 6(0.5) + 3(0.5)^2 + 2(0.5)^3 = 6$$

$$G_0 = -1 + 2(0.5) + 6(0.5)^2 + 3(0.5)^3 + 2(0.5)^4 = 2.375$$

Exercise 3.9

Suppose $\gamma = 0.9$ and the reward sequence is $R_1 = 2$ followed by an infinite sequence of 7s. What are G_1 and G_0?

Utilizing the fact that the expected reward is a geometric series.
Solve G_0 first:

$$G_0 = R_1 + \gamma R_2 + \gamma^n R_{n+1}$$

$$G_0 = 2 + 7(\gamma + \gamma^n)$$

$$G_0 = 2 + 7\sum_{k=1}^{\infty} \gamma^k$$

Define:

$$S = 7\sum_{k=1}^{\infty} \gamma^k = 7(\gamma + \gamma^2 + ... + \gamma^n)$$

Multiply each side by γ:

$$\gamma S = 7(\gamma^2 + \gamma^3 + ... + \gamma^{n+1})$$

Subtract γS from S:

$$(1 - \gamma)S = 7(\gamma - \gamma^{n+1})$$

$$S = \frac{7(\gamma - \gamma^{n+1})}{(1 - \gamma)}$$

As $n \to \infty$ at $\gamma = 0.9$:

$$G_0 = 2 + \frac{7\gamma}{(1 - \gamma)} = 65$$

Following a similar procedure for G_1:

$$G_1 = 7\sum_{k=0}^{\infty} \gamma^k = 7(1 + \gamma + \gamma^2 + ... + \gamma^n)$$

$$\lim_{n\to\infty} G_1 = \frac{7(1 - \gamma^{n+1})}{(1 - \gamma)}$$

$$G_1 = \frac{7}{(1 - \gamma)} = 70$$

Exercise 3.10

Prove the second inequality in (3.10)

$$G_t = \sum_{k=0}^{k} = \frac{1}{1-\gamma}$$

Geometric Series:

$$G_t = \sum_{k=0}^{\infty} \gamma^k = 1 + \gamma + \gamma^2 + ... + \gamma^n$$

$$\gamma G_t = \sum_{k=0}^{\infty} \gamma^k = \gamma + \gamma^2 + \gamma^3 + ... + \gamma^{n+1}$$

$$(1-\gamma) G_t = \sum_{k=0}^{\infty} \gamma^k = 1 - \gamma^{n+1}$$

$$\lim_{n\to\infty} G_t = \frac{(1-\gamma^{n+1})}{(1-\gamma)} = \frac{1}{(1-\gamma)}, \text{ QED}$$

Exercise 3.11

If the current state is S_t, and actions are selected according to stochastic policy π, then what is the expectation of R_{t+1} in terms of π and the four-argument function of p(3.2)?

To fully describe the probability of the next state with $s = S_t$ and a given policy:

$$\pi(a = A_t | s = S_t) p(s' = S_{t+1}, r = R_{t+1} | s = S_t, a = A_t)$$

Therefore the following equation describes the expected reward with a given state and policy:

$$r(s,a) = \sum_{r \in \mathcal{R}} r \sum_{a \in \mathcal{A}} \pi(a|s) \sum_{s' \in \mathcal{S}} p(s', r | s, a)$$

Exercise 3.12

Give an equation for v_π in terms of q_π and π.

$$v_\pi(s) = \mathbb{E}_\pi \big[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s \big]$$

$$q_\pi(s) = \mathbb{E}_\pi \big[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s, A_t = a \big]$$

$$v_\pi(s) = \sum_a \pi(a|s) q_\pi(s,a)$$

Exercise 3.13

Give an equation for q_π in terms of v_π and the four-argument p

$$v_\pi(s) = \mathbb{E}_\pi\big[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s\big]$$

$$q_\pi(s) = \mathbb{E}_\pi\big[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s, A_t = a\big]$$

Stealing from equation 3.14 on the next page:

$$q_\pi(s) = \frac{v_\pi(s)}{\sum_a \pi(a|s)} = \frac{\cancel{\sum_a \pi(a|s)} \sum_{s',r} p(s', r|s, a)[r + \gamma v_\pi(s')]}{\cancel{\sum_a \pi(a|s)}}$$

$$q_\pi(s) = \sum_{s',r} p(s', r|s, a)[r + \gamma v_\pi(s')]$$

Exercise 3.14

The Bellman equation (3.14) must hold for each state for the value function v_π shown in Figure 3.2 (right) of Example 3.5. Show numerically that this equation hold for the center state, value at +0.7, with respect to its four neighboring states, valued at +2.3, +0.4, -0.4, and +0.7. (These numbers are accurate only to one decimal point.)

