CSE 150A-250A AI: Probabilistic Models

Lecture 16

Fall 2025

Trevor Bonjour
Department of Computer Science and Engineering
University of California, San Diego

Slides adapted from previous versions of the course (Prof. Lawrence, Prof. Alvarado, Prof Berg-Kirkpatrick)

Agenda

Review

Value Functions

- State Value Function

$$
\begin{aligned}
V^{\pi}(s) &= \mathrm{E}^{\pi}\left[\sum_{t=0}^{\infty} \gamma^{t} R(s_t) \,\middle|\, s_0 = s\right] \\
&= R(s) + \gamma \sum_{s'} P(s' | s, \pi(s))\, V^{\pi}(s')
\end{aligned}
$$

- Action Value Function

$$
\begin{aligned}
Q^{\pi}(s, a) &= \mathrm{E}^{\pi}\left[\sum_{t=0}^{\infty} \gamma^{t} R(s_t) \,\middle|\, s_0 = s, a_0 = a\right] \\
&= R(s) + \gamma \sum_{s'} P(s' | s, a)\, V^{\pi}(s')
\end{aligned}
$$

Optimality

- **Goal**

  Find the optimal policy given the environment that the agent is in.

- **Planning**

  If reward function and transition probabilities are known.

- **Reinforcement Learning**
  If reward function and transition probabilities are unknown.

Optimality

There exists **at most** one policy π^* such that $V^{\pi^*}(s) \geq V^{\pi}(s)$ for all policies π and states s of the MDP.

True (A) or False (B)?

Optimal value functions, $Q^*(s, a)$ and $V^*(s)$ are unique and all optimal policies share the same value functions.

True (A) or False (B)?

Optimality

- **Theorem**

  There exists at least one policy π^* (and perhaps many) such that $V^{\pi^*}(s) \geq V^{\pi}(s)$ for all policies π and states s of the MDP.

- **Notation**

$$\begin{aligned} V^*(s) &= V^{\pi^*}(s) \\ Q^*(s,a) &= Q^{\pi^*}(s,a) \end{aligned}$$

  These optimal value functions are **unique**.
  (All optimal policies share the same value functions.)

Optimality

We can get the optimal policy π^* from the optimal value function $V^*(s)$ but not from the optimal action value function $Q^*(s, a)$.

True (A) or False (B)?

Relations at optimality

- From the optimal action value function:

$$V^*(s) = \max_a \left[Q^*(s,a)\right]$$
$$\pi^*(s) = \operatorname*{argmax}_a \left[Q^*(s,a)\right]$$

- From the optimal state value function:

$$Q^*(s,a) = R(s) + \gamma \sum_{s'} P(s'|s,a) V^*(s')$$

$$\pi^*(s) = \operatorname*{argmax}_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a) V^*(s')\right]$$

- Why are these relations useful?

  Sometimes it can be easier to estimate $Q^*(s,a)$ or $V^*(s)$ (which are continuous) than to learn $\pi^*(s)$ (which is discrete).

Planning in MDPs

Given a complete model of the agent and its environment as a Markov decision process, namely

$$\text{MDP} \;=\; \{\mathcal{S}, \mathcal{A}, P(s'|s,a), R(s), \gamma\},$$

how can we *efficiently* compute (i.e., in time *polynomial in the number of states*) any of the following:

1. an optimal policy $\pi^*(s)$?
2. the optimal state value function $V^*(s)$?
3. the optimal action value function $Q^*(s,a)$?

This is the problem of **planning** in MDPs.

Policy Based

Algorithms

1. **Policy evaluation**

   How to compute $V^{\pi}(s)$ for some fixed policy π?

2. **Policy improvement**

   How to compute a policy π' such that $V^{\pi'}(s) \geq V^{\pi}(s)$?

3. **Policy iteration**

   How to compute an optimal policy $\pi^{*}(s)$?

Policy evaluation

- How to compute the state value function?

$$V^\pi(s) = \mathrm{E}^\pi\left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \,\middle|\, s_0 = s\right]$$

R(s,s',a)

- Bellman equation:

$$V^\pi(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi(s)) V^\pi(s')$$

- **Solve linear system:** There are n equations for n unknowns (where $s = 1, 2, \ldots, n$).

Solving the linear system (con't)

- Solution

$$R = \left[I - \gamma P^{\pi}\right] V^{\pi} \implies V^{\pi} = \underbrace{\left(I - \gamma P^{\pi}\right)^{-1}}_{\text{matrix inverse}} R$$

- Complexity

  It takes $O(n^3)$ operations to solve this system of equations.

Policy improvement

- **Problem statement**

  Given a policy π and its state value function $V^{\pi}(s)$, how to compute a policy π' such that

  $$V^{\pi'}(s) \geq V^{\pi}(s) \quad \text{for all states } s?$$

- **Definition**

  Given the action value function $Q^{\pi}(s, a)$ for policy π, we define the **greedy policy** π' by

  $$\pi'(s) = \operatorname*{argmax}_a \left[Q^{\pi}(s, a) \right].$$

action

Greedy policies

- In terms of the state value function:

$$\begin{aligned}\pi'(s) &= \operatorname*{argmax}_a \left[Q^\pi(s,a) \right] \\ &= \operatorname*{argmax}_a \left[R(s) + \gamma \sum\nolimits_{s'} P(s'|s,a)\, V^\pi(s') \right] \\ &= \operatorname*{argmax}_a \left[\sum\nolimits_{s'} P(s'|s,a)\, V^\pi(s') \right]\end{aligned}$$

