



Deep Reinforcement Learning

Professor Mohammad Hossein Rohban

Homework 7:

Value-Based Theory

By:

Ashkan Majidi
400109984



Spring 2025

Contents

1	Iteration Family	1
1.1	Positive Rewards	1
1.2	General Rewards.....	3
1.3	Policy Turn	4
2	Bellman or Bellwoman	8
2.1	Bellman Operators	8
2.2	Bellman Residuals.....	9

Grading

The grading will be based on the following criteria, with a total of 100 points:

Section	Points
Positive Rewards	15
General Rewards	10
Policy Turn	25
Bellman Operators	15
Bellman Residuals	35
Bonus 1: Writing your report in Latex	5
Bonus 2: Question 2.2.11	5

1 Iteration Family

Let $M = (S, A, R, P, \gamma)$ be a finite MDP with $|S| < \infty$, $|A| < \infty$, bounded rewards $|R(s, a)| \leq R_{\max} \forall (s, a)$, and discount factor $\gamma \in [0, 1)$. In this section, we will first explore an alternative proof approach for the value iteration algorithm, then we cover policy iteration which is discussed in the class more precisely.

1.1 Positive Rewards

Assume $R(s, a) \geq 0$ for all s, a .

- Derive an upper bound for the optimal k -step value function V_k^* .

Solution:

Using induction on k ($k - 1 \rightarrow k$) we show that $V_k^*(s) \leq R_{\max} \frac{1-\gamma^k}{1-\gamma}$.

base case ($k = 0$):

$$0 = V_0^*(s) \leq R_{\max} \frac{1-\gamma^0}{1-\gamma} = 0$$

Induction step:

The optimal k -step value function $V_k^*(s)$ satisfies:

$$V_k^*(s) = \max_a \sum_{s'} P(s' | s, a) (R(s, a, s') + \gamma V_{k-1}^*(s'))$$

To derive an upper bound, we observe that the reward $R(s, a, s')$ is bounded by R_{\max} and we know that $\sum_{s'} P(s'|s, a) = 1$, so:

$$\begin{aligned} V_k^*(s) &\leq \max_a \sum_{s'} P(s'|s, a) R_{\max} + \gamma \max_a \sum_{s'} P(s'|s, a) V_{k-1}^*(s') \\ &\leq R_{\max} + \gamma \max_a \sum_{s'} P(s'|s, a) V_{k-1}^*(s') \\ &\leq R_{\max} + \gamma R_{\max} \frac{1-\gamma^{k-1}}{1-\gamma} \\ &= R_{\max} \frac{1-\gamma^k}{1-\gamma} \end{aligned}$$

So proof done by induction.

- Prove V_k^* is non-decreasing in k . Giving a policy π such that:

$$V_{k+1}^\pi \geq V_k^*.$$

Use this to show convergence of Value Iteration to a solution satisfying the Bellman equation.

Solution:

To prove that V_k^* is non-decreasing in k , we will use induction ($k \rightarrow k + 1$).

Base case:

When $k = 0$, we have

$$V_0^*(s) = 0 \leq V_1^*(s) \quad (\text{since } V_1^*(s) \text{ is obviously non-negative}).$$

Induction step:

Assume that $V_k^*(s) \geq V_{k-1}^*(s)$ holds for some k . Now, for $k+1$, we have

$$\begin{aligned} V_{k+1}^*(s) &= \max_a \sum_{s'} P(s'|s, a) (R(s, a, s') + \gamma V_k^*(s')) \\ &\geq \max_a \sum_{s'} P(s'|s, a) (R(s, a, s') + \gamma V_{k-1}^*(s')) \\ &= V_k^*(s) \end{aligned}$$

Hence, $V_k^*(s)$ is non-decreasing in k .

