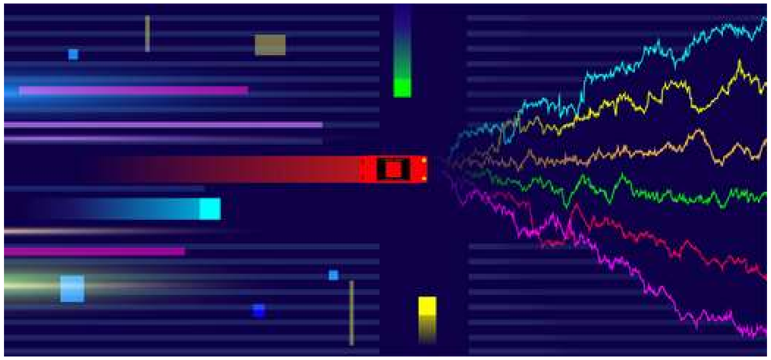
Value Iteration and Policy Iteration

Sid Banerjee
Cornell University



Infinite Horizon Discounted

A MDP is defined by: $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, r, T, s_0, \gamma\}$

$\mathcal{S}$	State space
$\mathcal{A}$	Action space
$r : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$	Reward
$T : \mathcal{S} \times \mathcal{A} \rightarrow \Delta(\mathcal{S})$	Transitions
$\gamma \in [0, 1)$	Discount

Value Function

The Value Function is expected return for policy

$$V^\pi(s) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(S_h, A_h) \mid S_0 = s, A_h \sim \pi(S_h), S_{h+1} \sim T(\cdot \mid S_h, A_h)\right]$$

$$Q^\pi(s,a) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(S_h, A_h) \mid (S_0, A_0) = (s,a), A_h \sim \pi(S_h), S_{h+1} \sim T(\cdot \mid S_h, A_h)\right]$$

The Bellman Equations note that:

$$V^\pi(s) = \mathbb{E}_{A\sim\pi(s)}[r(s,A) + \gamma \mathbb{E}_{S'\sim T(\cdot|s,A)}[V^\pi(S')]]$$

$$Q^\pi(s,a) = r(s,a) + \gamma \mathbb{E}_{S'\sim T(\cdot|s,a)}[V^\pi(S')]$$

Main Question

Given a MDP, how do we find the optimal policy?

Main Question

Given a MDP, how do we find the optimal policy?

Fully Known Model

- Reward function, transition distribution fully known
- Understand computational complexity to scale to large problems

Generative Model

- Sample from reward function / transition distribution from arbitrary (state,action)
- Understand statistical complexity to scale to large problems
- No issue of dynamic environment

Online Model

- Sample trajectory under current policy, update policy, repeat
- Understand statistical complexity
- *"Most complex"*, additional correlations in estimates

Main Question

Maybe a better model….

Exogenous MDP

- Unknown distribution over exogenous inputs (i.e. arrivals)
- Known reward and transition as function of exogenous trace
- Access to historical data of exogenous inputs

Main Question

Given a MDP, how do we find the optimal policy?

Fully Known Model

- Reward function, transition distribution fully known
- Understand computational complexity to scale to large problems

Generative Model

- Sample from reward function / transition distribution from arbitrary (state,action)
- Understand statistical complexity to scale to large problems
- No issue of dynamic environment

Online Model

- Sample trajectory under current policy, update policy, repeat
- Understand statistical complexity
- *"Most complex"*, additional correlations in estimates

Main Question

Given a MDP, how do we find the optimal policy?

Fully Known Model

- Reward function, transition distribution fully known
- Understand computational complexity to scale to large problems

Two Approaches

Value Iteration
Policy Iteration

Bellman Operator

Define the Bellman Operator, which given an arbitrary function:

$$(\mathscr{T} f)(s,a) = r(s,a) + \gamma \mathbb{E}_{S' \sim T(\cdot|s,a)}[\max_{a' \in \mathcal{A}} f(S',a')]$$

By Bellman Optimality we know: $\mathscr{T} Q^* = Q^*$

Bellman Operator

Define the Bellman Operator, which given an arbitrary function:

$$(\mathscr{T} f)(s,a) = r(s,a) + \gamma \mathbb{E}_{S' \sim T(\cdot|s,a)}[\max_{a' \in \mathcal{A}} f(S', a')]$$

By Bellman Optimality we know: $\mathscr{T} Q^* = Q^*$

If it was a contraction (it is), iterate!