- Test your understanding:

$\pi'(s) = \pi(s)$ for some $s \in \mathcal{S}$? not necessarily

$\pi'(s) \neq \pi(s)$ for some $s \in \mathcal{S}$? not necessarily

$Q^\pi(s, \pi'(s)) \geq Q^\pi(s, \pi(s))$ for all $s \in \mathcal{S}$? TRUE

Policy improvement

Policy improvement

- Greedy policy:

Policy improvement

- Greedy policy:

$$\pi'(s) = \operatorname*{argmax}_a Q^\pi(s, a)$$

Policy improvement

- **Greedy policy:**

$$\pi'(s) \quad = \quad \operatorname*{argmax}_a Q^\pi(s,a)$$

- **Theorem:**
  The greedy policy $\pi'(s) = \arg\max_a Q^\pi(s,a)$ improves everywhere on the policy π from which it was derived:

Policy improvement

- **Greedy policy:**

$$\pi'(s) \quad = \quad \operatorname*{argmax}_a Q^\pi(s,a)$$

- **Theorem:**
  The greedy policy $\pi'(s) = \arg\max_a Q^\pi(s,a)$ improves everywhere on the policy π from which it was derived:

$$V^{\pi'}(s) \; \geq V^\pi(s) \quad \text{for all states } s \in \mathcal{S}$$

Policy improvement

- **Greedy policy:**

$$\pi'(s) \quad = \quad \underset{a}{\operatorname{argmax}}\; Q^\pi(s,a)$$

- **Theorem:**

  The greedy policy $\pi'(s) = \arg\max_a Q^\pi(s,a)$ improves everywhere on the policy π from which it was derived:

$$V^{\pi'}(s) \;\geq V^\pi(s) \quad \text{for all states } s \in \mathcal{S}$$

- **Intuition:**

Policy improvement

- **Greedy policy:**

$$\pi'(s) \quad = \quad \operatorname*{argmax}_a \, Q^\pi(s,a)$$

- **Theorem:**
  The greedy policy $\pi'(s) = \arg\max_a Q^\pi(s,a)$ improves everywhere on the policy π from which it was derived:

$$V^{\pi'}(s) \; \geq V^\pi(s) \quad \text{for all states } s \in \mathcal{S}$$

- **Intuition:**
  If it's better to choose action a in state s before following π, then it's always better to make this choice.

Policy improvement

- **Greedy policy:**

$$\pi'(s) \quad = \quad \underset{a}{\text{argmax}}\, Q^{\pi}(s,a)$$

- **Theorem:**
  The greedy policy $\pi'(s) = \arg\max_a Q^{\pi}(s,a)$ improves everywhere on the policy π from which it was derived:

$$V^{\pi'}(s) \;\geq V^{\pi}(s) \quad \text{for all states } s \in \mathcal{S}$$

- **Intuition:**
  If it's better to choose action a in state s before following π, then it's always better to make this choice.

- **Proof idea:**
  We'll prove a key inequality for *one-step deviations* from π, then we'll extend this inequality by an iterative argument.

Proof — 1. Deriving the inequality

- Comparing value functions:

Proof — 1. Deriving the inequality

- Comparing value functions:

$$V^{\pi}(s) \quad = \quad Q^{\pi}(s, \pi(s))$$

action that comes from π

Proof — 1. Deriving the inequality

- Comparing value functions:

$$\begin{aligned} V^\pi(s) &= Q^\pi(s, \pi(s)) \\ &\leq \max_a Q^\pi(s, a) \end{aligned}$$

Proof — 1. Deriving the inequality

- Comparing value functions:

$$\begin{aligned} V^\pi(s) &= Q^\pi(s, \pi(s)) \\ &\leq \max_a Q^\pi(s, a) \\ &= Q^\pi(s, \pi'(s)) \end{aligned}$$

Proof — 1. Deriving the inequality

- Comparing value functions:

$$
\begin{aligned}
V^\pi(s) &= Q^\pi(s, \pi(s)) \\
&\leq \max_a Q^\pi(s, a) \\
&= Q^\pi(s, \pi'(s)) \\
&= R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^\pi(s')
\end{aligned}
$$

- Comparing value functions:

$$\begin{aligned} V^\pi(s) &= Q^\pi(s, \pi(s)) \\ &\leq \max_a Q^\pi(s, a) \\ &= Q^\pi(s, \pi'(s)) \\ &= R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^\pi(s') \end{aligned}$$

- Combining these steps:

Proof — 1. Deriving the inequality

- Comparing value functions:

$$\begin{aligned} V^\pi(s) &= Q^\pi(s, \pi(s)) \\ &\leq \max_a Q^\pi(s, a) \\ &= Q^\pi(s, \pi'(s)) \\ &= R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^\pi(s') \end{aligned}$$

- Combining these steps:

$$V^\pi(s) \leq R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^\pi(s')$$

Proof — 1. Deriving the inequality

- Comparing value functions:

$$\begin{aligned} V^\pi(s) &= Q^\pi(s, \pi(s)) \\ &\leq \max_a Q^\pi(s, a) \\ &= Q^\pi(s, \pi'(s)) \\ &= R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^\pi(s') \end{aligned}$$

- Combining these steps:

$$V^\pi(s) \leq R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^\pi(s')$$

- Intuition:

Proof — 1. Deriving the inequality

- Comparing value functions:

$$\begin{aligned} V^\pi(s) &= Q^\pi(s, \pi(s)) \\ &\leq \max_a Q^\pi(s, a) \\ &= Q^\pi(s, \pi'(s)) \\ &= R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^\pi(s') \end{aligned}$$

- Combining these steps:

$$V^\pi(s) \leq R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^\pi(s')$$

- Intuition:

  It is better to take one step under π', then revert to π, than to always follow π.

