Chapter 3: Finite Markov Decision Processes

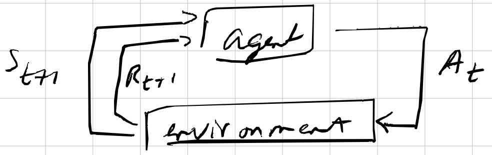
- formalize problem of finite MDPs, solve in rest of book
- evaluative & associative
- MDP's tackle delayed rewards and tradeoffs
+ now involve an s for state:
    - $q_*(s, a)$ → value of action given state
    - $v_*(s)$ → value of state given optimal action
- mathematical tractability vs application still at play, do our best.

The Agent Environment Interface

- agent → learner or decision maker
- environment → everything not agent that is interacted with

- env provides rewards that agent still wants to maximize
+ discrete time step interactions $t = 0, 1, 2 \ldots$
+ every step agent gets state representation $S_t \in \mathcal{S}$
+ and selects action $A_t \in \mathcal{A}(s)$
+ consequence of action is rewards one time step later $R_{t+1} \in \mathcal{R} \subset \mathbb{R}$ → new state S_{t+1}
* sequence looks like:

$$S_0 A_0 R_1 S_1 A_1 R_2 S_2 A_2 R_3 S_3 \ldots$$



- $\boxed{\text{finite}}$ MDPs mean all sets are discrete
+ well defined discrete probability dists:

$$p(s', r \mid s, a) \doteq \Pr\{S_t = s', R_t = r \mid S_{t-1} = s, A_{t-1} = a\}$$

probability of s' and r given s and a is the likelihood of s' and r given s and a were done in the previous timestep

+ dynamics function $p: \mathcal{S} \times \mathcal{R} \times \mathcal{S} \times \mathcal{A} = [0, 1]$

ordinary deterministic function

- Markov DP → probability p completely characterize environment dynamics

+ $p(S_t, R_t)$ ONLY ever depends on S_{t-1}, A_{t-1}

+ if state does not capture all prev info, not have Markov property

- can compute anything about env given dynamics function

+ state transition probabilities: $p: \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to [0,1]$

$$p(s' \mid s, a) \doteq \Pr\{S_t = s' \mid S_{t-1} = s, A_{t-1} = a\} = \sum_{r \in \mathcal{R}} p(s', r \mid s, a)$$

+ expected rewards for state action pairs $r: \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

$$r(s, a) \doteq \mathbb{E}[R_t \mid S_{t-1} = s, A_{t-1} = a] = \sum_{r \in \mathcal{R}} r \sum_{s' \in \mathcal{S}} r(s', r \mid s, a)$$

+ expected rewards of state-action-state triples
$r: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$

$$r(s, a, s') = \mathbb{E}[R_t \mid S_t = s', S_{t-1} = s, A_{t-1} = a] = \sum_{r \in \mathcal{R}} r \left(\frac{p(s', r \mid s a)}{p(s' \mid s, a)} \right)$$

- MDP framework abstract & flexible
+ time is arbitrary based on env
+ same with actions & states
- agent is often not robot or animal body
+ linkages, sensory organs, are part of env
+ boundary of agent is very tight
* e.g. reward is computed inside human's body, but we consider part of env, external to agent.
* Agent-environment boundary represents limits of absolute control not of knowledge

- all you need is 3 signals

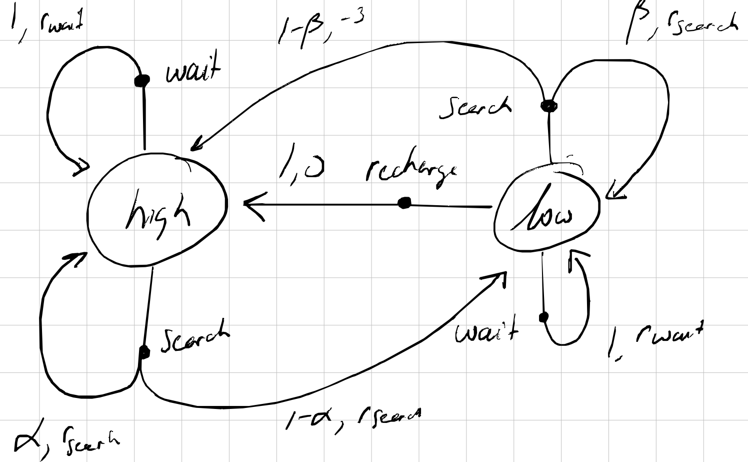
EXAMPLE: Mobile robot in search of cans
- 3 actions: search, wait, recharge.
- recharge pointless when high charge
+ $S = \{low, high\}$
+ $A(high) = \{search, wait\}$ $A(low) = \{search, wait, recharge\}$
- most of the time, reward = 0
+ positive iff can secured
+ large & negative if energy depleted
- when high, can search easily