Convergence of Value Iteration:

Value Iteration uses the following update rule:

$$V_{k+1}(s) = \max_a \sum_{s'} P(s'|s, a) (R(s, a, s') + \gamma V_k(s')) .$$

Since V_k^* is non-decreasing and Value Iteration computes V_k^* iteratively, it follows that

$$\lim_{k \rightarrow \infty} V_k^*(s) = V^*(s),$$

which satisfies the Bellman equation. Note that it happened because we assumed we had finite Action Space and State Space, so values should finally not change and converge to a constant value function.

Thus, Value Iteration converges to a solution satisfying the Bellman equation.

3. By taking the limit in the Bellman equation, prove that the V^* is optimal.

Solution:

To prove that V^* is optimal by taking the limit in the Bellman equation, we start with the Bellman equation for the optimal value function:

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

This equation holds for the optimal value function at each state s .

Since we have shown that $V_k^*(s)$ is non-decreasing in k and that $\lim_{k \rightarrow \infty} V_k^*(s) = V^*(s)$, we can take the limit of the Bellman equation as $k \rightarrow \infty$.

As $k \rightarrow \infty$, we have

$$V_{k+1}^*(s) = \max_a \sum_{s'} P(s'|s, a) (R(s, a, s') + \gamma V_k^*(s')) .$$

Substituting $V_k^*(s)$ with $V^*(s)$ in the limit, we get:

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

This is the Bellman equation, proving that $V^*(s)$ satisfies the Bellman equation.

Thus, by taking the limit in the Bellman equation, we have shown that V^* is optimal.

1.2 General Rewards

Remove the non-negativity constraint on $R(s, a)$. Assume no terminating states exist. Consider a new MDP defined by adding a constant reward r_0 to all rewards of the current MDP. That is, for all (s, a) , the new reward is:

$$\hat{R}(s, a) = R(s, a) + r_0$$

4. By deriving the optimal action and V_k^* in terms of the original MDP's values and r_0 , show that Value Iteration still converges to the optimal value function V^* (and optimal policy) of the original MDP even if rewards are negative. Also compute the new value V^* .

Solution:

We are given a new MDP with the reward function defined as:

$$\hat{R}(s, a) = R(s, a) + r_0,$$

where r_0 is a constant added to all rewards of the current MDP.

1. Value Function in Terms of the Original MDP:

The Bellman equation for the new value function $\hat{V}^*(s)$ is:

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

Substituting the new reward function:

$$\hat{V}^*(s) = \max_a \sum_{s'} P(s'|s, a) (R(s, a, s') + r_0 + \gamma \hat{V}^*(s')) .$$

Simplifying:

$$\hat{V}^*(s) = \max_a \sum_{s'} P(s'|s, a) (R(s, a, s') + \gamma \hat{V}^*(s')) + r_0 .$$

Thus, the new value function $\hat{V}^*(s)$ differs from the original value function $V^*(s)$ by a constant term r_0 .

2. Relating the New and Original Value Functions:

Let $V^*(s)$ be the original value function. Then:

$$\begin{aligned} \hat{V}^*(s) &= V^*(s) + \sum_{i=0}^{\infty} \gamma^i r_0 \\ &= V^*(s) + \frac{r_0}{1 - \gamma} . \end{aligned}$$

This shows that the new value function is simply the original value function shifted by a constant value, depending on the reward shift r_0 .

3. Optimal Policy:

The optimal action in the new MDP, $\hat{a}^*(s)$, is determined by:

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

Since the optimal action depends on maximizing the reward and value function sum, the optimal action will not change because the added constant r_0 affects all actions equally. Thus, the optimal

policy remains the same as in the original MDP.

4. Value Iteration Convergence:

In problems 1,2, and 3, we proved non-decreasing and convergence properties for MDP with non-negative rewards. Now we can choose r_0 such that all rewards become positive and have the desired properties. So since the value function is only shifted by a constant r_0 , the convergence process still works the same way. In the limit, Value Iteration will converge to $V^*(s)$, the optimal value function, in the original MDP, and the new value function will be shifted by the same constant $\frac{r_0}{1-\gamma}$.