Proof — 2. Leveraging the inequality

Proof — 2. Leveraging the inequality

- One-step inequality:

Proof — 2. Leveraging the inequality

- **One-step inequality:**

$$V^{\pi}(s) \;\leq\; R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^{\pi}(s')$$

What happens if we plug this inequality into itself?

Proof — 2. Leveraging the inequality

- One-step inequality:

$$V^{\pi}(s) \;\leq\; R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^{\pi}(s')$$

What happens if we plug this inequality into itself?
Then we obtain ...

Proof — 2. Leveraging the inequality

- **One-step inequality:**

$$V^{\pi}(s) \;\leq\; R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^{\pi}(s')$$

What happens if we plug this inequality into itself?
Then we obtain ...

- **Two-step inequality:**

- **One-step inequality:**

$$V^{\pi}(s) \;\le\; R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^{\pi}(s')$$

What happens if we plug this inequality into itself?
Then we obtain ...

- **Two-step inequality:**

$$V^{\pi}(s) \;\le\; R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) \left[R(s') + \gamma \sum_{s''} P(s''|s', \pi'(s')) V^{\pi}(s'') \right]$$

- **One-step inequality:**

$$V^{\pi}(s) \;\leq\; R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi}(s')$$

What happens if we plug this inequality into itself?
Then we obtain ...

- **Two-step inequality:**

$$V^{\pi}(s) \;\leq\; R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) \left[R(s') \;+\; \gamma \sum_{s''} P(s''|s',\pi'(s')) V^{\pi}(s'') \right]$$

- **Intuition:**

Proof — 2. Leveraging the inequality

- **One-step inequality:**

$$V^{\pi}(s) \;\leq\; R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^{\pi}(s')$$

  What happens if we plug this inequality into itself?
  Then we obtain ...

- **Two-step inequality:**

$$V^{\pi}(s) \;\leq\; R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) \left[R(s') \;+\; \gamma \sum_{s''} P(s''|s', \pi'(s')) V^{\pi}(s'') \right]$$

- **Intuition:**

  It is better to take **two** steps under π', then revert to π, than to always follow π.

Proof — 3. Taking the limit

Proof — 3. Taking the limit

- Two-step inequality:

Proof — 3. Taking the limit

- Two-step inequality:

$$V^{\pi}(s) \leq R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) \left[R(s') + \gamma \sum_{s''} P(s''|s', \pi'(s')) V^{\pi}(s'') \right]$$

Proof — 3. Taking the limit

- Two-step inequality:

$$V^{\pi}(s) \leq R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) \left[R(s') + \gamma \sum_{s''} P(s''|s', \pi'(s'))V^{\pi}(s'') \right]$$

- Apply the inequality t times:

Proof — 3. Taking the limit

- **Two-step inequality:**

$$V^{\pi}(s) \le R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) \left[R(s') + \gamma \sum_{s''} P(s''|s', \pi'(s')) V^{\pi}(s'') \right]$$

- **Apply the inequality t times:**

  It is better to take t steps under π', then revert to π, than to always follow π.

Proof — 3. Taking the limit

- **Two-step inequality:**

$$V^{\pi}(s) \leq R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) \left[R(s') + \gamma \sum_{s''} P(s''|s', \pi'(s')) V^{\pi}(s'') \right]$$

- **Apply the inequality t times:**

  It is better to take t steps under π', then revert to π, than to always follow π. Last term is of order $O(\gamma^t)$.

Proof — 3. Taking the limit

- **Two-step inequality:**

$$V^{\pi}(s) \leq R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) \left[R(s') + \gamma \sum_{s''} P(s''|s', \pi'(s')) V^{\pi}(s'') \right]$$

- **Apply the inequality t times:**

  It is better to take t steps under π', then revert to π, than to always follow π. Last term is of order $O(\gamma^t)$.

- **Take the limit $t \to \infty$:**

Proof — 3. Taking the limit

- **Two-step inequality:**

$$V^\pi(s) \le R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) \left[R(s') + \gamma \sum_{s''} P(s''|s',\pi'(s')) V^\pi(s'') \right]$$

- **Apply the inequality t times:**

  It is better to take t steps under π', then revert to π, than to always follow π. Last term is of order $O(\gamma^t)$.

- **Take the limit $t \to \infty$:**

  It is better to follow π' (always) than to follow π (always).

Proof — 3. Taking the limit

- **Two-step inequality:**

$$V^\pi(s) \le R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) \left[R(s') + \gamma \sum_{s''} P(s''|s',\pi'(s'))V^\pi(s'') \right]$$

- **Apply the inequality t times:**

  It is better to take t steps under π', then revert to π, than to always follow π. Last term is of order $O(\gamma^t)$.

- **Take the limit $t \to \infty$:**

  It is better to follow π' (always) than to follow π (always). Conclude that $V^\pi(s) \le V^{\pi'}(s)$ for all states $s \in \mathcal{S}$.

Policy iteration

Policy iteration

How to compute π^*?

Policy iteration

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.

Policy iteration

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.
2. Repeat until convergence:

Policy iteration

How to compute π^*?