+ if searching when high, probability staying high is α. going low $1-\alpha$

+ if searching when low, $pr(low) = \beta$. depletion $= 1-\beta$

- can collected, reward = 1
- rescued, reward = -3
- r_{search}, r_{wait} $(r_{search} > r_{wait})$ represents # of cans that will be collected for the action (aka expected reward)
- now have a finite MDP, all transition probs known

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



- in table, some pr(s, a, s') are 0 because they are impossible
- transition graph uses state nodes and action nodes
  - + state nodes for each state
  - + action node for each state-action pair

EXERCISE 3.4 Give a table but for $p(s', r \mid s, a)$

s	a	s'	r	$p(s',r \mid s,a)$
h	s	h	r_s	α
		l		$1-\alpha$
	w	h	r_w	1
l	s	h	-3	$1-\beta$
		l	r_s	β
	w		r_w	1
	r	h	0	1

Goals and Rewards

- reward hypothesis:

  "That all of what we mean by goals and purposes can be well thought of as the maximization of the expected value of the cumulative sum of a received scalar signal (called reward)"

- fairly flexible reward designation, but must indicate precisely our terminal desires
- what, not how

Returns and Episodes

- how to formalize long term desire?
+ maximizing expected return G_t
* G_t is some specific function of the reward sequence
* simplest case:

$$G_t \doteq R_{t+1} + R_{t+2} + R_{t+3} + \dots + R_T$$

- T is terminal, or final time step
+ T works for notions of episodes where there is a final timestep or break
* ends in a special terminal state
- known as episodic tasks if distinguishable

+ often separate nonterminal states and terminal states $\mathcal{S}$ vs $\mathcal{S}^+$
+ time of final step T is a random variable from ep to ep
- other hand, continuing tasks could have

∞ G. doesn't work

- introduce discounting to maximize discounted return

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

- $0 \leq \gamma \leq 1$ discount rate : present value of future rewards
  + "a reward received k-timesteps in the future is worth only γ^{k-1} times what it would be worth if it were received immediately"
  + $\gamma < 1$ sum as finite value if $\{R_k\}$ is bounded
  + $\gamma = 0$, myopic, maximize immediate rewards
  * goal would be choosing an A_t that maximizes R_{t+1}
- as γ approaches 1, return obj takes future rewards into account more strongly
- returns from each timestep matter to each other

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \ldots$$

$$R_{t+1} + \gamma (R_{t+2} + \gamma R_{t+3} + \ldots)$$

$$R_{t+1} + \gamma G_{t+1}$$

- works $\forall\, t < T$, even if termination occurs at $t+1$ iff $G_T = 0$
- Convergence occurrence $= 1$

  e.g. reward $= +1$ G_t becomes

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

EXERCISE 3.5: Modify

$$p(s', r \mid s, a) = Pr\{S_t = s', R_t = r \mid S_{t-1} = s, A_{t-1} = a\}$$

I didn't get it, so asked for help

Episodic tasks have a terminal state of which no transitions are possible

$$pr(s', r \mid s, a) =$$

$$\begin{cases} Pr\{S_t = s', R_t = r \mid S_{t-1} = s, A_{t-1} = a & \text{if } s \neq \text{terminal} \\ 0 & \text{if } s = \text{terminal} \end{cases}$$

Mathematical statement requires explicit statement, others certain algos will fail

EXAMPLE 3.4 Pole Balancing

apply forces to a cart to keep a pole balanced upright.

