Understanding (Exact) Dynamic Programming through Bellman Operators

Ashwin Rao
ICME, Stanford University
January 15, 2019
Overview

1. Value Functions as Vectors
2. Bellman Operators
3. Contraction and Monotonicity
4. Policy Evaluation
5. Policy Iteration
6. Value Iteration
7. Policy Optimality
Value Functions as Vectors

- Assume State space $S$ consists of $n$ states: $\{s_1, s_2, \ldots, s_n\}$
- Assume Action space $A$ consists of $m$ actions $\{a_1, a_2, \ldots, a_m\}$
- This exposition extends easily to continuous state/action spaces too.
- We denote a stochastic policy as $\pi(a|s)$ (probability of “a given s”)
- Abusing notation, deterministic policy denoted as $\pi(s) = a$
- Consider $n$-dim space $\mathbb{R}^n$, each dim corresponding to a state in $S$
- Think of a Value Function (VF) $v: S \rightarrow \mathbb{R}$ as a vector in this space
- With coordinates $[v(s_1), v(s_2), \ldots, v(s_n)]$
- Value Function (VF) for a policy $\pi$ is denoted as $v_\pi : S \rightarrow \mathbb{R}$
- Optimal VF denoted as $v_* : S \rightarrow \mathbb{R}$ such that for any $s \in S$,
  $$v_*(s) = \max_\pi v_\pi(s)$$
Some more notation

- Denote $R_s^a$ as the Expected Reward upon action $a$ in state $s$
- Denote $P_{s,s'}^a$ as the probability of transition $s \rightarrow s'$ upon action $a$
- Define

$$R_\pi(s) = \sum_{a \in A} \pi(a | s) \cdot R_s^a$$

$$P_\pi(s, s') = \sum_{a \in A} \pi(a | s) \cdot P_{s,s'}^a$$

- Denote $R_\pi$ as the vector $[R_\pi(s_1), R_\pi(s_2), \ldots, R_\pi(s_n)]$
- Denote $P_\pi$ as the matrix $[P_\pi(s_i, s_{i'})], 1 \leq i, i' \leq n$
- Denote $\gamma$ as the MDP discount factor
Bellman Operators $B_\pi$ and $B_*$

- We define operators that transform a VF vector to another VF vector.
- **Bellman Policy Operator** $B_\pi$ (for policy $\pi$) operating on VF vector $v$:
  \[
  B_\pi v = R_\pi + \gamma P_\pi \cdot v
  \]
  $B_\pi$ is a linear operator with fixed point $v_\pi$, meaning $B_\pi v_\pi = v_\pi$

- **Bellman Optimality Operator** $B_*$ operating on VF vector $v$:
  \[
  (B_* v)(s) = \max_a \{R^a_s + \gamma \sum_{s' \in S} P^a_{s,s'} \cdot v(s')\}
  \]
  $B_*$ is a non-linear operator with fixed point $v_*$, meaning $B_* v_* = v_*$

- Define a function $G$ mapping a VF $v$ to a deterministic “greedy” policy $G(v)$ as follows:
  \[
  G(v)(s) = \arg \max_a \{R^a_s + \gamma \sum_{s' \in S} P^a_{s,s'} \cdot v(s')\}
  \]
  $B_{G(v)} v = B_* v$ for any VF $v$ (Policy $G(v)$ achieves the max in $B_*$)
Both $B_\pi$ and $B_*$ are $\gamma$-contraction operators in $L^\infty$ norm, meaning:

For any two VFs $v_1$ and $v_2$,

$$\|B_\pi v_1 - B_\pi v_2\|_\infty \leq \gamma \|v_1 - v_2\|_\infty$$

$$\|B_* v_1 - B_* v_2\|_\infty \leq \gamma \|v_1 - v_2\|_\infty$$

So we can invoke Contraction Mapping Theorem to claim fixed point.