1. **Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.**
2. **Repeat until convergence:**

   Compute the action value function $Q^\pi(s, a)$. ?

Policy iteration

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$.

2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a \, Q^\pi(s, a)$.

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.

2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a Q^\pi(s, a)$.
   Replace π by π'.

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.

2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a \, Q^\pi(s, a)$.
   Replace π by π'.

π_0

Policy iteration

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$.

2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a\, Q^\pi(s, a)$.
   Replace π by π'.

$$\pi_0 \xrightarrow{\text{evaluate}}$$

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.
2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a\, Q^\pi(s, a)$.
   Replace π by π'.

$$\pi_0 \xrightarrow{\text{evaluate}} V^{\pi_0}(s)$$

Policy iteration

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.
2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \mathrm{argmax}_a \, Q^\pi(s, a)$.
   Replace π by π'.

$$\pi_0 \quad \xrightarrow{\text{evaluate}} \quad \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s, a) \end{matrix}$$

Policy iteration

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.

2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \mathrm{argmax}_a\, Q^\pi(s, a)$.
   Replace π by π'.

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s, a) \end{matrix} \xrightarrow{\text{improve}}$$

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.

2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a \, Q^\pi(s, a)$.
   Replace π by π'.

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{array}{c} V^{\pi_0}(s) \\ Q^{\pi_0}(s, a) \end{array} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}}$$

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.

2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a\, Q^\pi(s, a)$.
   Replace π by π'.

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s, a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} V^{\pi_1}(s)$$

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.

2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a \, Q^\pi(s, a)$.
   Replace π by π'.

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s, a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_1}(s) \\ Q^{\pi_1}(s, a) \end{matrix}$$

How to compute π^*?

1. Choose an initial policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$.
2. Repeat until convergence:

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a Q^\pi(s, a)$.
   Replace π by π'.

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_1}(s) \\ Q^{\pi_1}(s,a) \end{matrix} \xrightarrow{\text{improve}}$$

How to compute π^*?

1. **Choose an initial policy $\pi : \mathcal{S} \to \mathcal{A}$.**

2. **Repeat until convergence:**

   Compute the action value function $Q^\pi(s, a)$.
   Compute the greedy policy $\pi'(s) = \text{argmax}_a \, Q^\pi(s, a)$.
   Replace π by π'

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s, a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_1}(s) \\ Q^{\pi_1}(s, a) \end{matrix} \xrightarrow{\text{improve}} \cdots$$

Policy iteration is guaranteed to terminate.

True (A) or False (B)?

Policy iteration

Policy iteration

- How to compute π^*?

Policy iteration

- How to compute π^*?

$$\pi_0 \xrightarrow{\quad\text{evaluate}\quad} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\quad\text{improve}\quad} \pi_1 \xrightarrow{\quad\text{evaluate}\quad} \cdots$$

Policy iteration

- How to compute π^*?

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \cdots$$

This process is guaranteed to terminate.

Policy iteration

- How to compute π^*?

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{array}{c} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{array} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \cdots$$

This process is guaranteed to terminate.
But does it converge to an optimal policy?

Policy iteration

- How to compute π^*?

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \cdots$$

This process is guaranteed to terminate.
But does it converge to an optimal policy?

- Theorem

Policy iteration

- How to compute π^*?

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \cdots$$

This process is guaranteed to terminate.
But does it converge to an optimal policy?

- **Theorem**

If $\pi'(s) = \arg\max_a Q^\pi(s,a)$ and $V^{\pi'}(s) = V^\pi(s)$ for all $s \in \mathcal{S}$,

Policy iteration

- How to compute π^*?

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \cdots$$

This process is guaranteed to terminate.
But does it converge to an optimal policy?

- Theorem

If $\pi'(s) = \arg\max_a Q^\pi(s,a)$ and $V^{\pi'}(s) = V^\pi(s)$ for all $s \in \mathcal{S}$, then $V^\pi(s) = V^*(s)$ for all $s \in \mathcal{S}$.

Policy iteration

- How to compute π^*?

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \cdots$$

  This process is guaranteed to terminate.
  But does it converge to an optimal policy?

- Theorem

  If $\pi'(s) = \arg\max_a Q^\pi(s,a)$ and $V^{\pi'}(s) = V^\pi(s)$ for all $s \in \mathcal{S}$, then $V^\pi(s) = V^*(s)$ for all $s \in \mathcal{S}$.

- Proof idea

Policy iteration

- **How to compute π^*?**

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \cdots$$

  This process is guaranteed to terminate.
  But does it converge to an optimal policy?

- **Theorem**

  If $\pi'(s) = \arg\max_a Q^\pi(s,a)$ and $V^{\pi'}(s) = V^\pi(s)$ for all $s \in \mathcal{S}$, then $V^\pi(s) = V^*(s)$ for all $s \in \mathcal{S}$.

- **Proof idea**

  Prove a key equality/inequality for terminal/non-terminal policies;

Policy iteration

- **How to compute π^*?**

$$\pi_0 \xrightarrow{\text{evaluate}} \begin{matrix} V^{\pi_0}(s) \\ Q^{\pi_0}(s,a) \end{matrix} \xrightarrow{\text{improve}} \pi_1 \xrightarrow{\text{evaluate}} \cdots$$

  This process is guaranteed to terminate.
  But does it converge to an optimal policy?