- if episodic, $r = +1$ for every timestep until failure
+ return = total timesteps of episode
+ successful balance = ∞
- if continuous, $r = -1$ on failure, 0 all other times
+ expected reward is then $G_T = -\gamma^{K-1}$ where K is timesteps before failure
- both maximize reward

EXERCISE 3.6: Pole balancing, episodic, but $r = -1$ on failure and 0 all else, with discounting

- normal equation $G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$

- episodic, fix'd final time T

$$G_t = \sum_{k=0}^{T-t-1} \gamma^{k} R_{t+k+1}$$

$R_{t+k+1} = 0$ for all non failure action

$R_T = -1$ for failure

$$G_t = -\gamma^{T-t-1}$$

Still maximizes reward, in search of a smaller negative non-zero number (or 0 if never fails till time T

EXERCISE 3.7: Robot in a maze. +1 for escape, 0 all else. Episodic. No learning occurring. Why?

$$G_t = \sum_{k=0}^{T-t-1} \gamma^{k} R_{t+k+1}$$

0 for all $k \neq T-t-1$

1 for $k = T-t-1$

$$= \gamma^{T-t-1} R_T = \gamma^{T-t-1} = \gamma^{K-1}$$

if using discounting, $\gamma < 1$ which means bigger K ($K = T - t$) results in smaller G_t : Reward does not work. ✓

EXERCISE 3.8 $\gamma = 0.5$ $R_1 = -1$, $R_2 = 2$, $R_3 = 6$ $R_4 = 3$, $R_5 = 2$ $T = 5$. What are $G_0, G_1, \dots G_5$

$G_5 = 0$

$G_4 = R_5 + (0.5)\, G_5 = 2$

$G_3 = R_4 + (0.5)\, G_4 = 4$

$G_2 = R_3 + 0.5\, G_3 = 8$

$G_1 = R_2 + 0.5\, G_2 = 6$

$G_0 = R_1 + 0.5\, G_1 = 2$

✓

EXERCISE 3.9 : $\gamma = 0.9$ $R_1 = 2$, $R_2 \dots \infty = 7$

$G_0, G_1 = ?$

$$G_1 = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} = 7 \sum_{k=0}^{\infty} \gamma^k$$

$$= \frac{7}{1-\gamma} = \frac{7}{0.1} = \boxed{70}$$

$$G_0 = R_1 + 0.9(G_1) = 2 + 0.9(70) = \boxed{65}$$ ✓

∞

EXERCISE 3.10: prove $G_t = \sum_{k=0} \gamma^k R_{t+k+1} = \frac{1}{1-\gamma}$ if R_{t+k+1} is nonzero and constant

$$G_t = \sum_{k=0}^{\infty} \gamma^k \Rightarrow \gamma^0 + \gamma^1 + \gamma^2 + \dots$$

$$G_t = 1 + \gamma(1 + \gamma^0 + \gamma^1 + \dots)$$

$$G_t = 1 + \gamma G_t$$

$$G_t(1-\gamma) = 1$$

$$G_t = \frac{1}{1-\gamma}$$

geometric series

$$\alpha r + \alpha r^2 + \alpha r^3 + \dots = \frac{\alpha}{1-r}, \quad |r| < 1$$

Unified Notation for Episodic & Continuing Tasks

- Capture both forms fluidly
- first, more precise episodic language
- $S_{t,i}$ = state at time t in ep i

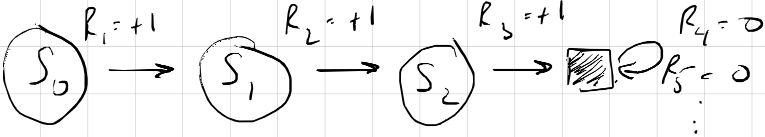
+ $A_{t,i}$ $R_{t,i}$, $\pi_{t,i}$ T_i, etc

* however, rarely do we refer to specific diff eps (1, 2, 3 ...). Either one or all

* hence, often drop i, implicitly understood

- how do we capture full return from an episode vs continuous

+ episode termination results in $\boxed{\text{absorbing state}}$ where infinite 0's are rewarded



- Same def for G_t now works, with discounting
- OR

$$G_t \doteq \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k$$

+ including $T = \infty$ or $\gamma = 1$, BUT NOT BOTH

Policies and Value Functions

- value functions estimate how good agent is doing, in a given state, or for a given action
- values functions per actions become policies
- $\boxed{\text{policy}}$ - mapping states to probabilities of each possible action

