Last Time:

- ☐ sequential decision-making
- ☐ MDPs

Lecture 3

HRI, FALL '25

Andrea Bajcsy

This Time:

- ☐ Bellman Eqn.
- ☐ Value Iteration + RL

Recap & Important MDP Quantities

- cumulative reward (R): sum of (discounted) rewards
  - ↳ the utility of one rollout / state-action traj.

  For traj. / rollout $s_0, a_0, s_1, a_1, \dots$

$$R(s_0, a_0, s_1, a_1, \dots) = \sum_{t=0}^{\infty} \gamma^t r(s_t) \quad \underline{\underline{\text{OR}}} \quad \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)$$

- optimal policy ($\pi^* : S \to A$): best action to take @ all states
  - ↳ maximizes expected discounted cumulative reward

$$\pi^* = \underset{\pi}{\operatorname{argmax}} \; \mathbb{E}_{\tau \sim p_\pi(\tau)} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right]$$

← expectation over all possible trajectories you generate via π

- optimal value function ($V^* : S \to \mathbb{R}$): expected sum of (discounted) rewards starting from s & acting optimally under π^*

$$V^*(s) := \mathbb{E}_{\tau \sim p_{\pi^*}(\tau)} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, \pi^*(s_t)) \;\middle|\; s_0 = s \right]$$

$$= \max_{\pi} \mathbb{E}_{\tau \sim p_\pi(\tau)} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \;\middle|\; s_0 = s \right]$$

- on-policy value function ($V^\pi : S \to \mathbb{R}$): expected sum of (discounted) rewards starting s when acting $\boxed{\text{under } \pi}$

$$V^\pi(s) := \mathbb{E}_{\tau \sim p_\pi(\tau)} \Bigg[\underbrace{\sum_{t=0}^{\infty} \gamma^t r(s_t, \pi(s_t))}_{\text{discounted sum of rewards}} \;\Bigg|\; s_0 = s \Bigg]$$

expectation over possible states you visit during rolling out π

Expand the expectation a bit more:

$$V^{\pi}(s) \triangleq \mathbb{E}_{\tau \sim p_{\pi}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^{t} r(s_t, \pi(s_t)) \,\middle|\, s_0 = s\right]$$

← defⁿ of value function

let's uncover a recursive structure

$$= \mathbb{E}_{\tau \sim p_{\pi}(\tau)}\left[\gamma^{0} r(s_0, \pi(s_0)) + \sum_{t=1}^{\infty} \gamma^{t} r(s_t, \pi(s_t)) \,\middle|\, s_0 = s\right]$$

$$= \mathbb{E}_{\tau \sim p_{\pi}(\tau)}\left[\gamma^{0} r(s_0, \pi(s_0)) + \gamma \sum_{t=1}^{\infty} \gamma^{t-1} r(s_t, \pi(s_t)) \,\middle|\, s_0 = s\right]$$

$$= \sum_{(s_0 = s, a_0, s_1, a_1, \ldots)} p_{\pi}(s_0, a_0, s_1, a_1, \ldots)\left[\gamma^{0} r(s_0, \pi(s_0)) + \gamma \sum_{t=1}^{\infty} \gamma^{t-1} r(s_t, \pi(s_t)) \,\middle|\, s_0 = s\right]$$

Know from Markov: $\sum_{\tau} P(s_0, a_0, s_1, a_1, \ldots) = \sum_{s_0} P(s_0) \sum_{s_1} P(s_1 | s_0, a_0) \sum_{s_2} P(s_2 | s_1, a_1) \cdots$

Plug in & rearrange:

$$= \sum_{s_1} P(s_1 | s_0, \pi(s_0))\left[r(s_0, \pi(s_0)) + \gamma \underbrace{\mathbb{E}_{\tau \sim p_{\pi}(\tau)}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r(s_t, \pi(s_t)) \,\middle|\, s_1\right]}_{:=\, V^{\pi}(s_1) \text{ by def}^{n}} \,\middle|\, s_0\right]$$

distribution over trajectories starting from s_1!

$$\boxed{V^{\pi}(s) \triangleq \sum_{s' \in S} P(s' | s, \pi(s)) \cdot \left[r(s, \pi(s)) + \gamma V^{\pi}(s')\right]}$$

↖ Bellman Equation ↗

We can also write / derive the <u>BELLMAN OPTIMALITY EQN for V^*</u>:

$$\boxed{V^{*}(s) \triangleq \max_{a \in \mathcal{A}} \sum_{s' \in S} P(s' | s, a) \cdot \left[r(s, a) + \gamma V^{*}(s')\right]}$$

Q-value function:

↳ like a value function but you're already committed to taking a particular action, a.

$$Q^*(s,a) \triangleq \underset{\substack{s_1 \sim P(\cdot|s_0,a) \\ \tau \sim P_{\pi^*}(\tau),\, t>0}}{\mathbb{E}}\left[r(s_0,a) + \sum_{t=1}^{\infty} \gamma^t r(s_t, \pi^*(s_t)) \,\Big|\, s_0 = s, a_0 = a\right]$$

← take current action a now... but afterwards act optimally!

commit to initial action

There is a nice relationship btwn. V^* and Q^*:

$$V^*(s) = \max_{a \in A} Q^*(s,a)$$

$$Q^*(s,a) = \sum_{s' \in S} P(s'|s,a) \cdot \left[r(s,a) + \gamma V^*(s')\right]$$

Intuition for Q-value: how "good" is taking action a?

$\boxed{\text{Optimal policy}}$: $\pi^*(s) = \arg\max_{a \in A} Q^*(s,a)$

Solving MDPs

Bellman equations are useful b/c they help us solve MDPs!

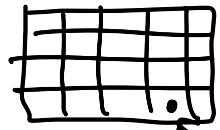
This comes from the recursive structure of Bellman eqn:

$$V(s) = \max_{a \in A}\left[r(s,a) + \gamma \cdot \sum_{s' \in S} P(s'|s,a)\, V(s')\right]$$

we need to have $V(s')$

$$V(s') = \max_{a \in A}\left[r(s',a) + \gamma \cdot \sum_{s'' \in S} P(s''|s',a)\, V(s'')\right]$$

VALUE ITERATION ALGORITHM:



$\leftarrow S$

grid world — $s \in S$

$V_0[s] \leftarrow 0, \forall s \in S$

for $k = 0, 1, 2, \dots$ until converged:

for each state $s \in S$:

$$V_{k+1}[s] \leftarrow \max_a \left[r(s,a) + \gamma \sum_{s' \in S} P(s' \mid s, a) \cdot V_k[s'] \right]$$

is another for loop over all next states

another for loop over all $a \in A$

return converged $V[s]$

Q-VALUE ITERATION:

$Q_0[s,a] \leftarrow 0 \quad \forall s \in S, a \in A$ // store current + updated estimate of Q

for $k = 0, 1, 2, \dots$ until converged:

for each state $s \in S$ and action $a \in A$:

$$Q_{k+1}[s,a] \leftarrow r[s,a] + \gamma \sum_{s' \in S} P(s' \mid s, a) \cdot \max_{a \in A} Q_k[s', a']$$

return converged $Q[s,a]$

Reinforcement learning (RL)

Q In MDPs, we assumed we know $P(s' \mid s, a)$ and rewards $r(s,a)$ .... but what if we don't?

In reality, it's hard to know P and r explicitly, so how do we obtain π^*?

A RL! It allows an agent to learn optimal policies by interacting with the environment.

⇒ agent tries out actions, observes rewards, and updates their policy to maximize rewards.