- **Theorem**

  If $\pi'(s) = \arg\max_a Q^\pi(s,a)$ and $V^{\pi'}(s) = V^\pi(s)$ for all $s \in \mathcal{S}$, then $V^\pi(s) = V^*(s)$ for all $s \in \mathcal{S}$.

- **Proof idea**

  Prove a key equality/inequality for terminal/non-terminal policies; iterate t times, then compare the limits as $t \to \infty$.

Proof — 1. Bellman optimality equation

- Suppose policy iteration converges to π'.

Proof — 1. Bellman optimality equation

- Suppose policy iteration converges to π'.

$$V^{\pi'}(s) \quad = \quad R(s) + \gamma \sum_{s'} P(s'|s, \pi'(s)) V^{\pi'}(s') \qquad \boxed{\text{Bellman equation}}$$

Proof — 1. Bellman optimality equation

- Suppose policy iteration converges to π'.

$$V^{\pi'}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi'}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\pi}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi}(s') \quad \boxed{\text{at convergence}}$$

Proof — 1. Bellman optimality equation

- Suppose policy iteration converges to π'.

$$V^{\pi'}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi'}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\pi}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi}(s') \quad \boxed{\text{at convergence}}$$

Now exploit that π' is greedy with respect to π ...

Proof — 1. Bellman optimality equation

- Suppose policy iteration converges to π'.

$$V^{\pi'}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi'}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\pi}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi}(s') \quad \boxed{\text{at convergence}}$$

Now exploit that π' is greedy with respect to π …

- Bellman optimality equation

Proof — 1. Bellman optimality equation

- Suppose policy iteration converges to π'.

$$V^{\pi'}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s))V^{\pi'}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\pi}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s))V^{\pi}(s') \quad \boxed{\text{at convergence}}$$

Now exploit that π' is greedy with respect to π ...

- Bellman optimality equation

$$\boxed{V^{\pi}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a)V^{\pi}(s')}$$

Proof — 1. Bellman optimality equation

- Suppose policy iteration converges to π'.

$$V^{\pi'}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi'}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\pi}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi}(s') \quad \boxed{\text{at convergence}}$$

Now exploit that π' is greedy with respect to π ...

- Bellman optimality equation

$$\boxed{V^{\pi}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\pi}(s')}$$

These equations are nonlinear due to the max operation.

Proof — 1. Bellman optimality equation

- Suppose policy iteration converges to π'.

$$V^{\pi'}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi'}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\pi}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\pi'(s)) V^{\pi}(s') \quad \boxed{\text{at convergence}}$$

Now exploit that π' is greedy with respect to π ...

- Bellman optimality equation

$$\boxed{V^{\pi}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\pi}(s')}$$

These equations are nonlinear due to the max operation.
There are n equations for n unknowns (where $s = 1, 2, \ldots, n$).

- Let $\tilde{\pi}$ be any policy of the MDP:

Proof — 2. Inequality

- Let $\tilde{\pi}$ be any policy of the MDP:

$$V^{\tilde{\pi}}(s) \quad = \quad R(s) + \gamma \sum_{s'} P(s'|s, \tilde{\pi}(s)) V^{\tilde{\pi}}(s') \qquad \boxed{\text{Bellman equation}}$$

Proof — 2. Inequality

- Let $\tilde{\pi}$ be any policy of the MDP:

$$
\begin{aligned}
V^{\tilde{\pi}}(s) &= R(s) + \gamma \sum_{s'} P(s'|s, \tilde{\pi}(s)) V^{\tilde{\pi}}(s') && \boxed{\text{Bellman equation}} \\
V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s, a) V^{\tilde{\pi}}(s') && \boxed{\text{greedy}}
\end{aligned}
$$

Proof — 2. Inequality

- Let $\tilde{\pi}$ be any policy of the MDP:

$$V^{\tilde{\pi}}(s) = R(s) + \gamma \sum_{s'} P(s'|s, \tilde{\pi}(s)) V^{\tilde{\pi}}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\tilde{\pi}}(s) \leq R(s) + \gamma \max_{a} \sum_{s'} P(s'|s, a) V^{\tilde{\pi}}(s') \quad \boxed{\text{greedy}}$$

- Compare to Bellman optimality equation (BOE):

- Let $\tilde{\pi}$ be any policy of the MDP:

$$V^{\tilde{\pi}}(s) = R(s) + \gamma \sum_{s'} P(s'|s, \tilde{\pi}(s))V^{\tilde{\pi}}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\tilde{\pi}}(s) \leq R(s) + \gamma \max_a \sum_{s'} P(s'|s, a)V^{\tilde{\pi}}(s') \quad \boxed{\text{greedy}}$$

- Compare to Bellman optimality equation (BOE):

$$V^{\pi}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s, a))V^{\pi}(s')$$

Proof — 2. Inequality

- Let $\tilde{\pi}$ be any policy of the MDP:

$$V^{\tilde{\pi}}(s) = R(s) + \gamma \sum_{s'} P(s'|s, \tilde{\pi}(s)) V^{\tilde{\pi}}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\tilde{\pi}}(s) \leq R(s) + \gamma \max_a \sum_{s'} P(s'|s, a) V^{\tilde{\pi}}(s') \quad \boxed{\text{greedy}}$$

- Compare to Bellman optimality equation (BOE):

$$V^{\pi}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s, a)) V^{\pi}(s')$$

- Understanding the difference:

Proof — 2. Inequality

- Let $\tilde{\pi}$ be any policy of the MDP:

$$
\begin{aligned}
V^{\tilde{\pi}}(s) &= R(s) + \gamma \sum_{s'} P(s'|s, \tilde{\pi}(s)) V^{\tilde{\pi}}(s') & \boxed{\text{Bellman equation}} \\
V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s, a) V^{\tilde{\pi}}(s') & \boxed{\text{greedy}}
\end{aligned}
$$

- Compare to Bellman optimality equation (BOE):

$$
V^{\pi}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s, a)) V^{\pi}(s')
$$

- Understanding the difference:

The inequality holds for any policy $\tilde{\pi}$ of the MDP.