+ if agent following policy π at time t

$$\pi(a|s) = Pr\{A_t = a \mid S_t = s\}$$

EXERCISE 3.11: S_t current state stochastic policy π, what is $\mathbb{E}[R_{t+1}]$ in terms of π and $p(s',r|s,a)$

Needed help to unravel

$$1 \quad \mathbb{E}[R_{t+1}] = \sum_a \pi(a|s)\, \mathbb{E}[R_{t+1} \mid s,a]$$

$$2 \quad \mathbb{E}[R_{t+1} \mid s,a] = \sum_{s',r} r \cdot p(s',r \mid s,a)$$

from 1

$$= \sum_a \pi(a|s) \sum_{s',r} r \cdot p(s',r|s,a)$$

$$= \sum_a \sum_{s',r} r \cdot \pi(a|s) \cdot p(s',r|s,a)$$

in English: expected value of R_{t+1} requires

1) the probability of choosing a when in s according to the policy
2) the probability of transitioning from s' with reward r given s and a
3) the reward r itself

- $\boxed{\text{value function}}$ of state s under a policy π is $v_\pi(s)$ = expected return when starting in s and following π after.

+ MDPs get formal def:

$$v_\pi(s) \doteq \mathbb{E}_\pi\left[G_t \mid S_t = s\right]$$

$$= \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{k+t+1} \mid S_t = s\right], \forall s \in \mathcal{S}$$

+ $\mathbb{E}_\pi[\cdot]$ expected value of random var if agent follows policy π

* terminal state value always 0

+ v_π is $\boxed{\text{state-value function of policy } \pi}$

- now, value of a from s under π

$$q_\pi(s,a) \doteq \mathbb{E}_\pi\left[G_t \mid S_t = s, A_t = a\right]$$

$$= \mathbb{E}_\pi\left[\sum_{k=0} \gamma^k R_{t+k+1} \,\middle|\, S_t = s, A_t = a\right]$$

+ q_π $\boxed{\text{action-value function for policy } \pi}$

EXERCISE 3.12: express v_π in q_π & π

in English, express the value of the state s in terms of the value of each action possible in s and the probability of doing a in s

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

EXERCISE 3.13 express q_π in v_π and 4-arg p

In English, express the action value of a given state in terms of the overall state value and the transition dynamics that state possesses

① $q_\pi(s,a) = \mathbb{E}[G_t | S_t = s, A_t = a]$

Now the expected value for the return at timestep t given state s and action a is the sum of every reward and new state possible from the given state and action times the probability of that transition occurring

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

② $\sum_{s',r} p(s',r | s,a) \cdot G_t$

to relate G_t to s', recall that

$G_t = R_{t+1} + \gamma G_{t+1}$

from 2

③ $\sum_{s',r} p(s',r | s,a) \cdot (R_{t+1} + \gamma G_{t+1})$

$R_{t+1} = r$ since r is the t+1 reward

now $v_\pi(s') = \mathbb{E}_\pi[G_t | S_t = s']$

$s' = S_{t+1}$ G_{t+1} applies at S_{t+1}

so $v_\pi(s') = G_{t+1}$ because they are both the expected return value at s' following policy π

Back into 3

$$\sum_{s',r} p(s',r|s,a) \cdot (r + \gamma v_\pi(s'))$$

✓

- v_π and q_π can be estimated from exp
  + maintain average of rewards using policy π, as $t \to \infty$ avg will converge on v_π
  + do it for action in policy, get q_π
- Known as Monte Carlo methods
- fundamental to RL and DP is recursive relationships
  + "for any policy π and state s, the following consistency holds between s and its successor states:"

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

$$\mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1} \mid S_t = s]$$

$$= \sum_a \pi(a \mid s) \sum_{s'} \sum_r p(s', r \mid s, a) \cdot$$

$$[r + \gamma \mathbb{E}_\pi[G_{t+1} \mid S_{t+1} = s']]$$

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

$$\forall s \in \mathcal{S}$$

- Known as the $\boxed{\text{Bellman Equation for } v_\pi}$
- to be explicit:

$$\mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1} \mid S_t = s]$$

$$\underset{\textcircled{1}}{\mathbb{E}_\pi[R_{t+1} \mid S_t = s]} + \underset{\textcircled{2}}{\mathbb{E}_\pi[\gamma G_{t+1} \mid S_t = s]}$$ by

linearity of expectation

① ↘ Sum of all probability function of action × probability of transition with reward × reward

$$= \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a) \cdot r$$

② → G_{t+1} is expected return from S_{t+1} which is s'