Equation 3.14:

$$v_\pi(s) = \underbrace{\sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)}_{\text{Valued at 0.25 each}}[r + \gamma v_\pi(s')]$$

The $\sum_a \pi(a|s)$ part of the equation represents the probability of choosing an action based on the current state. $\sum_{s',r} p(s', r|s, a)$ describes the probability of a future state and reward based on current action and state. The action is not given and based on the policy. This is why you need to multiply the policy against the $\sum_{s',r} p(s', r|s, a)$ term to get the total probability based on the current state. The problem statement gives that the s' states all have equal probability. That means $\pi(a|s) * p(s', r|s, a) = 0.25$.

$r = 0$ for all for s' states otuside of states A and B, and then negative rewards for running into the edges. $\gamma = 0.9$ as given by the problem statement. The equation can be expanded to:

$$v_\pi = (0.25)(0.9)[2.3 + 0.4 - 0.4 + 0.7] = 0.675 \approx +0.7$$

Exercise 3.15

In the gridworld example, rewards are positive for goals, negative for running into the edge of the world, and zero the rest of the time. Are the signs of these rewards important, or only the intervals between them? Prove, using (3.8), that adding a constant c to all rewards adds a constant, v_c, to the values of all states, and thus does not affect the relative values of any states under any policies. What is v_c in terms of c and γ.

Equation 3.8:

$$G_t \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + ... = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

The sign are important because negative rewards take away from any already earned the expected reward. If running into the edge of the world received 1 point, the agent would purposely run into the wall because the total expected reward continues to increase. (Give a kid 2 cookies. Positive behavior is +1 cookie. Bad behavior results in a cookie being taken away. If the kid does undesirable behavior and you reward them with a cookie, they are going to continue the behavior in hopes of getting more cookies.)

$$v_\pi \doteq \mathbb{E}_\pi[G_t | S_t = s]$$

$$G_{t,c} \doteq (R_t + c) + \gamma(R_{t+2} + c) + \gamma^2(R_{t+3} + c) + ... = \sum_{k=0}^{\infty} \gamma^k (R_{t+k+1} + c)$$

$$v_{\pi,c} = \mathbb{E}_\pi[(R_{t+1} + c) + \gamma G_{t,c+1} | S_t = s]$$

$$v_{\pi,c} = \sum_a \pi(a|s) \sum_{s',r} p(s', (r+c)|s, a) \big[(r+c) + \gamma \mathbb{E}_\pi[G_{t,c+1} | S_{t+1} = s']\big]$$

$$\Delta v_\pi = v_\pi - v_{\pi,c} = v_c = \sum_a \pi(a|s) \sum_{s',r} p(s', (r+c)|s, a) \big[c + \gamma \mathbb{E}_\pi[c + \gamma c + \sum_{k=2}^{\infty} \gamma^k c | S_{t+1} = s']\big]$$

Exercise 3.16

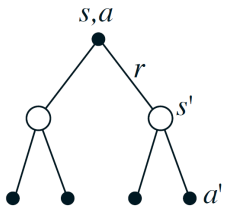
Now consider adding a constant c to all the rewards in an episodic task, such as maze running. Would this have any effect, or would it leave the task unchanged as in the continuing task above? Why or why not? Give an example.

Nothing would change because the difference between an episodic and continuing task is that the agent stops receiving a reward after time, T, and the existence of an additional state space, $\mathcal{S}^+$. The summation in the expectation term would not go to infinity but end at a finite number (the γ term is not needed). The state space $\mathcal{S}^+$ would also be updated. The only time it would matter is if the constant changes the sign of the reward.

In the maze running example, if each passing second was a reward of -5 and a constant of $c = +2$ was added, the sign of the reward would not change. The agent would not respond differently because the overall reward is still decreasing each second. However, if the reward was -1 for each passing second and 2 was added, the reward would be +1 for each passing second. The agent would have a reason to take its time.