We use the notation $v_1 \leq v_2$ for any two VFs $v_1, v_2$ to mean:

$$v_1(s) \leq v_2(s) \text{ for all } s \in S$$

Also, both $B_\pi$ and $B_*$ are monotonic, meaning:

For any two VFs $v_1$ and $v_2$,

$$v_1 \leq v_2 \Rightarrow B_\pi v_1 \leq B_\pi v_2$$

$$v_1 \leq v_2 \Rightarrow B_* v_1 \leq B_* v_2$$
Policy Evaluation

- $B_\pi$ satisfies the conditions of Contraction Mapping Theorem
- $B_\pi$ has a unique fixed point $v_\pi$, meaning $B_\pi v_\pi = v_\pi$
- This is a succinct representation of Bellman Expectation Equation
- Starting with any VF $v$ and repeatedly applying $B_\pi$, we will reach $v_\pi$
  \[ \lim_{N \to \infty} B_\pi^N v = v_\pi \text{ for any VF } v \]
- This is a succinct representation of the Policy Evaluation Algorithm
Let $\pi_k$ and $v_{\pi_k}$ denote the Policy and the VF for the Policy in iteration $k$ of Policy Iteration.

Policy Improvement Step is: $\pi_{k+1} = G(v_{\pi_k})$, i.e. deterministic greedy.

Earlier we argued that $B_\pi v = B_{G(v)} v$ for any VF $v$. Therefore,

$$B_\pi v_{\pi_k} = B_{G(v_{\pi_k})} v_{\pi_k} = B_{\pi_{k+1}} v_{\pi_k}$$  \hspace{1cm} (1)

We also know from operator definitions that $B_\pi v \geq B_\pi v$ for all $\pi, v$

$$B_\pi v_{\pi_k} \geq B_{\pi_k} v_{\pi_k} = v_{\pi_k}$$  \hspace{1cm} (2)

Combining (1) and (2), we get:

$$B_{\pi_{k+1}} v_{\pi_k} \geq v_{\pi_k}$$

Monotonicity of $B_{\pi_{k+1}}$ implies

$$B_{\pi_{k+1}}^{N} v_{\pi_k} \geq \ldots \geq B_{\pi_{k+1}}^{2} v_{\pi_k} \geq B_{\pi_{k+1}} v_{\pi_k} \geq v_{\pi_k}$$

$$v_{\pi_{k+1}} = \lim_{N \to \infty} B_{\pi_{k+1}}^{N} v_{\pi_k} \geq v_{\pi_k}$$
We have shown that in iteration $k + 1$ of Policy Iteration, $v_{\pi_{k+1}} \geq v_{\pi_k}$.

If $v_{\pi_{k+1}} = v_{\pi_k}$, the above inequalities would hold as equalities.

So this would mean $B_* v_{\pi_k} = v_{\pi_k}$.

But $B_*$ has a unique fixed point $v_*$.

So this would mean $v_{\pi_k} = v_*$.

Thus, at each iteration, Policy Iteration either strictly improves the VF or achieves the optimal VF $v_*$. 
B_* satisfies the conditions of Contraction Mapping Theorem
B_* has a unique fixed point v_*, meaning B_*v_* = v_*
This is a succinct representation of Bellman Optimality Equation
Starting with any VF v and repeatedly applying B_*, we will reach v_*

\[ \lim_{N \to \infty} B_*^N v = v_* \text{ for any VF } v \]

This is a succinct representation of the Value Iteration Algorithm
Earlier we argued that $B_{G(v)}v = B_*v$ for any VF $v$. Therefore,

$$B_{G(v_*)}v_* = B_*v_*$$

But $v_*$ is the fixed point of $B_*$, meaning $B_*v_* = v_*$. Therefore,

$$B_{G(v_*)}v_* = v_*$$

But we know that $B_{G(v_*)}$ has a unique fixed point $v_{G(v_*)}$. Therefore,

$$v_* = v_{G(v_*)}$$

This says that simply following the deterministic greedy policy $G(v_*)$ (created from the Optimal VF $v_*$) in fact achieves the Optimal VF $v_*$. In other words, $G(v_*)$ is an Optimal (Deterministic) Policy