RLearning:
Short guides to reinforcement learning

Unit 2-5: Policy Iteration

Davud Rostam-Afschar (Uni Mannheim)

How can we solve for the best policy of each state?

Policy Optimization

- Value iteration
  - Optimize value function
  - Extract induced policy in last step
- Can we directly optimize the policy?
  - Yes, by policy iteration

Readings: Policy Iteration

?, section 4.3
?, sections 6.4-6.5
?, section 17.3

Policy Iteration

- Alternate between two steps

$$\pi_0 \xrightarrow{E} V^{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} V^{\pi_1} \xrightarrow{I} \pi_2 \xrightarrow{E} \cdots \xrightarrow{I} \pi^* \xrightarrow{E} V^*$$

1. Policy Evaluation

$$V^\pi(s) = R(s, \pi(s)) + \gamma \sum_{s'} \mathbb{P}\left(s' \mid s, \pi(s)\right) V^\pi\left(s'\right) \ \forall s$$

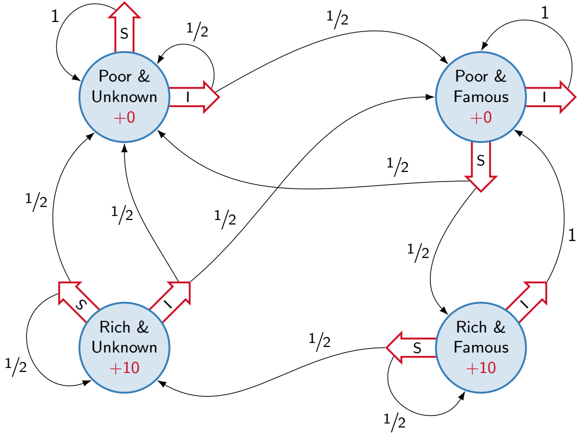
2. Policy improvement

$$\pi(s) \leftarrow \underset{a}{\operatorname{argmax}}\ R(s, a) + \gamma \sum_{s'} \mathbb{P}\left(s' \mid s, a\right) V^\pi\left(s'\right) \ \forall s$$

Policy Iteration Algorithm

policyIteration(MDP)
Initialize π_0 to any policy
$n \leftarrow 0$
Repeat
    Eval: $V_n = R^{\pi_n} + \gamma T^{\pi_n} V_n$
    Improve: $\pi_{n+1} \leftarrow \text{argmax } R^a + \gamma T^a V_n$
    $n \leftarrow n + 1$
Until $\pi_{n+1} = \pi_n$
Return π_n

Example (Policy Iteration)



t	$V(PU)$	$\pi(PU)$	$V(PF)$	$\pi(PF)$	$V(RU)$	$\pi(RU)$	$V(RF)$	$\pi(RF)$
0	0	I	0	I	10	I	10	I
1	31.6	I	38.6	S	44.0	S	54.2	S
2	31.6	I	38.6	S	44.0	S	54.2	S

Monotonic Improvement

- **Lemma 1:** Let V_n and V_{n+1} be successive value functions in policy iteration. Then $V_{n+1} \geq V_n$.

Monotonic Improvement

- **Lemma 1:** Let V_n and V_{n+1} be successive value functions in policy iteration. Then $V_{n+1} \geq V_n$.
- Proof:
  - We know that $H^*(V_n) \geq H^{\pi_n}(V_n) = (V_n)$
  - Let $\pi_{n+1} = \underset{a}{\operatorname{argmax}}\, R^a + \gamma T^a V_n$
  - Then $H^*(V_n) = R^{\pi_{n+1}} + \gamma T^{\pi_{n+1}} V_n \geq V_n$
  - Rearranging: $R^{\pi_{n+1}} \geq (I - \gamma T^{\pi_{n+1}}) V_n$
  - Hence $V_{n+1} = (I - \gamma T^{\pi_{n+1}})^{-1} R^{\pi_{n+1}} \geq V_n$

- **Theorem 2:** Policy iteration converges to π^* and V^* in finitely many iterations when S and A are finite.