Value Iteration

$$(\mathscr{T} f)(s,a) = r(s,a) + \gamma \mathbb{E}_{S' \sim T(\cdot|s,a)}[\max_{a' \in \mathcal{A}} f(S',a')]$$

Initialize: $Q^0(s,a) \in \left(0, \frac{1}{1-\gamma}\right)$

Iterate until convergence: $Q^{t+1} = \mathscr{T} Q^t$

Value Iteration

$$(\mathscr{T} f)(s,a) = r(s,a) + \gamma \mathbb{E}_{S' \sim T(\cdot|s,a)} [\max_{a' \in \mathcal{A}} f(S',a')]$$

Initialize: $Q^0(s,a) \in \left(0, \frac{1}{1-\gamma}\right)$

Iterate until convergence: $Q^{t+1} = \mathscr{T} Q^t$

Couple notes:

- Explicitly using known model
- Storage/time scales with size of action + state space

Value Iteration

$$(\mathscr{T} f)(s,a) = r(s,a) + \gamma \mathbb{E}_{S' \sim T(\cdot|s,a)}[\max_{a' \in \mathcal{A}} f(S',a')]$$

Initialize: $Q^0(s,a) \in \left(0, \frac{1}{1-\gamma}\right)$

Iterate until convergence: $Q^{t+1} = \mathscr{T} Q^t$

The Bellman operator is a $\boldsymbol{\gamma}$ contraction, so:

$$\|Q^t - Q^*\| \leq \gamma^t \|Q^0 - Q^*\|$$

Value Iteration

The Bellman operator is a $\boldsymbol{\gamma}$ contraction, so:

$$\|Q^t - Q^*\| \leq \gamma^t \|Q^0 - Q^*\|$$

Not a guarantee on value of final policy, since $Q^t \neq Q^{\pi^t}$

$\pi^t(s) = \mathrm{argmax}_{a \in \mathcal{A}} Q^t(s, a)$

$$V^{\pi^t}(s) \geq V^*(s) - \frac{2\gamma^t}{1-\gamma} \|Q^0 - Q^*\|_\infty$$

Main Question

Given a MDP, how do we find the optimal policy?

Fully Known Model

- Reward function, transition distribution fully known
- Understand computational complexity to scale to large problems

Two Approaches

Value Iteration
Policy Iteration

Value Function

The Bellman Equations note that:

$$V^\pi(s) = \mathbb{E}_{A\sim\pi(s)}[r(s,A) + \gamma \mathbb{E}_{S'\sim T(\cdot|s,A)}[V^\pi(S')]]$$
$$Q^\pi(s,a) = r(s,a) + \gamma \mathbb{E}_{S'\sim T(\cdot|s,a)}[V^\pi(S')]$$

Via some linear algebra.... $T^\pi(s',s) = \sum_a \pi(a \mid s) T(s' \mid s,a)$

$$V^\pi = (I - \gamma T^\pi)^{-1} r$$
$$Q^\pi = r + \gamma V^\pi$$

Policy Iteration

Initialize: $\pi^0(s) : \mathcal{A} \to \Delta(\mathcal{A})$

Evaluate / solve Bellman Eqs for: $Q^{\pi^t}(s, a)$

Policy Improvement: $\pi^{t+1}(s) = \mathrm{argmax}_{a \in \mathcal{A}} Q^{\pi^t}(s, a)$

Policy Iteration

Initialize: $\pi^0(s) : \mathcal{A} \rightarrow \Delta(\mathcal{A})$

Evaluate / solve Bellman Eqs for: $Q^{\pi^t}(s,a)$

Policy Improvement: $\pi^{t+1}(s) = \mathrm{argmax}_{a \in \mathcal{A}} Q^{\pi^t}(s,a)$

Couple notes:

- Explicitly using known model
- Storage/time scales with size of action + state space (solving Bellman Eqs)

Natural Question

Value Iteration vs Policy Iteration

Which one is faster? How many iterations (computational complexity) are needed to find optimal policy?