Therefore, even when rewards are negative (as long as they are finite), Value Iteration will converge to the optimal value function of the original MDP.

5. Why is it necessary to assume the absence of a terminating state? Try to explain with a counterexample.

Solution:

The “constant–reward shift” argument used in Problem 4 assumes that every trajectory accumulates the same infinite geometric series $\sum_{t=0}^{\infty} \gamma^t r_0 = \frac{r_0}{1-\gamma}$. That only holds when no episode can terminate. If a terminating state exists, different policies can induce episodes of different (expected) lengths; the uniform shift then adds *different* amounts of return to different policies, so action preferences and the optimal policy can change.

Counter-example: Let the discount factor satisfy $0 < \gamma < 1$. Consider an MDP with:

- states s_0 (start) and s_T (terminal);
- actions at s_0 :
 - (a) QUIT: transition to s_T with reward 0;
 - (b) STAY: remain in s_0 with reward 0.
- s_T is absorbing and yields no further reward.

Original rewards. Both actions deliver total return 0; any policy is optimal.

After shifting rewards by a constant $r_0 = 1$.

$$Q_{\text{QUIT}}(s_0) = 1,$$

$$Q_{\text{STAY}}(s_0) = \sum_{t=0}^{\infty} \gamma^t 1 = \frac{1}{1-\gamma} > 1.$$

The STAY action now dominates, so the optimal policy changes purely because we introduced the terminating state. If instead $r_0 < 0$, the agent would prefer the shortest-possible episode (QUIT), again altering the optimum.

Hence the assumption “no terminating state” is essential. only then does a uniform reward shift translate every value by the same constant $\frac{r_0}{1-\gamma}$, leaving optimal actions and the convergence proof unchanged.

1.3 Policy Turn

In this part we want to dive into the mathematical proof of policy iteration.

6. Let π_k be the policy at iteration k . Prove the following:

$$V^{\pi_{k+1}}(s) \geq V^{\pi_k}(s) \quad \forall s \in S,$$

with strict inequality for at least one state unless π_k is already optimal. Use the definition of the greedy policy and explain why policy improvement leads to a better or equal value function.

Solution:

Let π_k be the policy obtained after the k -th iteration of policy iteration, and let V^{π_k} denote its state-value function. Define the next (greedy) policy by

$$\pi_{k+1}(s) \in \arg \max_{a \in A} Q^{\pi_k}(s, a), \quad Q^{\pi_k}(s, a) = \sum_{s'} P(s'|s, a) (R(s, a, s') + \gamma V^{\pi_k}(s')).$$

For every state s

$$Q^{\pi_k}(s, \pi_{k+1}(s)) \geq Q^{\pi_k}(s, \pi_k(s)) = V^{\pi_k}(s),$$

because $\pi_{k+1}(s)$ maximises $Q^{\pi_k}(s, \cdot)$.

Bellman operator notation:

$$[T^\pi V](s) = \sum_{s'} P(s'|s, \pi(s)) (R(s, \pi(s), s') + \gamma V(s')).$$

The inequality above can be written compactly as

$$T^{\pi_{k+1}} V^{\pi_k} \geq V^{\pi_k} \quad (\text{component-wise}).$$

As discussed in class, The operator $T^{\pi_{k+1}}$ is a γ -contraction in the ℓ_∞ norm, so iterative application to any starting vector converges **monotonically** to its unique fixed point $V^{\pi_{k+1}}$. Because contractions preserve order, we obtain

$$V^{\pi_{k+1}} = \lim_{n \rightarrow \infty} (T^{\pi_{k+1}})^n V^{\pi_{k+1}} \geq V^{\pi_k} \quad \forall s \in S.$$

Strict improvement unless π_k is optimal:

If $\pi_{k+1} \neq \pi_k$, then by construction there exists at least one state s where $\pi_{k+1}(s) \neq \pi_k(s)$ and hence $Q^{\pi_k}(s, \pi_{k+1}(s)) > V^{\pi_k}(s)$. The monotonicity of the contraction therefore yields $V^{\pi_{k+1}}(s) > V^{\pi_k}(s)$ for that state. Conversely, if $\pi_{k+1} = \pi_k$, no better action exists in any state, so π_k already satisfies the Bellman optimality equation and is therefore optimal.

$$V^{\pi_{k+1}}(s) \geq V^{\pi_k}(s) \quad \forall s \in S, \quad \text{with strict } > \text{ for at least one } s \text{ unless } \pi_k \text{ is optimal.}$$

This completes the proof that each policy-improvement step yields a value function that is component-wise no worse than the one before.

7. Prove that Policy Iteration always converges to the optimal policy in a finite MDP. Specifically, show that after a finite number of policy evaluations and improvements, the algorithm reaches a policy π^* that satisfies the Bellman optimality equation. You may use theorems discussed in class, but if a result was not proven, please provide a full justification.

Solution:

By Part 6, each improvement step either

$$V^{\pi_{k+1}} > V^{\pi_k} \quad (\text{in at least one state}) \quad \text{or} \quad \pi_{k+1} = \pi_k \quad (\text{already optimal}).$$

In a finite MDP, the number of deterministic policies is $|A|^{|S|} < \infty$. Because the value vectors are totally ordered under \geq and strictly increase whenever the policy changes, the algorithm cannot cycle.

Therefore policy iteration must halt after a finite number of evaluations/improvements at some policy π^* . At that point π^* is greedy with respect to V^{π^*} and thus satisfies the Bellman optimality equation:

$$V^{\pi^*}(s) = \max_a \sum_{s'} P(s'|s, a)(R(s, a, s') + \gamma V^{\pi^*}(s')), \quad \forall s \in S.$$

Hence π^* is an optimal policy and $V^{\pi^*} = V^*$.

Consequently, *policy iteration always converges in a finite number of steps to the optimal policy and value function in any finite MDP*.

8. Prove that Value Iteration and Policy Iteration both converge to the same optimal value function V^* , even if the policies may differ. How the policies are still optimal despite possible differences?

Solution:

For a finite MDP with discount factor $0 < \gamma < 1$, we showed that both satisfy the bellman optimality equations and because q-function is γ -contraction we know it have a unique answer so the two algorithms may return *different* optimal policies (e.g. tie-breaking can differ), yet all such policies achieve the same state-value vector V^* .

9. Compare and contrast the computational cost of one step of Policy Iteration (i.e., full Policy Evaluation + Policy Improvement) versus one iteration of Value Iteration.

Solution:

Let $S = |\mathcal{S}|$ and $A = |\mathcal{A}|$.

Value Iteration. (one sweep)

For each state we compute $\max_a \sum_{s'} P(s'|s, a)(R(s, a, s') + \gamma V(s'))$. This is $\mathcal{O}(SA)$ arithmetic operations and we should compute it S times so it's of $\mathcal{O}(S^2A)$, if it's done parallel, it could be faster because it have a linear part (beside max part) and also computation of each point of value function could be done separately. Additionally, it costs $\mathcal{O}(S)$ memory. Convergence typically requires many sweeps (on the order of $\log_{1/\gamma}(\frac{1}{\varepsilon})$) to reach an ε -optimal value.

Policy Iteration. (one “full” step)

- (a) *Policy Evaluation.* Solve $(I - \gamma P^\pi)V = R^\pi$.

* Exact (matrix inversion or LU): $\mathcal{O}(S^3)$.

* Iterative evaluation: m Bellman sweeps

$\mathcal{O}(mS^2)$ where typically $m > 1 + \log_{1/\gamma}(\frac{1}{\varepsilon})$.