Exercise 3.17

What is the Bellman equation for action values, that is, for q_π? It must give the action value $q_\pi(s, a)$ in terms of the action values, $q_\pi(s', a')$, of possible successors to the state-action pair (s,a). Hint: the backup diagram to the right correspond to this equation. Show the sequence of equations analogous to (3.14), but for action values.



$$q_\pi(s) = \mathbb{E}_\pi\big[G_t | S_t = s, A_t = a\big]$$

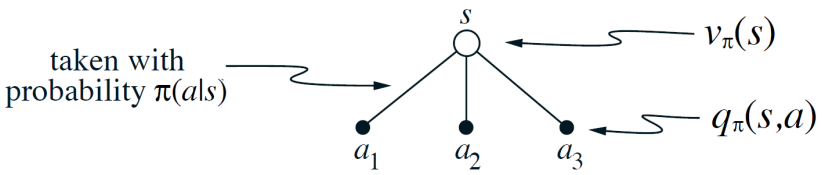
$$q_\pi(s) = \mathbb{E}_\pi\big[R_{t+1} + \gamma G_{t+1} | S_t = s, A_t = a\big]$$

$$q_\pi(s) = \sum_{s',r} p(s', r|s, a)\big[r + \gamma \mathbb{E}_\pi[G_{t+1}|S_{t+1} = s', A_{t+1} = a']\big]$$

$$q_\pi(s) = \sum_{s',r} p(s', r|s, a)\big[r + \gamma q_\pi(s')\big]$$

Exercise 3.18

The value of a state depends on the actions possible in that state and on how likely each action is to be taken under the current policy. We can think of this in terms of a small backup diagram rooted at the state and considering each possible action:



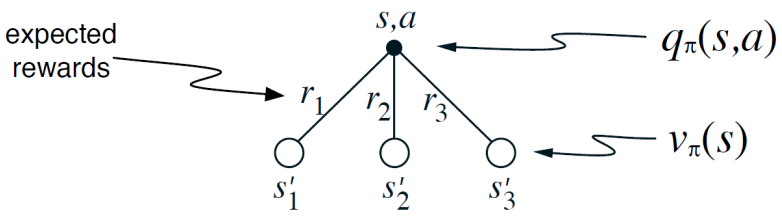
Give the equation corresponding to this intuition and diagram for the value at the root node, $v_\pi(s)$, in terms of the value at the expected leaf node, $q_\pi(s,a)$, given $S_t = s$. This equation should include an expectation conditioned on following the policy, π. Then give a second equation in which the expected value is written out explicitly in terms of $\pi(a|s)$ such that no expected value notation appears in the equation.

$$v_\pi(s) = \mathbb{E}_\pi[q_\pi(s,a)|S_t = s]$$

$$v_\pi(s) = \sum \pi(a|s) q_\pi(s,a)$$

Exercise 3.19

The value of an action, $q_\pi(s,a)$, depends on the expected next reward and the expected sum of the remaining rewards. Again we can think of this in terms of a small backup diagram, this one rooted at an action (state-action pair) and branching to the possible next states:



Give the equation corresponding to this intuition and diagram for the action value, $q_\pi(s,a)$, in terms of the expected next reward, R_{t+1}, and the expected next state value, $v_\pi(S_{t+1})$, given that $S_t = s$ and $A_t = a$. This equation should include an expectation but not one conditioned on following the policy. Then give a second equation, writing out the expected value explicitly in terms of $p(s',r|s,a)$ defined by (3.2), such that no expected value notation appears in the equation.

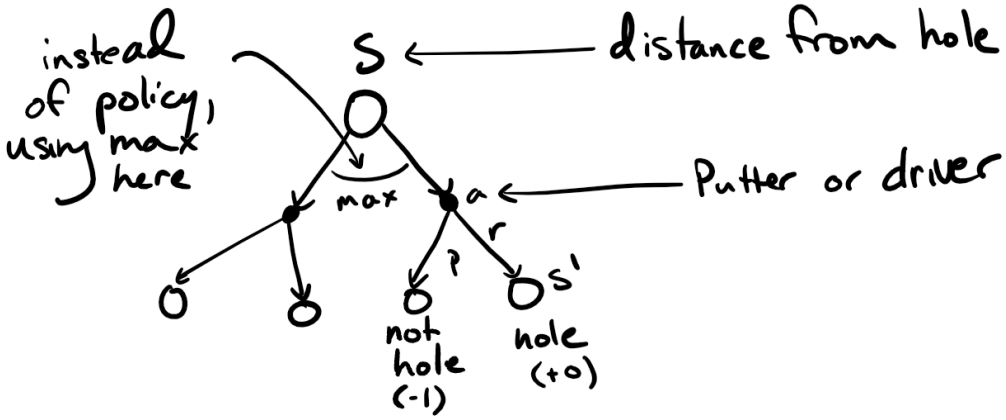
The expectation does not need to be conditioned on the policy because the policy defined a' in the q_π Bellman equation dependant on $q_\pi(s',a')$.