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

$$= \gamma \sum_a \pi(a|s) \cdot \underbrace{\mathbb{E}_\pi [G_{t+1} | S_t = s, A_t = a]}_{③}$$

③ →

$$= \sum_{s',r} p(s',r|s,a) \cdot \mathbb{E} [G_{t+1} | S_{t+1} = s']$$

now $v_\pi(s') = \mathbb{E}_\pi [G_{t+1} | S_{t+1} = s']$

$$= \sum_{s',r} p(s',r|s,a) \cdot v_\pi(s')$$

into ②

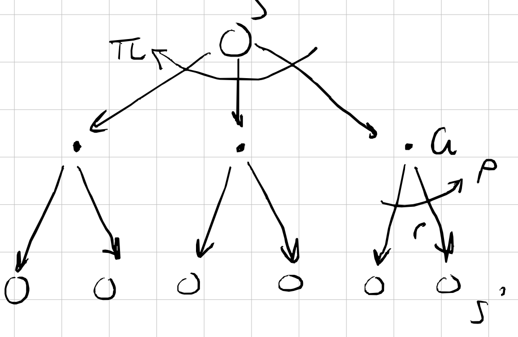
$$= \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a) \cdot \gamma \, v_\pi(s')$$

with ①

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

- In English, we compute the weighted probability $\pi(a|s)\, p(s',r|s,a)$ over the triple a, s', r, and sum over their multiplication by the bracketed term to get the expectation

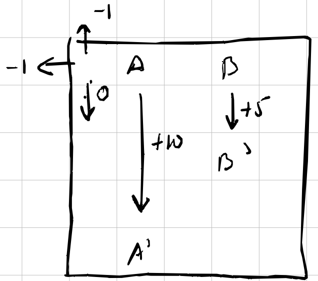
- represents the relation between value of state s and value of successor states



- backup diagram at heart of RL's backup

or update loops
+ transfer inf back to state (or state-action pair) from successor states or pairs

EXAMPLE 3.5: GRID WORLD



$\mathcal{A} = \{N, S, E, W\}$

policy $\pi(a|s) = 0.25 \quad \forall a \in \mathcal{A}, s \in \mathcal{S}$
(equiprobable)

$\gamma = 0.9$

$V_\pi =$

3.3	8.8	5.3
1.5	...	
0.1		0.4
⋮		
⋮	-1.3	

can be solved for using linear equations

$A \to A'$ expected return < 10 because likelihood of falling off the map

$B \to B' > 5$ for same reason

EXERCISE 3.14

S = center square

$V_\pi(S) = 0.7$

from Bellman Equation

$$0.7 = \sum_a 0.25 \sum_{s',r} 1 \cdot [0 + 0.9 V_\pi(s')]$$

$$= 0.25(0.9)[2.3 + 0.4 - 0.4 + 0.7]$$

$$= 0.25(0.9)(3)$$

$$= 0.675$$

$$\approx 0.7 \checkmark$$

EXERCISE 3.15 if $R_t = R_t + V_c$

then $G_t = R_{t+1} + v_c + \gamma(R_{t+2} + v_c) + \dots$

$$= \sum_{k=0} \gamma^k R_{t+k+1} + \gamma^k v_c$$

$$v_\pi(s) = \mathbb{E}[G_t \mid S_t = s]$$

$$= \mathbb{E}\left[\sum_{k=0} \gamma^k R_{t+k+1} + \gamma^k v_c \mid S_t = s\right]$$

linearity of exception

$$= \underset{\pi}{\mathbb{E}}\left[\sum_{k=0} \gamma^k R_{t+k+1} / S_t = s\right] + \underset{\pi}{\mathbb{E}}\left[\sum_{k=0} \gamma^k v_c / S_t = s\right]$$

① ②

① is just what $v_\pi(s)$ was before constant, denote $v_\pi'(s)$

so now $v_\pi(s) = v_\pi'(s) +$ ②

② does not depend on state

$$c \sum_{k=0}^{\infty} \gamma^k = \frac{c}{1-\gamma}$$ infinite geometric series

$$v_\pi(s) = v_\pi'(s) + \frac{c}{1-\gamma}$$

no longer state dependent, makes no diff.