- **Theorem 2:** Policy iteration converges to π^* and V^* in finitely many iterations when S and A are finite.
- Proof:
  - We know that $V_{n+1} \geq V_n \quad \forall n$ by Lemma 1.
  - Since A and S are finite, there are finitely many policies and therefore the algorithm terminates in finitely many iterations.
  - At termination, $\pi_n = \pi_{n+1}$ and therefore V_n satisfies

    Bellman's equation:

$$V_n = V_{n+1} = \max_a R^a + \gamma T^a V_n$$

Complexity

- Value Iteration:
  - Cost per iteration: $\mathcal{O}\left(|S|^2|A|\right)$
  - Many iterations: linear convergence
- Policy Iteration:
  - Cost per iteration: $\mathcal{O}\left(|S|^3 + |S|^2|A|\right)$
  - Few iterations: (early) linear, (late) quadratic convergence

Modified Policy Iteration Algorithm

- Alternate between two steps
  1. **Partial** Policy evaluation
     Repeat k times:

$$V^{\pi}(s) = R(s, \pi(s)) + \gamma \sum_{s'} \mathbb{P}\left(s' \mid s, \pi(s)\right) V^{\pi}\left(s'\right) \ \forall s$$

  2. Policy improvement

$$\pi(s) \leftarrow \underset{a}{\operatorname{argmax}} \ R(s, a) + \gamma \sum_{s'} \mathbb{P}\left(s' \mid s, a\right) V^{\pi}\left(s'\right) \ \forall s$$

Modified Policy Iteration Algorithm

modifiedPolicyIteration(MDP)
Initialize π_0 and V_0 to anything
$n \leftarrow 0$
Repeat
    Eval: Repeat k times
    $V_n \leftarrow R^{\pi_n} + \gamma T^{\pi_n} V_n$
    Improve: $\pi_{n+1} \leftarrow \underset{a}{\text{argmax}}\ R^a + \gamma T^a V_n$
    $V_{n+1} \leftarrow \max_a R^a + \gamma T^a V_n$
    $n \leftarrow n + 1$
Until $\|V_n - V_{n-1}\|_\infty \leq \epsilon$
Return π_n

Convergence

- Same convergence guarantees as value iteration:
  - Value function V_n : $\quad \|V_n - V^*\|_\infty \leq \frac{\epsilon}{1-\gamma}$
  - Value function V^{π_n} of policy π_n:

$$\|V^{\pi_n} - V^*\|_\infty \leq \frac{2\epsilon}{1-\gamma}$$

  - Proof: somewhat complicated **?**, section 6.5

Complexity

- Value Iteration:
  - Each iteration: $\mathcal{O}\left(|S|^2|A|\right)$
  - Many iterations: linear convergence

Complexity

- Value Iteration:
  - Each iteration: $\mathcal{O}\left(|S|^2|A|\right)$
  - Many iterations: linear convergence
- Policy Iteration:
  - Each iteration: $\mathcal{O}\left(|S|^3 + |S|^2|A|\right)$
  - Few iterations: linear-quadratic convergence

Complexity

- Value Iteration:
  - Each iteration: $\mathcal{O}\left(|S|^2|A|\right)$
  - Many iterations: linear convergence
- Policy Iteration:
  - Each iteration: $\mathcal{O}\left(|S|^3 + |S|^2|A|\right)$
  - Few iterations: linear-quadratic convergence
- Modified Policy Iteration:
  - Each iteration: $\mathcal{O}\left(k|S|^2 + |S|^2|A|\right)$
  - Few iterations: linear-quadratic convergence

Puterman, M. L. (2014): *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Russell, S. J., and P. Norvig (2016): *Artificial intelligence: a modern approach*. Pearson.

Sutton, R. S., and A. G. Barto (2018): "Reinforcement learning: An introduction," *A Bradford Book*, Available at http://incompleteideas.net/book/the-book-2nd.html.

Takeaways

How Does Policy Iteration Work?

- Alternates policy evaluation and improvement, ensuring monotonic value gains
- Converges in finite steps for finite MDPs to the optimal policy and value
- Modified policy iteration trades off full evaluation for efficiency
- Fewer iterations than value iteration, but each is costlier