10-301/601: Introduction to Machine Learning Lecture 25: Value and Policy Iteration

Henry Chai

7/25/23

Front Matter

- Announcements
  - PA6 released 7/20, due 7/27 at 11:59 PM
    - Please be mindful of your grace day usage (see the course syllabus for the policy)
  - Wellness day on 7/31 (next Monday): no lecture or OH
- Recommended Readings
  - Mitchell, Chapter 13

Recall: Bellman Equations

- $V^{\pi}(s) = \mathbb{E}[$discounted total reward of starting in state s and executing policy π forever$]$

$$= \mathbb{E}[R(s_0, \pi(s_0)) + \gamma R(s_1, \pi(s_1)) + \gamma^2 R(s_2, \pi(s_2)) + \cdots \mid s_0 = s]$$

$$= R(s, \pi(s)) + \gamma \mathbb{E}[R(s_1, \pi(s_1)) + \gamma R(s_2, \pi(s_2)) + \ldots \mid s_0 = s]$$

$$= R(s, \pi(s)) + \gamma \sum_{s_1 \in \mathcal{S}} p(s_1 \mid s, \pi(s)) \big(R(s_1, \pi(s_1)) + \gamma \mathbb{E}[R(s_2, \pi(s_2)) + \cdots \mid s_1]\big)$$

$$V^{\pi}(s) = R(s, \pi(s)) + \gamma \sum_{s_1 \in \mathcal{S}} p(s_1 \mid s, \pi(s)) V^{\pi}(s_1)$$

Bellman equations

Optimality

- Optimal value function:

$$V^*(s) = \max_{a \in \mathcal{A}} R(s,a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s,a) V^*(s')$$

  - System of $|\mathcal{S}|$ equations and $|\mathcal{S}|$ variables

- Optimal policy:

$$\pi^*(s) = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \; \underbrace{R(s,a)}_{\text{Immediate reward}} + \underbrace{\gamma \sum_{s' \in \mathcal{S}} p(s' \mid s,a) V^*(s')}_{\text{(Discounted) Future reward}}$$

Fixed Point Iteration

- Iterative method for solving a system of equations
- Given some equations and initial values

$$x_1 = f_1(x_1, \dots, x_n)$$
$$\vdots$$
$$x_n = f_n(x_1, \dots, x_n)$$
$$x_1^{(0)}, \dots, x_n^{(0)}$$

- While not converged, do

$$x_1^{(t+1)} \leftarrow f_1\left(x_1^{(t)}, \dots, x_n^{(t)}\right)$$
$$\vdots$$
$$x_n^{(t+1)} \leftarrow f_n\left(x_1^{(t)}, \dots, x_n^{(t)}\right)$$

Fixed Point Iteration: Example

$\left(\frac{1}{3}\right)\left(-\frac{1}{2}\right) + \frac{1}{2} = -\frac{1}{6} + \frac{1}{2} = \frac{2}{6} = \frac{1}{3}$

$$x_1 = x_1 x_2 + \frac{1}{2}$$

$$x_2 = -\frac{3x_1}{2}$$

$-\frac{3\left(\frac{1}{3}\right)}{2} = -\frac{1}{2}$

$$x_1^{(0)} = x_2^{(0)} = 0$$

$$\hat{x}_1 = \frac{1}{3}, \hat{x}_2 = -\frac{1}{2}$$

t	$x_1^{(t)}$	$x_2^{(t)}$
0	0	0

Value Iteration

- Inputs: $R(s, a), p(s' \mid s, a)$
- Initialize $V^{(0)}(s) = 0 \ \forall \ s \in \mathcal{S}$ (or randomly) and set $t = 0$
- While not converged, do:
  - For $s \in \mathcal{S}$

$$V^{(t+1)}(s) \leftarrow \max_{a \in \mathcal{A}} R(s, a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s, a) V^{(t)}(s')$$

$O(|S||A||S|)$

$Q(s, a)$

  - $t = t + 1$
- For $s \in \mathcal{S}$

$$\pi^*(s) \leftarrow \underset{a \in \mathcal{A}}{\operatorname{argmax}} \ R(s, a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s, a) V^{(t)}(s')$$

- Return π^*

When survey is active, respond at **pollev.com/301601polls**

Lecture 25 Polls

0 done

0 underway

Start the presentation to see live content. For screen share software, share the entire screen. Get help at **pollev.com/app**

What is the runtime of one iteration of value iteration?