EXERCISE 3.16 +c for episodic task.

~~It would be the same because the representation has not changed~~ WRONG

$$G_t = R_{t+1} + c + \dots + \gamma^{T-t-1}(R_T + c)$$

$$= (R_{t+1} + \gamma R_{t+2} + \dots + \gamma^{T-t-1} R_T) + (c + \gamma c + \dots + \gamma^{T-t-1} c)$$

$$= G_t' + c\sum_{k=0}^{T-t-1} \gamma^k$$

$$+ c\,\frac{1-\gamma^{T-t-1}}{1-\gamma}$$ finite geometric sum

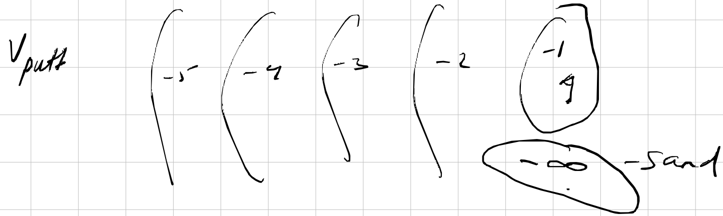
It does matter, as reward now is dependent on T

EXAMPLE 3.6: GOLF

- -1 reward for each stroke until in hole
- state is ball location
- value is amount of strokes, negative, to hole
- actions {
  - which club we select 1
  - where to aim 2
  - how to swing 3

+ assuming $A_{2,3}$ are given to us, only figure out A_1

* putter or driver



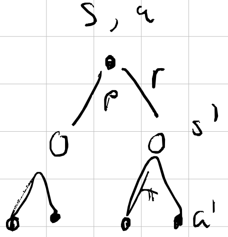
- 6 strokes from tee to hole with putting

EXERCISE 3.17 Bellman Equation for the example's action values q_π

$q_\pi(s,a)$ in terms $q_\pi(s',a')$

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

$$= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1} \mid S_t = s, A_t = a]$$



$$= \mathbb{E}_\pi[R_{t+1} \mid S_t = s\ A_t = a] + \mathbb{E}_\pi[\gamma G_{t+1} \mid S_t = s, A_t = a]$$

$$\sum_{s', a} p(s', r \mid s, a) \cdot r \qquad +$$

$$\sum_{s', a} p(s', r \mid s, a) \sum_{a'} \pi(a' \mid s') \ \gamma \ q_\pi(s', a')$$

BECAUSE the value of the action-value is given by $q_\pi(s', a')$

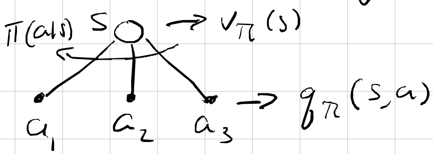
$$= \sum_{s', r'} p(s', r' \mid s, a) \quad \times$$

$$\left[r + \gamma \sum_{a'} \pi(a' \mid s') \ q_\pi(s', a') \right]$$

✓

EXERCISE 3.18:

express $v_\pi(s)$ as $q_\pi(a,s)$ for $S_t = s$



$$v_\pi(s) = \mathbb{E}_\pi[G_t \mid S_t = s]$$

$$v_\pi(s) = \mathbb{E}_\pi[q_\pi(s,a) \mid S_t = s]$$ ①

Simple, straightforward. Now finish

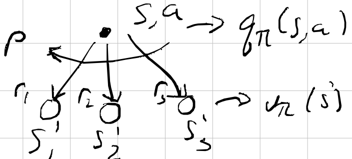
from ①

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

Extremely simple understanding that all value in v_π is captured in all q_π from v_π

✓

EXERCISE 3.19:



express $q_\pi(s,a)$ as expectation, then expand

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

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

G_{t+1} is $v_\pi(s')$ where $s' = S_{t+1}$. so

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

now expand $(R_{t+1} = r)$

$$= \sum_{s',r} p(s', r \mid s,a) \cdot r +$$
$$\sum_{s',r} p(s', r \mid s,a)\, \gamma\, v_\pi(s')$$