$$q_\pi(s,a) = \mathbb{E}[R_{t+1} + \gamma v_\pi(s')|S_t = s, A_t = a]$$

$$q_\pi(s,a) = \sum_{s',r} p(s',r|s,a)[r + \gamma v_\pi(s')]$$

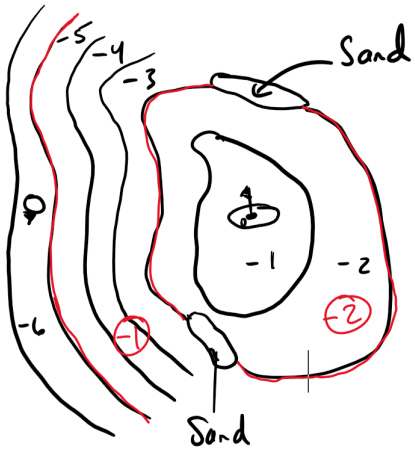
Exercise 3.20

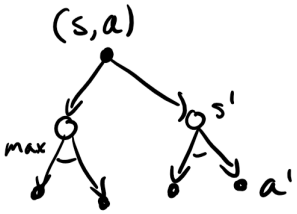
Draw or describe the optimal state-value function for the golf example.



Exercise 3.21

Draw or describe the contours of the optimal action-value function for putting, $q_\star(s,$ ***putter)****, for the golf example*



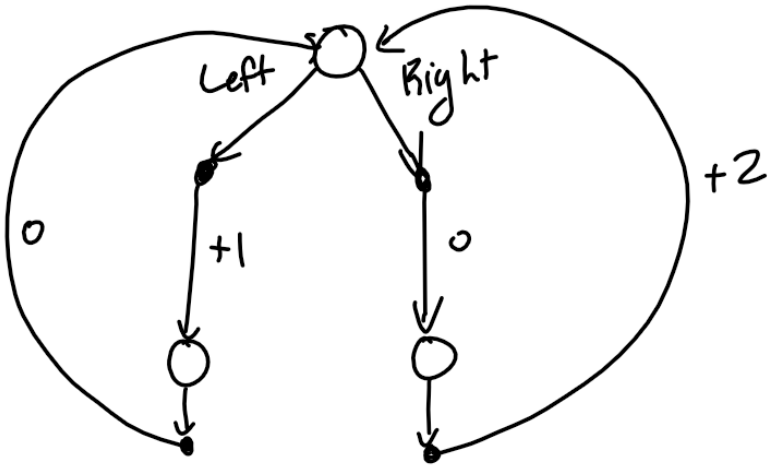


Unlike the v_{putt} case, a policy does not define the choice between putter and driver. In the $q_\star(s, putter)$ case, the next state is determined by the probability of the given state and action. There is a high probability that the agent will try to hit the hole (or at least move to a closer state). However, the next action will be chosen by what produces the best reward. In the case drawn above, the agent will choose the driver because it minimizes the negative reward.

The black represents the v_{putt} case and the red represents the field after the $q_\star(s, putter)$ step. The next step would be $q_\star(s', driver)$.

Exercise 3.22

Consider the continuing MDP shown on to the right. The only decision to be made is that in the top state, where two actions are available, ***left*** *and* ***right****. The numbers show the rewards that are received deterministically after each action. There are exactly two deterministic policies,* $\pi_{\textbf{left}}$ *and* $\pi_{\textbf{right}}$*. What policy is optimal if* $\gamma = 0$*? If* $\gamma = 0.9$*? If* $\gamma = 0.5$*?*



Utilizing the following equation:

$$G_t = \sum_{k=0}^{\infty} R_{t+1+k}\gamma^k = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + ... + \gamma^n R_{t+n-1}$$

$\gamma = 0 \rightarrow$ Go left because the first reward is the only one experienced.