Value Iteration

$$(\mathscr{T}f)(s,a) = r(s,a) + \gamma \mathbb{E}_{S' \sim T(\cdot|s,a)}[\max_{a' \in \mathcal{A}} f(S',a')]$$

Initialize: $Q^0(s,a) \in \left(0, \frac{1}{1-\gamma}\right)$

Iterate until convergence: $Q^{t+1} = \mathscr{T} Q^t$

Per iteration complexity: $S^2 A$

Policy Iteration

Initialize: $\pi^0(s) : \mathcal{A} \to \Delta(\mathcal{A})$

Evaluate / solve Bellman Eqs for: $Q^{\pi^t}(s,a)$

Policy Improvement: $\pi^{t+1}(s) = \mathrm{argmax}_{a\in\mathcal{A}} Q^{\pi^t}(s,a)$

Per iteration complexity: hard to quantify, need to evaluate Q value for current policy at each iteration

Natural Question

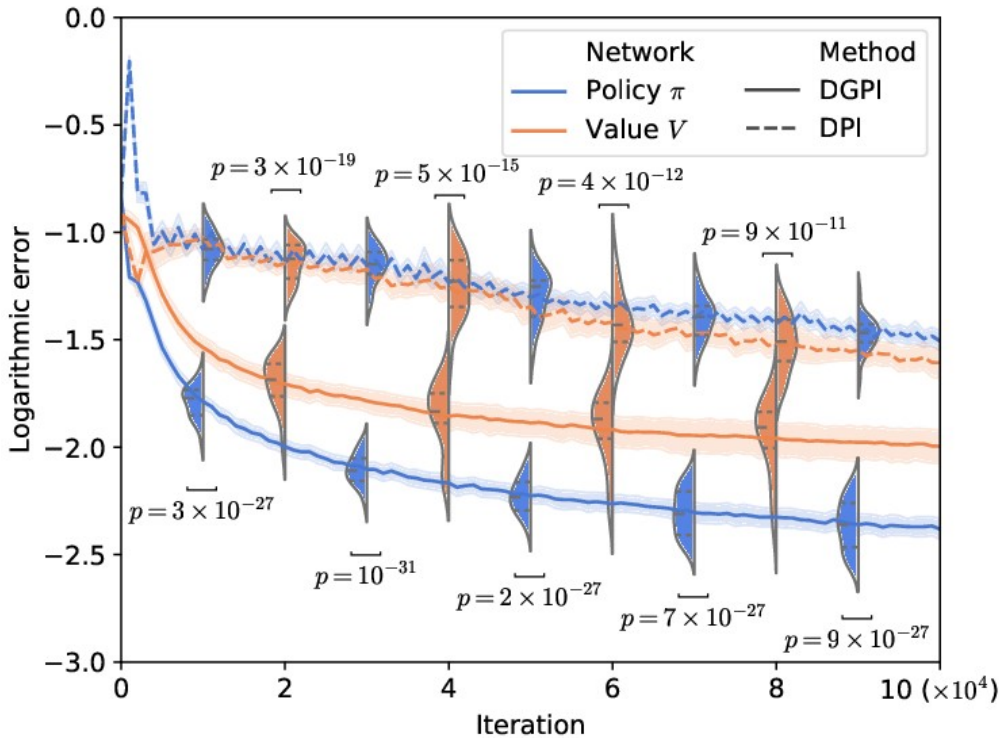
Value Iteration vs Policy Iteration

Which one is faster? How many iterations (computational complexity) are needed to find optimal policy?

Neither are strongly polynomial time, but PI observed to be faster than VI

Natural Question

Neither are strongly polynomial time, but PI observed to be faster than VI



Natural Question

Can we design a polynomial time algorithm?