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

Optimal Policies and Optimal Value Functions

- solve RL - find policy that does the best in long run
+ Finite MDPs have precise optimal policy π
* $\pi \geq \pi'$ iff $V_\pi(s) \geq V_{\pi'}(s) \quad \forall s \in \mathcal{S}$
* In English, a policy π is better than or equal to policy π' if and only if the state value $V_\pi(s)$ is better than or equal to $V_{\pi'}(s)$ for all s in $\mathcal{S}$
- there is at least 1 policy that is better than all others.
+ called $\boxed{\text{optimal policy}}$
* all such policies are denoted π_*
* share V_* $\boxed{\text{optimal state-value function}}$

* $V_* = \max_\pi V_\pi(s) \quad \forall s \in \mathcal{S}$

- same for $\boxed{\text{optimal action-value function}}$

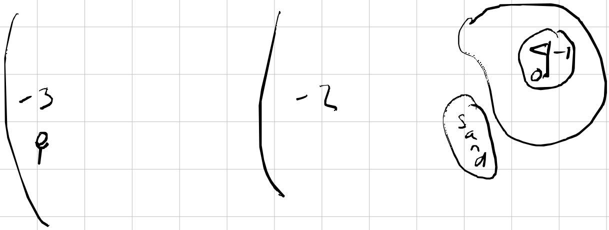
+ $q_* = \max_\pi q_\pi(s,a) \quad \forall s \in \mathcal{S}, a \in \mathcal{A}$

- written as each other

$$q_*(s,a) = \mathbb{E}\left[R_{t+1} + \gamma V_*(s) \mid S_t = s, A_t = a\right]$$

EXAMPLE 3.7: OPTIMAL VALUE IN GOLF

$q_*(s, \text{driver})$



- from tee, optimal value is 3 strokes
- write the $\boxed{\text{Bellman optimality equation}}$

$$\begin{aligned}
V_*(s) &= \max_{a \in \mathcal{A}(s)} q(s,a) \\
&= \max_a \mathbb{E}\left[G_t \mid S_t = s, A_t = a\right] \\
&= \max_a \mathbb{E}\left[R_{t+1} + \gamma V_*(S_{t+1}) \mid S_t = s, A_t = a\right] \\
&= \max_a \sum_{s',r} p(s', r \mid s, a)\left[r + \gamma V_*(s')\right]
\end{aligned}$$

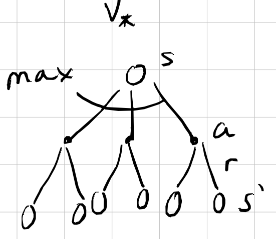
+ In English, "the value of a state under an optimal policy must equal the expected return of the best action from that state"

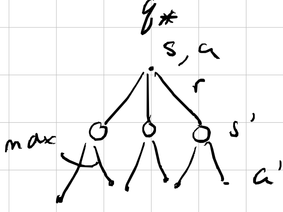
- for q_*, similar

$$q_*(s,a) = \mathbb{E}\left[R_{t+1} + \gamma \max_{a'} q_*(S_{t+1}, a) \mid S_t = s, A_t = a\right]$$

$$= \sum_{s',r} p(s',r \mid s,a)\left[r + \gamma \max_{a} q_*(s', a')\right]$$

- backup diagrams





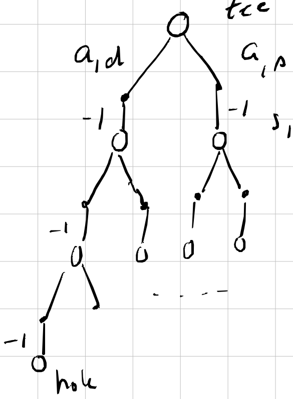
- optimality for finite MDPs can be solved
- n states, n equations for n unknowns
+ if dynamics p known, can solve for v_* using any nonlinear equation solver method
+ then solve related q_*

- know v_*, know optimal policy
+ use greedy, immediate evaluation, because v_* has already accounted for long term behavior
- knowing q_* means no search necessary.
+ given s, use whatever maximizes $q_*(s, a')$
* Caches results for immediate lookup
* cost of representing state-action pairs, you get instant lookup