$\gamma = 0.5 \rightarrow$ As shown below, the reward will be the same whether you go left or right.

$$G_{t,left} = \underbrace{[1+0]}_{=1} + \underbrace{[(0.5)^2+0]}_{=0.25} + \underbrace{[(0.5)^3+0]}_{=0.125} + ...$$

$$G_{t,right} = \underbrace{[0+2(0.5)]}_{=1} + \underbrace{[0+2(0.5)^3]}_{=0.25} + \underbrace{[0+2(0.5)^4]}_{=0.125} + ...$$

$\gamma = 0.9 \rightarrow$ Go right because later rewards on the right have more weight than the earlier rewards on the left. On the first round, left receives +1 while the right receives +1.8 after discount.

Exercise 3.23

Give the Bellman equation for $q_\star$ for the recycling robot.

$$q_\star = \sum_{s',r} p(s', r|s, a)[r + \gamma \max_{a'} q_\star(s', a')]$$

$$q_\star = \sum_{s'} p(s'|s, a)[r(s, a, s') + \gamma \max_{a'} q_\star(s', a')]$$

$$q_\star(h, s) = p(h|h, s)\left[r(h, s, h) + \gamma \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix}\right] + p(l|h, s)\left[r(h, s, l) + \gamma \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]$$

$$q_\star(h, s) = \alpha\left[r_s + \gamma \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix}\right] + (1 - \alpha)\left[r_s + \gamma \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]$$

$$\boxed{q_\star(h, s) = r_s + \gamma\left[\alpha \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix} (1 - \alpha) \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]}$$

$$q_\star(h, w) = p(h|h, w)\left[r(h, w, h) + \gamma \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix}\right] + p(l|h, w)\left[r(h, w, l) + \gamma \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]$$

$$q_\star(h, w) = 1\left[r_w + \gamma \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix}\right] + 0\left[r_w + \gamma \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]$$

$$\boxed{q_\star(h, w) = r_w + \gamma \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix}}$$

$$q_\star(l, s) = p(h|l, s)\left[r(l, s, h) + \gamma \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix}\right] + p(l|l, s)\left[r(l, s, l) + \gamma \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]$$

$$q_\star(l, s) = \beta\left[r_s + \gamma \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix}\right] + (1 - \beta)\left[r_s + \gamma \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]$$

$$\boxed{q_\star(l, s) = r_s + \gamma\left[\alpha \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix} (1 - \beta) \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]}$$

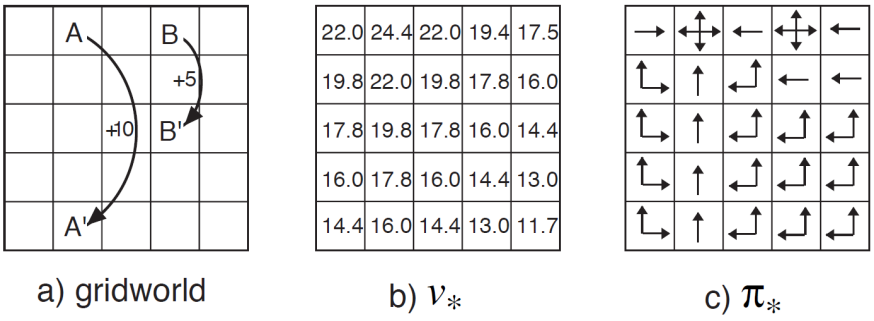
$$q_\star(l, w) = p(h|l, w)\left[r(l, w, h) + \gamma \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix}\right] + p(l|l, w)\left[r(l, w, l) + \gamma \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]$$

$$q_\star(l, w) = 0\left[r_w + \gamma \max \begin{Bmatrix} q_\star(h, s) \\ q_\star(h, w) \end{Bmatrix}\right] + 1\left[r_w + \gamma \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}\right]$$

$$\boxed{q_\star(l, w) = r_w + \gamma \max \begin{Bmatrix} q_\star(l, s) \\ q_\star(l, w) \end{Bmatrix}}$$

Exercise 3.24

Figure 3.5 gives the optimal value of the best state of the grid-world as 24.4, to one decimal place. Use your knowledge of the optimal policy and (3.8) to express this value symbolically, and then to compute it to three decimal places.