Proof — 2. Inequality

- Let $\tilde{\pi}$ be any policy of the MDP:

$$V^{\tilde{\pi}}(s) = R(s) + \gamma \sum_{s'} P(s'|s,\tilde{\pi}(s)) V^{\tilde{\pi}}(s') \quad \boxed{\text{Bellman equation}}$$

$$V^{\tilde{\pi}}(s) \leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\tilde{\pi}}(s') \quad \boxed{\text{greedy}}$$

- Compare to Bellman optimality equation (BOE):

$$V^{\pi}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a)) V^{\pi}(s')$$

- Understanding the difference:

  The inequality holds for any policy $\tilde{\pi}$ of the MDP.
  The **BOE** only holds for a solution π from policy iteration.

Proof — 3. Taking the limit

Proof — 3. Taking the limit

- Iterating the inequality:

Proof — 3. Taking the limit

- Iterating the inequality:

$$V^{\tilde{\pi}}(s) \leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\tilde{\pi}}(s')$$

Proof — 3. Taking the limit

- Iterating the inequality:

$$
\begin{aligned}
V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\tilde{\pi}}(s') \\
&\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\tilde{\pi}}(s'') \right]
\end{aligned}
$$

Proof — 3. Taking the limit

- Iterating the inequality:

$$
\begin{aligned}
V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum\nolimits_{s'} P(s'|s,a) V^{\tilde{\pi}}(s') \\
&\leq R(s) + \gamma \max_a \sum\nolimits_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum\nolimits_{s''} P(s''|s',a') V^{\tilde{\pi}}(s'') \right]
\end{aligned}
$$

- Iterating the BOE:

Proof — 3. Taking the limit

- Iterating the inequality:

$$
\begin{aligned}
V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\tilde{\pi}}(s') \\
&\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\tilde{\pi}}(s'') \right]
\end{aligned}
$$

- Iterating the BOE:

$$
V^{\pi}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\pi}(s')
$$

Proof — 3. Taking the limit

- Iterating the inequality:

$$\begin{aligned} V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum\nolimits_{s'} P(s'|s,a) V^{\tilde{\pi}}(s') \\ &\leq R(s) + \gamma \max_a \sum\nolimits_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum\nolimits_{s''} P(s''|s',a') V^{\tilde{\pi}}(s'') \right] \end{aligned}$$

- Iterating the BOE:

$$\begin{aligned} V^{\pi}(s) &= R(s) + \gamma \max_a \sum\nolimits_{s'} P(s'|s,a) V^{\pi}(s') \\ &= R(s) + \gamma \max_a \sum\nolimits_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum\nolimits_{s''} P(s''|s',a') V^{\pi}(s'') \right] \end{aligned}$$

Proof — 3. Taking the limit

- Iterating the inequality:

$$
\begin{aligned}
V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\tilde{\pi}}(s') \\
&\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\tilde{\pi}}(s'') \right]
\end{aligned}
$$

- Iterating the BOE:

$$
\begin{aligned}
V^{\pi}(s) &= R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\pi}(s') \\
&= R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\pi}(s'') \right]
\end{aligned}
$$

- Iterating t times:

Proof — 3. Taking the limit

- **Iterating the inequality:**

$$
\begin{aligned}
V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\tilde{\pi}}(s') \\
&\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\tilde{\pi}}(s'') \right]
\end{aligned}
$$

- **Iterating the BOE:**

$$
\begin{aligned}
V^{\pi}(s) &= R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\pi}(s') \\
&= R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\pi}(s'') \right]
\end{aligned}
$$

- **Iterating t times:**

Both right sides agree up to term of order γ^t.

Proof — 3. Taking the limit

- Iterating the inequality:

$$
\begin{aligned}
V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\tilde{\pi}}(s') \\
&\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\tilde{\pi}}(s'') \right]
\end{aligned}
$$

- Iterating the BOE:

$$
\begin{aligned}
V^{\pi}(s) &= R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\pi}(s') \\
&= R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\pi}(s'') \right]
\end{aligned}
$$

- Iterating t times:

  Both right sides agree up to term of order γ^t.
  Taking the limit $t \to \infty$, we find $V^{\tilde{\pi}}(s) \leq V^{\pi}(s)$ for all $s \in \mathcal{S}$.

Proof — 3. Taking the limit

- **Iterating the inequality:**

$$
\begin{aligned}
V^{\tilde{\pi}}(s) &\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\tilde{\pi}}(s') \\
&\leq R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\tilde{\pi}}(s'') \right]
\end{aligned}
$$

- **Iterating the BOE:**

$$
\begin{aligned}
V^{\pi}(s) &= R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^{\pi}(s') \\
&= R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) \left[R(s') + \gamma \max_{a'} \sum_{s''} P(s''|s',a') V^{\pi}(s'') \right]
\end{aligned}
$$

- **Iterating t times:**

  Both right sides agree up to term of order γ^t.
  Taking the limit $t \to \infty$, we find $V^{\tilde{\pi}}(s) \leq V^{\pi}(s)$ for all $s \in \mathcal{S}$.