- explicitly solving is cool, but rarely useful
+ requires
  1) dynamics are known
  2) sufficient compute resources
  3) states have Markov property
* "backgammon, 10^{20} states, huge compute
- must settle for approximations
+ e.g. expand tree up to x nodes deep, heuristic eval of leaf nodes
* e.g.: A* episodic pathfinding example
* e.g.: Dynamic Programming closer to approximation
- Many RL methods are trying to solve Bellman optimality equation in approximation

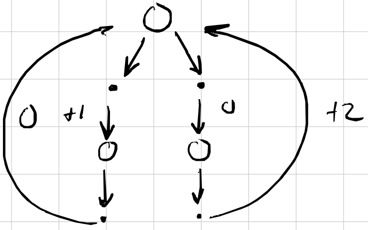
EXAMPLE 3.20 optimal $v_*(s)$ for golf

left = drive
right = putt



EXERCISE 3.2

two policies, π_L, π_r
if $\gamma = 0$, what is optimal



if $\gamma = 0$

$V_{\pi_L}(s) = 1 + 0(V_{\pi_L}(s'))$
$= 1$

$V_{\pi_r}(s) = 0 + 0(2) = 0$

$\boxed{\pi_L}$

if $\gamma = 0.9$

$V_{\pi_L}(s) = 1 + 0.9(v_{\pi_L}(s'))$

$s'' = s$

$V_{\pi_L}(s') = 0 + 0.9(V_{\pi_L}(s''))$

$V_{\pi_l}(s) = 1 + 0.9(0 + 0.9(V_{\pi_l}(s)))$

$1 + 0.81(V_{\pi_l}(s))$

$0.09\, V_{\pi_l}(s) = 1 \qquad V_{\pi_l}(s) = 11.\overline{1}$

$V_{\pi_r}(s) = 0 + 0.9(V_{\pi_r}(s'))$ from above

$= 0 + 0.9(2 + 0.9(V_{\pi_r}(s)))$

$= 0 + 1.8 + 0.81\, V_{\pi_r}(s)$

$V_{\pi_r}(s) = \frac{1.8}{0.09} = 20 \qquad \boxed{\pi_r}$

if $\gamma = 0.5$

$V_{\pi_l}(s) = \frac{1}{0.75} = \frac{4}{3} \qquad \boxed{\text{either}}$

$V_{\pi_r}(s) = \frac{1}{0.75} = \frac{4}{3} \qquad ✓$

I decided to skip the other exercise and leave them to the end

Optimality & Approximation

- use this notion to describe optimality, even if we only can approximate it
- computation & memory limitations

+ finite small cases, tabular techniques for representing each case

- however, often decisions that are suboptimal in near term do little to affect overall outcomes

+ learn to make good decisions in frequently encountered states than optimal in infrequent.

EXERCISE 3.25 v_* in q_*

$$v_*(s) = \max_a q_{\pi_*}(s,a)$$

EXERCISE 3.26 q_* in v_* and p

$$q_*(s,a) = \max_{s',r} v_{p_*}(s)$$

$$= \mathbb{E}_{p_*}[R_{t+1} + \gamma G_{t+1} \mid S_t = s, A_t = a]$$

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

intuition: $v_*(s)$ requires choosing the action a to maximize, "ending in q_*"

$q_*(s,a)$ is then simply taking action a, being at the mercy of p, and discounting the optimal reward of the next state.

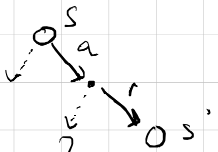
EXERCISE 3.27 π_* in q_*

$$\pi_*(a|s) = \max_a q_*(s,a)$$

EXERCISE 3.28 π_* in terms of v_*, p

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

EXERCISE 3.29 express v_π, v_*, P_π, P_* in terms of $p(s'|s,a)$, $r(s,a)$



$$p(s'|s,a)$$

$$\doteq Pr\{S_t = s' \mid S_{t-1} = s, A_{t-1} = a\}$$

$$= \sum_{r \in R} p(s', r \mid s, a)$$

$$r(s,a) \doteq \mathbb{E}[R_t \mid S_{t-1} = s, A_{t-1} = a]$$

$$= \sum_{s', r} p(s', r \mid s, a) \cdot r$$

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

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

①

↗

$$v_* = \max_{a}$$ ✓

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

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

$$q_\pi(s',a)$$

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

✓