In comes linear programming.....

Primal LP

Can equivalently write Bellman Equation as a linear program

$$\min V(s_0)$$
$$\text{s.t.} V(s) \geq r(s,a) + \mathbb{E}_{S' \sim T(\cdot|s,a)}[V(S')]$$

Primal LP

Can equivalently write Bellman Equation as a linear program

$$\min V(s_0)$$
$$\text{s.t.} V(s) \geq r(s,a) + \mathbb{E}_{S' \sim T(\cdot|s,a)}[V(S')]$$

$$V^*(s) = \max_{a \in \mathcal{A}} r(s,a) + \gamma \mathbb{E}_{S' \sim T(\cdot|s,a)}[V^*(S')]$$

Encoding max operator through constraints

Primal LP

Can equivalently write Bellman Equation as a linear program

$$\min V(s_0)$$
$$\text{s.t.} V(s) \geq r(s,a) + \mathbb{E}_{S' \sim T(\cdot|s,a)}[V(S')]$$

Generic polytime LP solver gives polytime algorithm (interior point algorithm is strongly polynomial)

Primal LP

Can equivalently write Bellman Equation as a linear program

$$\min V(s_0)$$
$$\text{s.t.} V(s) \geq r(s,a) + \mathbb{E}_{S' \sim T(\cdot|s,a)}[V(S')]$$

Generic polytime LP solver gives polytime algorithm (interior point algorithm is strongly polynomial)

VI ~ Fixed Point Algorithm
PI ~ Block Simplex Algorithm

Primal LP

Can equivalently write Bellman Equation as a linear program

$$
\begin{aligned}
&\min V(s_0) \\
&\text{s.t.} V(s) \geq r(s,a) + \mathbb{E}_{S' \sim T(\cdot|s,a)}[V(S')]
\end{aligned}
$$

Generic polytime LP solver gives polytime algorithm (interior point algorithm is strongly polynomial)

What is the dual?

Dual LP

The dual of the Bellman Equation LP:

$$\max \sum_{s,a} \nu(s,a) r(s,a)$$

$$\text{s.t.} \sum_{a} \nu(s,a) = (1-\gamma)\mathbb{I}[s = s_0] + \gamma \sum_{s',a'} T(s \mid s', a') \nu(s', a')$$

Flow constraints

$\nu(s,a)$ State-action visitation distribution for optimal policy

$$\nu(s,a) = (1-\gamma) \textstyle\sum_{t=0}^{\infty} \gamma^t \Pr(S_t = S, A_t = A \mid s_0)$$

Dual LP

The dual of the Bellman Equation LP:

$$\max \sum_{s,a} \nu(s,a) r(s,a)$$

$$\text{s.t.} \sum_{a} \nu(s,a) = (1-\gamma)\mathbb{I}[s = s_0] + \gamma \sum_{s',a'} T(s \mid s', a') \nu(s', a')$$

Lots of recent work on understanding and exploiting LP properties of RL formulation

References

[Puterman1994] Martin Puterman. “Markov Decision Processes: Discrete Stochastic Dynamic Programming”. *John Wiley + Sons*, 1994.
[Sutton2018] Richard Sutton. “Reinforcement Learning: An Introduction.” *MIT Press*, 2018.
[Agarwal2021] Alekh Agarwal, Nan Jiang, Sham M. Kakade, Wen Sun. “Reinforcement Learning: Theory and Algorithms”. 2021.
[Slivkins2019] Alexsandrs Slivkins. “Introduction to Multi-Armed Bandits.” *Foundations and Trends in ML*, 2019.
[Powell2021] Warren Powell. “Reinforcement Learning and Stochastic Optimization.” 2021.
[Meyn2021] Sean Meyn. “Control Systems and Reinforcement Learning”. *Cambridge University Press*, 2021.
[Neu2020] Gergely Neu, Ciara Pike-Burke. “A Unifying View of Optimism in Episodic Reinforcement Learning”. *NeurIPS*, 2020.