- (b) *Policy Improvement.* One greedy sweep for each state: $\mathcal{O}(S^2A)$.

Thus a single Policy Iteration step costs $\mathcal{O}(S^3 + S^2A)$ (exact) or $\mathcal{O}(mS^2 + S^2A)$ (iterative).

In conclusion, one Value Iteration step is much cheaper than a full Policy Iteration step, but far more steps are required. In practice, Policy Iteration converges faster to the optimal policy, so it's more often used.

10. In the context of a (MDP) with an infinite horizon, when the discount factor $\gamma = 1$, analyze how both Value Iteration and Policy Iteration behave.

Solution:

With an infinite horizon and $\gamma = 1$ the cumulative return is

$$G_t = \sum_{i=0}^{\infty} R_{t+i}.$$

The standard theory assuming a γ -contraction ($\gamma < 1$) no longer applies.

- (a) *Value Iteration.* The Bellman operator $TV(s) = \max_a \sum_{s'} P(s'|s, a)(R + V(s'))$ has spectral radius 1, so it is *not* a contraction.

- If every policy reaches a terminal state in finite expected time (episodic tasks), Value Iteration still converges because values are bounded.
- Otherwise the sequence V_k can diverge to $+\infty$ or oscillate. A different criterion (e.g. *average reward*) is required.

- (b) *Policy Iteration.* Exact policy evaluation requires solving $(I - P^\pi)V = R^\pi$. The matrix $I - P^\pi$ is singular when $\gamma = 1$; a solution exists only up to an additive constant and, in general, is undefined for non-terminating tasks. Practical algorithms switch to *relative value iteration* or *average-reward policy iteration*, which subtract the long-run average reward to regain a contraction-like property.

With $\gamma = 1$ If the MDP is *proper* (all policies terminate with probability 1) the standard algorithms still converge. Otherwise one uses the average-reward or bias-optimality framework; vanilla Value Iteration/Policy Iteration may fail to converge or yield unbounded values.

2 Bellman or Bellwoman

[1] Recall that a value function is a $|S|$ -dimensional vector where $|S|$ is the number of states of the MDP. When we use the term V in these expressions as an “arbitrary value function”, we mean that V is an arbitrary $|S|$ -dimensional vector which need not be aligned with the definition of the MDP at all. On the other hand, V^π is a value function that is achieved by some policy π in the MDP. For example, say the MDP has 2 states and only negative immediate rewards. $V = [1, 1]$ would be a valid choice for V even though this value function can never be achieved by any policy π , but we can never have a $V^\pi = [1, 1]$. This distinction between V and V^π is important for this question and more broadly in reinforcement learning.

2.1 Bellman Operators

In the first part of this problem, we will explore some general and useful properties of the Bellman backup operator. We know that the Bellman backup operator B , defined below, is a contraction with the fixed point as V^* , the optimal value function of the MDP. The symbols have their usual meanings. γ is the discount factor and $0 \leq \gamma < 1$. In all parts, $\|v\| = \max_s |v(s)|$ is the infinity norm of the vector.

$$(BV)(s) = \max_a \left[r(s, a) + \gamma \sum_{s' \in S} p(s'|s, a)V(s') \right]$$

We also saw the contraction operator B^π with the fixed point V^π , which is the Bellman backup operator for a particular policy given below:

$$(B^\pi V)(s) = r(s, \pi(s)) + \gamma \sum_{s' \in S} p(s'|s, \pi(s))V(s')$$

In this case, we'll assume π is deterministic, but it doesn't have to be in general. You have seen that $\|BV - BV'\| \leq \gamma\|V - V'\|$ for two arbitrary value functions V and V' .