  Since $\tilde{\pi}$ is arbitrary, we conclude that π is optimal.

Value Iteration

Motivation

Motivation

- How policy iteration works:

Motivation

- How policy iteration works:

  It searches directly (and quite efficiently) through the combinatorially large space of policies in the MDP.

Motivation

- How policy iteration works:

  It searches directly (and quite efficiently) through the combinatorially large space of policies in the MDP.

- Is there another way?

Motivation

- How policy iteration works:

  It searches directly (and quite efficiently) through the combinatorially large space of policies in the MDP.

- Is there another way?

  Given an MDP = $\{\mathcal{S}, \mathcal{A}, P(s'|s, a), R(s), \gamma\}$, recall how its optimal policies and value functions are connected:

Motivation

- How policy iteration works:

  It searches directly (and quite efficiently) through the combinatorially large space of policies in the MDP.

- Is there another way?

  Given an MDP = $\{\mathcal{S}, \mathcal{A}, P(s'|s,a), R(s), \gamma\}$, recall how its optimal policies and value functions are connected:

$$\pi^*(s) \quad = \quad \operatorname*{argmax}_a \left[Q^*(s,a) \right]$$

Motivation

- How policy iteration works:

  It searches directly (and quite efficiently) through the combinatorially large space of policies in the MDP.

- Is there another way?

  Given an MDP = $\{\mathcal{S}, \mathcal{A}, P(s'|s,a), R(s), \gamma\}$, recall how its optimal policies and value functions are connected:

$$\begin{aligned} \pi^*(s) &= \operatorname*{argmax}_a \left[Q^*(s,a) \right] \\ &= \operatorname*{argmax}_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right] \end{aligned}$$

Motivation

- How policy iteration works:

  It searches directly (and quite efficiently) through the combinatorially large space of policies in the MDP.

- Is there another way?

  Given an MDP = $\{\mathcal{S}, \mathcal{A}, P(s'|s,a), R(s), \gamma\}$, recall how its optimal policies and value functions are connected:

$$\begin{aligned}\pi^*(s) &= \underset{a}{\text{argmax}}\left[Q^*(s,a)\right] \\ &= \underset{a}{\text{argmax}}\left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s')\right]\end{aligned}$$

  So if we can directly compute the optimal value function $V^*(s)$,

- How policy iteration works:

  It searches directly (and quite efficiently) through the combinatorially large space of policies in the MDP.

- Is there another way?

  Given an MDP = $\{\mathcal{S}, \mathcal{A}, P(s'|s,a), R(s), \gamma\}$, recall how its optimal policies and value functions are connected:

$$\begin{aligned}\pi^*(s) &= \underset{a}{\operatorname{argmax}} \left[Q^*(s,a) \right] \\ &= \underset{a}{\operatorname{argmax}} \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right]\end{aligned}$$

  So if we can directly compute the optimal value function $V^*(s)$, then we can use it to derive an optimal policy π^*.

Bellman optimality equation

Bellman optimality equation

- Derivation:

- Derivation:

$$V^*(s) \quad = \quad \max_a \left[Q^*(s,a) \right]$$

- Derivation:

$$\begin{aligned} V^*(s) &= \max_a \left[Q^*(s,a) \right] \\ &= \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right] \end{aligned}$$

Bellman optimality equation

- Derivation:

$$\begin{aligned} V^*(s) &= \max_a \Big[Q^*(s,a) \Big] \\ &= \max_a \Big[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \Big] \end{aligned}$$

- Solution?

Bellman optimality equation

- Derivation:

$$
\begin{aligned}
V^*(s) &= \max_a \left[Q^*(s,a) \right] \\
&= \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right]
\end{aligned}
$$

- Solution?

Suppose we know the parameters $\{R(s), P(s'|s,a), \gamma\}$.

Bellman optimality equation

- Derivation:

$$\begin{aligned} V^*(s) &= \max_a \left[Q^*(s,a) \right] \\ &= \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right] \end{aligned}$$

- Solution?

  Suppose we know the parameters $\{R(s), P(s'|s,a), \gamma\}$.
  Then the above gives us n equations for n unknowns:

Bellman optimality equation

- Derivation:

$$
\begin{aligned}
V^*(s) &= \max_a \left[Q^*(s,a) \right] \\
&= \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right]
\end{aligned}
$$

- Solution?

Suppose we know the parameters $\{R(s), P(s'|s,a), \gamma\}$.
Then the above gives us n equations for n unknowns:

$$
V^*(s) = \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right]
$$

Bellman optimality equation

- **Derivation:**

$$
\begin{aligned}
V^*(s) &= \max_a \Big[Q^*(s,a) \Big] \\
&= \max_a \Big[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \Big]
\end{aligned}
$$

- **Solution?**

Suppose we know the parameters $\{R(s), P(s'|s,a), \gamma\}$.
Then the above gives us n equations for n unknowns:

$$
V^*(s) = \max_a \Big[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \Big]
$$

But how to solve these nonlinear equations for $V^*(s)$?