a) gridworld b) v_* c) π_*

Equation 3.8:

$$G_t = \sum_{k=0}^{\infty} R_{t+1+k}\gamma^k = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + ... + \gamma^n R_{t+n-1}$$

$$v_\star(s) \doteq \max_\pi v_\pi(s) = \max_a \mathbb{E}_{\pi\star}[G_t|S_t = s, A_t = a]$$

$$v_\star(s) = \max_a \sum_{s',r} p(s', r|s, a)[r + \gamma v_\star(s')]$$

According to $\pi_\star$, from state A, you could take any action and always receive +10 reward. This gives a 0.25 probability for each action since each action leads to the same outcome. After any action, you receive +10, and then are transported to state A'. A' has a $v_\star$ value of 16.0. $\gamma = 0.9$ in this example.

$$v_\star(s) = p(A', r|A, up/down/right/left)[r + \gamma v_\star(A')]$$

$$v_\star(s) = \underbrace{(1)}_{p}[\underbrace{10}_{r} + \underbrace{0.9}_{\gamma} * \underbrace{16.0}_{v_{\star(s')}}] = 24.4$$

Exercise 3.25

Give an equation for $v_\star$ in terms of $q_\star$.

$$q_\star(s, a) \doteq \max_\pi q_\pi(s, a)$$

$$v_\star(s) = \max_{a\in\mathcal{A}(s)} q_{\pi_\star(s,a)}$$

Exercise 3.26

Give an equation for $q_\star$ in terms of $v_\star$ and the four-argument p.

Equation 3.20:

$$q_\star(s,a) = \sum_{s',r} p(s',r|s,a)[r + \underbrace{\gamma \max_{a'} q_\star(s',a')}_{\text{Look at Exercise 3.25}}]$$

$$q_\star(s,a) = \sum_{s',r} p(s',r|s,a)[r + \gamma v_\star(s')]$$

Exercise 3.27

Give an equation for $\pi_\star$ in terms of $q_\star$.

Since optimal policies share the same optimal action-value function:

$$\pi_\star = \max_a q_\star(s,a)$$

Exercise 3.28

Give an equation for $\pi_\star$ in terms of $v_\star$ and the four-argument p.

Equation from exercise 3.26:

$$q_\star(s,a) = \sum_{s',r} p(s',r|s,a)[r + \gamma v_\star(s')]$$

Equation from exercise 3.27:

$$\pi_\star = \max_a q_\star(s,a)$$

Combine equations:

$$\pi_\star = \max_a \sum_{s',r} p(s',r|s,a)[r + \gamma v_\star(s')]$$

Exercise 3.29

Rewrite the four Bellman equations for the four value functions v_π, $v_\star$, q_π, and $q_\star$ in terms of the three argument function p (3.4) and the two-argument function r (3.5).

Using the following relationship:

$$r(s,a) = \sum_{s' \in S} p(s'|s,a) r(s,a,s') = \sum_r r \sum_{s'} p(s', r|s, a)$$

Equation 3.12 and 3.9 combined:

$$v_\pi(s) = \mathbb{E}_\pi\big[R_{t+1} + \gamma G_{t+1} | S_t = s\big]$$

$$v_\pi(s) = \underbrace{\mathbb{E}_\pi\big[R_{t+1}|S_t = s\big]}_{r(s,a)} + \gamma \underbrace{\mathbb{E}_\pi\big[G_{t+1}|S_t = s\big]}_{v_{\pi(s')}}$$

$$\boxed{v_\pi(s) = \sum_a \pi(a|s)\big[r(s,a) + \sum_{s'} p(s'|s,a)\gamma v_\pi(s')\big]}$$

Equation 3.18:

$$v_\star(s) = \max_a \mathbb{E}\big[R_{t+1} + \gamma v_\star(s_{t+1}) | S_t = s, A_t = a\big]$$

$$\boxed{v_\star(s) = \max_a \big[r(s,a) + \sum_{s'} p(s'|s,a)\gamma v_\star(s')\big]}$$

From exercise 3.17:

$$q_\pi(s) = \mathbb{E}_\pi\big[R_{t+1} + \gamma G_{t+1} | S_t = s, A_t = a\big]$$

$$\boxed{q_\pi(s) = r(s,a) + \sum_{s'} p(s'|s,a)\gamma q_\pi(s', a')}$$

From step before equation 3.20:

$$q_\pi(s,a) = \mathbb{E}\big[R_{t+1} + \gamma \max_{a'} G_{t+1} | S_t = s, A_t = a\big]$$

$$\boxed{q_\star(s,a) = r(s,a) + \sum_{s'} p(s'|s,a)\gamma \max_{a'} q_\star(s', a')}$$