1. Show that the analogous inequality, $\|B^\pi V - B^\pi V'\| \leq \gamma\|V - V'\|$, holds.

Solution:

Fix an arbitrary state s and write

$$(B^\pi V - B^\pi V')(s) = \gamma \sum_{s'} p(s' | s, \pi(s)) (V(s') - V'(s')).$$

Because the transition probabilities are non-negative and sum to 1, Jensen's inequality for the absolute value gives

$$|(B^\pi V - B^\pi V')(s)| \leq \gamma \max_{s'} |V(s') - V'(s')| = \gamma \|V - V'\|.$$

Taking the maximum over s yields the desired bound, so B^π is a γ -contraction under $\|\cdot\|_\infty$.

2. Prove that the fixed point for B^π is unique. Recall that the fixed point is defined as V satisfying $V = B^\pi V$. You may assume that a fixed point exists.

Solution:

Assume V and W are two fixed points of B^π : $B^\pi V = V$ and $B^\pi W = W$. Then

$$\|V - W\| = \|B^\pi V - B^\pi W\| \leq \gamma \|V - W\|.$$

Since $0 \leq \gamma < 1$, the only solution of $\|V - W\| \leq \gamma \|V - W\|$ is $\|V - W\| = 0$, i.e. $V = W$. Hence the fixed point of B^π is unique.

3. Suppose that V and V' are vectors satisfying $V(s) \leq V'(s)$ for all s . Show that $B^\pi V(s) \leq B^\pi V'(s)$ for all s . Note: all of these inequalities are elementwise.

Solution:

$$B^\pi V'(s) - B^\pi V(s) = \sum_{s'} P(s'|s, \pi(s))(r(s, \pi(s), s') + \gamma(V'(s') - V(s'))) \geq 0$$

2.2 Bellman Residuals

We can extract a greedy policy π from an arbitrary value function V using the equation below:

$$\pi(s) = \arg \max_a \left[r(s, a) + \gamma \sum_{s' \in S} p(s'|s, a) V(s') \right]$$

It is often helpful to know what the performance will be if we extract a greedy policy from an arbitrary value function. To see this, we introduce the notion of a Bellman residual.

Define the Bellman residual to be $(BV - V)$ and the Bellman error magnitude to be $\|BV - V\|$.

4. For what value function V does the Bellman error magnitude $\|BV - V\|$ equal 0? Why?

Solution:

Because B is a γ -contraction ($0 < \gamma < 1$) it has a unique fixed point V^* satisfying $BV^* = V^*$. Hence $\|BV - V\| = 0$ iff V is that fixed point, i.e. $\|BV - V\| = 0 \iff V = V^*$.

5. Prove the following statements for an arbitrary value function V and any policy π .

$$\begin{aligned} \|V - V^\pi\| &\leq \frac{\|V - B^\pi V\|}{1 - \gamma} \\ \|V - V^*\| &\leq \frac{\|V - BV\|}{1 - \gamma} \end{aligned}$$

Solution:

(a) *Policy-specific bound.*

Subtract V from both sides of the Bellman equation for V^π :

$$\begin{aligned} V^\pi - V &= B^\pi V^\pi - B^\pi V + B^\pi V - V \\ &= \gamma P^\pi(V^\pi - V) + \underbrace{B^\pi V - V}_{\text{residual}} \end{aligned}$$

where P^π is the transition matrix under π . Re-arrange, take norms, and use $\|P^\pi x\| \leq \|x\|$:

$$\|V^\pi - V\| \leq \|B^\pi V - V\| + \gamma \|V^\pi - V\|, \quad \Rightarrow \quad \|V^\pi - V\| \leq \frac{\|B^\pi V - V\|}{1 - \gamma}.$$

(b) *Optimality bound.*

Identical algebra with B (whose fixed point is V^*) gives

$$\|V^* - V\| \leq \frac{\|BV - V\|}{1 - \gamma}.$$

6. Let V be an arbitrary value function and π be the greedy policy extracted from V . Let $\varepsilon = \|BV - V\|$ be the Bellman error magnitude for V . Prove the following for any state s .

$$V^\pi(s) \geq V^*(s) - \frac{2\varepsilon}{1 - \gamma}$$

Solution:

Let π be *greedy* w.r.t. V so that $BV = B^\pi V$, and let $\varepsilon = \|BV - V\|$. Applying the bounds of part 5:

$$\|V - V^*\| \leq \frac{\varepsilon}{1 - \gamma}, \quad \|V - V^\pi\| \leq \frac{\varepsilon}{1 - \gamma}.$$

Using the triangle inequality,

$$\|V^* - V^\pi\| \leq \|V^* - V\| + \|V - V^\pi\| \leq \frac{2\varepsilon}{1 - \gamma}.$$

Therefore, for every state s ,

$$V^\pi(s) \geq V^*(s) - \frac{2\varepsilon}{1 - \gamma},$$

showing the greedy policy π is at most $\frac{2\varepsilon}{1-\gamma}$ away from optimal in value terms.

7. Give an example real-world application or domain where having a lower bound on $V^\pi(s)$ would be useful.

Solution:

Safety-critical resource management.

Consider an autonomous drone that must return to its charging station before its battery is depleted. States encode the drone's location and remaining charge; rewards are -1 per time-step until docking (so higher value means faster, safer completion).

If we maintain an approximate value function V together with the lower bound

$$V^\pi(s) \geq V(s) - \frac{2\varepsilon}{1 - \gamma}, \quad \varepsilon = \|BV - V\|,$$

the controller can verify *before take-off* that the worst-case return from the current state still exceeds a safety threshold (e.g. "at least -50 " which corresponds to ≤ 50 expected steps). Should the lower bound drop below this threshold, the system can abort exploration or switch to a conservative fallback policy. Similar reasoning applies in finance (guaranteed minimum portfolio return), healthcare scheduling (minimum expected patient benefit), and robotics (guaranteed probability of mission success).

8. Suppose we have another value function V' and extract its greedy policy π' . $\|BV' - V'\| = \varepsilon = \|BV - V\|$. Does the above lower bound imply that $V^\pi(s) = V^{\pi'}(s)$ at any s ?

Solution:

No.

If $\varepsilon > 0$ we only know that $V^{\pi'}(s)$ and $V^\pi(s)$ both lie in the interval $[V^*(s) - \frac{2\varepsilon}{1-\gamma}, V^*(s)]$.

Equality $V^\pi(s) = V^{\pi'}(s)$ for all states occurs iff $\varepsilon = 0$, i.e. V' is already the optimal value function.

Say $V \leq V'$ if $\forall s, V(s) \leq V'(s)$.

What if our algorithm returns a V that satisfies $V^* \leq V$? I.e., it returns a value function that is better than the optimal value function of the MDP. Once again, remember that V can be any vector, not necessarily achievable in the MDP, but we would still like to bound the performance of V^π where π is extracted from said V . We will show that if this condition is met, then we can achieve an even tighter bound on policy performance.

9. Using the same notation and setup as part 5, if $V^* \leq V$, show the following holds for any state s . Recall that for all π , $V^\pi \leq V^*$ (why?)

$$V^\pi(s) \geq V^*(s) - \frac{\varepsilon}{1-\gamma}$$

Solution:

From part 5 we already have

$$\|V^\pi - V\| \leq \frac{\varepsilon}{1-\gamma}. \quad (\text{A})$$

For any state s ,

$$V^\pi(s) \geq V(s) - \|V^\pi - V\| \stackrel{(\text{A})}{\geq} V(s) - \frac{\varepsilon}{1-\gamma} \geq V^*(s) - \frac{\varepsilon}{1-\gamma}.$$

Hence

$$V^\pi(s) \geq V^*(s) - \frac{\varepsilon}{1-\gamma} \quad \forall s \in \mathcal{S}.$$

Because the “triangle–inequality loss” appears only once (not twice as in part 6), the gap to optimality tightens from 2ε to ε in the numerator.