Value iteration

Value iteration

- Idea in a nutshell

Value iteration

- **Idea in a nutshell**

  Replace the equality sign in the Bellman optimality equation by an assignment operation:

Value iteration

- **Idea in a nutshell**

  Replace the equality sign in the Bellman optimality equation by an assignment operation:

$$V^*(s) \quad = \quad \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right] \quad \boxed{\text{BOE}}$$

Value iteration

- Idea in a nutshell

  Replace the equality sign in the Bellman optimality equation by an assignment operation:

$$V^*(s) = \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right] \quad \boxed{\text{BOE}}$$

$$V_{\text{new}}(s) \longleftarrow \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V_{\text{old}}(s') \right] \quad \boxed{\text{algorithm}}$$

Value iteration

- Idea in a nutshell

  Replace the equality sign in the Bellman optimality equation by an assignment operation:

$$V^*(s) \quad = \quad \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right] \quad \boxed{\text{BOE}}$$

$$V_{\text{new}}(s) \quad \longleftarrow \quad \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V_{\text{old}}(s') \right] \quad \boxed{\text{algorithm}}$$

- Why this might work

- **Idea in a nutshell**

  Replace the equality sign in the Bellman optimality equation by an assignment operation:

$$V^*(s) \quad = \quad \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right] \quad \boxed{\text{BOE}}$$

$$V_{\text{new}}(s) \quad \longleftarrow \quad \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V_{\text{old}}(s') \right] \quad \boxed{\text{algorithm}}$$

- **Why this might work**

  The value function $V^*(s)$ is a *fixed point* of this iteration.

- **Idea in a nutshell**

  Replace the equality sign in the Bellman optimality equation by an assignment operation:

$$V^*(s) \quad = \quad \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V^*(s') \right] \quad \boxed{\text{BOE}}$$

$$V_{\text{new}}(s) \quad \longleftarrow \quad \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V_{\text{old}}(s') \right] \quad \boxed{\text{algorithm}}$$

- **Why this might work**

  The value function $V^*(s)$ is a *fixed point* of this iteration. But does this iteration always converge to a valid solution?

Algorithm for value iteration

Algorithm for value iteration

1. **Initialize:** $V_0(s) = 0$ for all $s \in \mathcal{S}$.

1. **Initialize:** $V_0(s) = 0$ for all $s \in \mathcal{S}$.
2. **Iterate until convergence:**

Algorithm for value iteration

1. **Initialize:** $V_o(s) = 0$ for all $s \in \mathcal{S}$.

2. **Iterate until convergence:**

$$V_{k+1}(s) \; = \; \max_a \left[R(s) \, + \, \gamma \sum_{s'} P(s'|s,a) \, V_k(s') \right]$$

Algorithm for value iteration

1. **Initialize:** $V_0(s) = 0$ for all $s \in \mathcal{S}$.

2. **Iterate until convergence:**

$$V_{k+1}(s) \ = \ \max_a \left[R(s) \ + \ \gamma \sum_{s'} P(s'|s,a)\, V_k(s') \right] \text{ for all } s \in \mathcal{S}.$$

1. **Initialize:** $V_0(s) = 0$ for all $s \in \mathcal{S}$.

2. **Iterate until convergence:**

$$V_{k+1}(s) \; = \; \max_a \left[R(s) \, + \, \gamma \sum_{s'} P(s'|s,a) \, V_k(s') \right] \text{ for all } s \in \mathcal{S}.$$

3. **Solve for optimal policy**:

Algorithm for value iteration

1. **Initialize:** $V_o(s) = 0$ for all $s \in \mathcal{S}$.

2. **Iterate until convergence:**

$$V_{k+1}(s) \; = \; \max_a \left[R(s) \; + \; \gamma \sum_{s'} P(s'|s,a)\, V_k(s') \right] \text{ for all } s \in \mathcal{S}.$$

3. **Solve for optimal policy**:

$$Q_k(s,a) \quad = \quad R(s) + \gamma \sum_{s'} P(s'|s,a)\, V_k(s'),$$

Algorithm for value iteration

1. **Initialize:** $V_0(s) = 0$ for all $s \in \mathcal{S}$.

2. **Iterate until convergence:**

$$V_{k+1}(s) = \max_a \left[R(s) + \gamma \sum_{s'} P(s'|s,a)\, V_k(s') \right] \text{ for all } s \in \mathcal{S}.$$

3. **Solve for optimal policy:**

$$\begin{aligned} Q_k(s,a) &= R(s) + \gamma \sum_{s'} P(s'|s,a)\, V_k(s'), \\ \pi^*(s) &= \lim_{k \to \infty} \operatorname{argmax}_a Q_k(s,a). \end{aligned}$$

Value iteration (VI) versus policy iteration (PI)

Value iteration (VI) versus policy iteration (PI)

- Compare and contrast:

Value iteration (VI) versus policy iteration (PI)

- **Compare and contrast:**

  PI searches through the combinatorial space of policies.

Value iteration (VI) versus policy iteration (PI)

- **Compare and contrast:**

  PI searches through the combinatorial space of policies. VI searches through the continuous space of value functions.

Value iteration (VI) versus policy iteration (PI)

- **Compare and contrast:**

  PI searches through the combinatorial space of policies.
  VI searches through the continuous space of value functions.

- **Convergence:**

Value iteration (VI) versus policy iteration (PI)

- **Compare and contrast:**

  PI searches through the combinatorial space of policies.
  VI searches through the continuous space of value functions.

- **Convergence:**

  PI converges in a finite number of steps.

Value iteration (VI) versus policy iteration (PI)

- **Compare and contrast:**

  PI searches through the combinatorial space of policies.
  VI searches through the continuous space of value functions.

- **Convergence:**

  PI converges in a finite number of steps.
  VI converges asymptotically (in the limit).

That’s all folks!