$O(|\mathcal{S}||\mathcal{A}|^2)$

$O(|\mathcal{S}|^2|\mathcal{A}|^2)$

Synchronous Value Iteration

- Inputs: $R(s, a)$, $p(s' \mid s, a)$
- Initialize $V^{(0)}(s) = 0 \; \forall \; s \in \mathcal{S}$ (or randomly) and set $t = 0$
- While not converged, do:
  - For $s \in \mathcal{S}$
    - For $a \in \mathcal{A}$

      $$Q(s, a) = R(s, a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s, a) V^{(t)}(s')$$

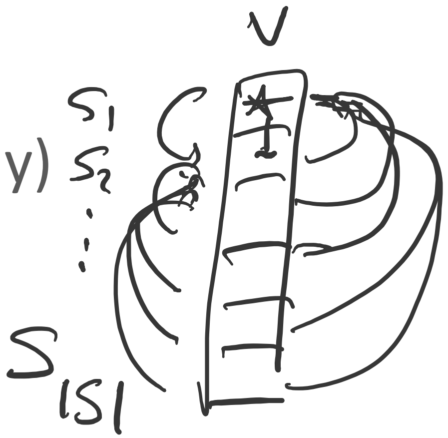
    - $V^{(t+1)}(s) \leftarrow \max_{a \in \mathcal{A}} Q(s, a)$
  - $t = t + 1$
- For $s \in \mathcal{S}$

  $$\pi^*(s) \leftarrow \operatorname{argmax}_{a \in \mathcal{A}} R(s, a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s, a) V^{(t)}(s')$$

- Return π^*

Asynchronous Value Iteration

- Inputs: $R(s, a)$, $p(s' \mid s, a)$
- Initialize $V^{(0)}(s) = 0 \; \forall \, s \in \mathcal{S}$ (or randomly)



- While not converged, do:
  - For $s \in \mathcal{S}$
    - For $a \in \mathcal{A}$

$$Q(s, a) = R(s, a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s, a) V(s')$$

    - $V(s) \leftarrow \max_{a \in \mathcal{A}} Q(s, a)$

- For $s \in \mathcal{S}$

$$\pi^*(s) \leftarrow \underset{a \in \mathcal{A}}{\operatorname{argmax}} \; R(s, a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s, a) V(s')$$

- Return π^*

Value Iteration Theory

- **Theorem 1**: Value function convergence

  V will converge to V^* if each state is "visited" infinitely often (Bertsekas, 1989)

- **Theorem 2**: Convergence criterion

  if $\max_{s \in \mathcal{S}} \left| V^{(t+1)}(s) - V^{(t)}(s) \right| < \epsilon$,

  then $\max_{s \in \mathcal{S}} \left| V^{(t+1)}(s) - V^{*}(s) \right| < \frac{2\epsilon\gamma}{1-\gamma}$ (Williams & Baird, 1993)

- **Theorem 3**: Policy convergence

  The "greedy" policy, $\pi(s) = \operatorname{argmax}_{a \in \mathcal{A}} Q(s,a)$, converges to the optimal π^* in a finite number of iterations, often before the value function has converged! (Bertsekas, 1987)

Policy Iteration

- Inputs: $R(s, a)$, $p(s' \mid s, a)$
- Initialize π randomly
- While not converged, do:
  - Solve the Bellman equations defined by policy π

$$V^{\pi}(s) = R(s, \pi(s)) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s, \pi(s)) V^{\pi}(s')$$

  - Update π

$$\pi(s) \leftarrow \underset{a \in \mathcal{A}}{\operatorname{argmax}}\ R(s, a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s, a) V^{\pi}(s')$$

- Return π

Policy Iteration Theory

- In policy iteration, the policy improves in each iteration.
- Given finite state and action spaces, there are finitely many possible policies

Which of the following is an upper bound on the number of possible policies?

- $|S| + |A|$
- $|S||A|$
- $|S|^{|A|}$

Policy Iteration Theory

- In policy iteration, the policy improves in each iteration.
- Given finite state and action spaces, there are finitely many possible policies
- Thus, the number of iterations needed to converge is bounded!
- Value iteration takes $O(|\mathcal{S}|^2|\mathcal{A}|)$ time / iteration
- Policy iteration takes $O(|\mathcal{S}|^2|\mathcal{A}| + |\mathcal{S}|^3)$ time / iteration
  - However, empirically policy iteration requires fewer iterations to converge

Two big Q's

1. What can we do if the reward and/or transition functions/distributions are unknown?

2. How can we handle infinite (or just very large) state/action spaces?

Key Takeaways

- If the reward and transition functions are known, we can solve for the optimal policy (and value function) using value or policy iteration
  - Both algorithms are instances of fixed point iteration and are guaranteed to converge (under some assumptions)