Intuition: A useful way to interpret the results from parts (8) and (9) is based on the observation that a constant immediate reward of r at every time-step leads to an overall discounted reward of

$$r + \gamma r + \gamma^2 r + \dots = \frac{r}{1-\gamma}$$

Thus, the above results say that a state value function V with Bellman error magnitude ε yields a greedy policy whose reward per step (on average), differs from optimal by at most 2ε . So, if we develop an algorithm that reduces the Bellman residual, we’re also able to bound the performance of the policy extracted from the value function outputted by that algorithm, which is very useful!

10. It’s not easy to show that the condition $V^* \leq V$ holds because we often don’t know V^* of the MDP. Show that if $BV \leq V$ then $V^* \leq V$. Note that this sufficient condition is much easier to check and does not require knowledge of V^* .

Hint: Try to apply induction. What is $\lim_{n \rightarrow \infty} B^n V$?

Solution:

For any U, W , $U \leq W \Rightarrow BU \leq BW$. Hence $BV \leq V$ implies $B^2V \leq BV \leq V$, and inductively

$$B^{n+1}V \leq B^nV \leq \dots \leq V \quad \forall n \geq 1.$$

Because B is a γ -contraction, the sequence $\{B^n V\}_{n=0}^{\infty}$ converges to the unique fixed point V^* :

$$\lim_{n \rightarrow \infty} B^n V = V^*.$$

Every term in the sequence is $\leq V$. Order is preserved under pointwise limits, so

$$V^* = \lim_{n \rightarrow \infty} B^n V \leq V.$$

□

11. (Bonus) It is possible to make the bounds from parts (9) and (10) tighter. Let V be an arbitrary value function and π be the greedy policy extracted from V . Let $\varepsilon = \|BV - V\|$ be the Bellman error magnitude for V . Prove the following for any state s :

$$V^\pi(s) \geq V^*(s) - \frac{2\gamma\varepsilon}{1-\gamma}$$

Further, if $V^* \leq V$, prove for any state s

$$V^\pi(s) \geq V^*(s) - \frac{\gamma\varepsilon}{1-\gamma}$$

Solution:

First note that we know from former parts:

$$\|V^\pi - V\| \leq \frac{\varepsilon}{1-\gamma}, \quad \|V^* - V\| \leq \frac{\varepsilon}{1-\gamma}. \quad (1)$$

now we want to find upper bound for:

$$\begin{aligned} \|V^* - V^\pi\| &= \|V^* - BV + BV - v^\pi\| \\ &\leq \|V^* - B^\pi V\| + \|B^\pi V - V^\pi\| \\ &\leq \|V^* - B^\pi V\| + \|B^\pi V - B^\pi V^\pi\| \\ &\stackrel{\gamma\text{-contraction}}{\leq} \|V^* - B^\pi V\| + \gamma \|V - V^\pi\| \\ &\stackrel{(1)}{\leq} \|V^* - B^\pi V\| + \frac{\gamma\varepsilon}{1-\gamma} \end{aligned}$$

If we show $\|V^* - B^\pi V\| \leq \frac{\gamma\varepsilon}{1-\gamma}$ we get our wanted upper bound.

$$\begin{aligned} \frac{\gamma\varepsilon}{1-\gamma} &= \frac{\gamma \|BV - V\|}{1-\gamma} \\ &\geq \frac{\|(B)^2 V - BV\|}{1-\gamma} \\ &\geq \|V^* - BV\| \end{aligned}$$

And this yields our needed inequality.

For the next part:

$$\begin{aligned} V^* - V^\pi &\leq BV - V^\pi \\ &\leq \|B^\pi V - V^\pi\| \\ &\leq \frac{\gamma\varepsilon}{1-\gamma} \end{aligned}$$

References

- [1] Based on CS 234: Reinforcement Learning, Stanford University. Spring 2024.
- [2] Cover image designed by freepik
- [3] ChatGPT o3