Sharif University of Technology

Masoud Tahmasbi Fard

Student ID: 402200275

CE957: Reinforcement Learning

Assignment #2

April 16, 2024

Table of Contents

1 Policy Iteration

1.1 Contraction Mapping

$$\mathbf{U}(\mathbf{V}) = \mathbf{R} + \gamma \mathbf{P}\mathbf{V}$$

We first prove that the operator $\mathbf{U}(\mathbf{V})$ is a strict contraction. For every $U_1, U_2 \in \mathbb{R}^{|S|}$, we have:

$$\begin{aligned}\|\mathbf{U}(\mathbf{V}_1) - \mathbf{U}(\mathbf{V}_2)\|_\infty &= \gamma \|\mathbf{P}V_1 - \mathbf{P}V_2\|_\infty = \gamma \|\mathbf{P}(V_1 - V_2)\|_\infty \\ &\leq \gamma \|\mathbf{P}\|_\infty \|V_1 - V_2\|_\infty = \gamma \|V_1 - V_2\|_\infty\end{aligned}$$

where the second step follows by the following theorem:

$$\forall x \in \mathbb{R}^n, \|\mathbf{A}x\|_\infty \leq \|\mathbf{A}\|_\infty \|x\|_\infty$$

proof. Let $k = \arg\max_{1 \leq i \leq n} |(\mathbf{A}x)_i|$. Then,

$$\|\mathbf{A}x\|_\infty = |(\mathbf{A}x)_k| = \left| \sum_{j=1}^{n} a_{kj} x_j \right|.$$

By the triangle inequality, we have:

$$\left| \sum_{j=1}^{n} a_{kj} x_j \right| \leq \sum_{j=1}^{n} |a_{kj} x_j|.$$

Now, applying the definition of $\|\mathbf{A}\|_\infty$ and $\|x\|_\infty$, we get:

$$\sum_{j=1}^{n} |a_{kj} x_j| \leq \max_{1 \leq i \leq n} \sum_{j=1}^{n} |a_{ij}| \cdot \max_{1 \leq j \leq n} |x_j| = \|\mathbf{A}\|_\infty \|x\|_\infty.$$

Therefore,

$$\|\mathbf{A}x\|_\infty = \left| \sum_{j=1}^{n} a_{kj} x_j \right| \leq \|\mathbf{A}\|_\infty \|x\|_\infty.$$

Thus as $0 < \gamma < 1$, we conclude that $\mathbf{U}(\mathbf{V})$ is a strict contraction on $\mathbb{R}^{|S|}$.

1.2 Convergence

The following properties hold for a contraction mapping:

- Only has 1 fixed point (the point reach if apply a contraction operator many times)

  - If had two, then would not get closer when apply contraction function, violating definition of contraction

- When apply contraction function to any argument, value must get closer to fixed point
  - Fixed point doesnt move
  - Repeated function applications yield fixed point

Now we proceed to prove the convergence of the policy evaluation. Note that V^π is a fixed point of $\mathbf{U}$. In addition, according to the previous section, $\mathbf{U}$ is a contraction mapping. Therefore,

$$\|\mathbf{U}^n(V) - V^\pi\|_\infty = \left\|\mathbf{U}(\mathbf{U}^{n-1}(V)) - \mathbf{U}(V^\pi)\right\|_\infty \leq \gamma \left\|\mathbf{U}^{n-1}(V) - V^\pi\right\|_\infty \leq \cdots \leq \gamma^n \left\|\mathbf{U}^0(V) - V^\pi\right\|_\infty .$$

Let $n \to \infty$, and we have $\|\mathbf{U}^n(V) - V^\pi\|_\infty \to 0$. Thus $\lim_{n\to\infty} \mathbf{U}^n(V) = V^\pi$.

1.3 Error Bound

We have $\left\|\mathbf{U}^k(V) - \mathbf{U}^{k-1}(V)\right\|_\infty \leq \epsilon$. Then using the triangle inequality and the fact that $\mathbf{U}^k(V) = \mathbf{U}(\mathbf{U}^{k-1}(V))$ we get,

$$\begin{aligned}
\left\|\mathbf{U}^{k-1}(V) - V^\pi\right\|_\infty &\leq \left\|\mathbf{U}^{k-1}(V) - \mathbf{U}^k(V)\right\|_\infty + \left\|\mathbf{U}^k(V) - V^\pi\right\|_\infty \\
&= \left\|\mathbf{U}^{k-1}(V) - \mathbf{U}^k(V)\right\|_\infty + \left\|\mathbf{U}(\mathbf{U}^{k-1}(V)) - \mathbf{U}(V^\pi)\right\|_\infty \\
&\leq \left\|\mathbf{U}^{k-1}(V) - \mathbf{U}^k(V)\right\|_\infty + \gamma \left\|\mathbf{U}^{k-1}(V) - V^\pi\right\|_\infty = \epsilon + \gamma \left\|\mathbf{U}^{k-1}(V) - V^\pi\right\|_\infty
\end{aligned}$$

and so $\left\|\mathbf{U}^{k-1}(V) - V^\pi\right\|_\infty \leq \frac{\epsilon}{1-\gamma}$. This finally allows us to conclude that

$$\left\|\mathbf{U}^k(V) - V^\pi\right\|_\infty = \left\|\mathbf{U}(\mathbf{U}^{k-1}(V)) - \mathbf{U}(V^\pi)\right\|_\infty \leq \gamma \left\|\mathbf{U}^{k-1}(V) - V^\pi\right\|_\infty \leq \frac{\epsilon\gamma}{1-\gamma} \leq \frac{\epsilon}{1-\gamma}$$

2 Gridworld

2.1 Every Visit Monte Carlo

Every visit Monte Carlo updates values by averaging returns from all visits to a state. Values of the states after the first episode, using every visit Monte Carlo will be:

-10	-10	-10	0	0
0	-10	-10	0	0
0	-10	0	0	0
0	-10	0	10	10
0	-10	5	5	0

2.2 First Visit Monte Carlo

First visit Monte Carlo updates values only after the first visit to a state pair in an episode. Values of the states after the first episode, using first visit Monte Carlo will be:

-10	-10	-10	0	0
0	-10	-10	0	0
0	-10	0	0	0
0	-10	0	10	10
0	-10	0	0	0

3 Optimal Policy

3.1 Initial Distribution

$$
\begin{aligned}
v_M^\pi(s) =& \mathbb{E}^\pi \left[\sum_{t=0}^{\infty} \gamma^t r_t \mid S_0 = s \right] \\
=& \sum_{k=0}^{\infty} \gamma^k \mathbb{E}^\pi \left[r_{t+k} \mid S_0 = s \right] \\
=& \sum_{a \in \mathcal{A}} \pi(s,a) \left(r(s,a) + \sum_{k=1}^{\infty} \gamma^k \mathbb{E}^\pi \left[r_{t+k} \mid S_0 = s \right] \right) \\
=& \sum_{a \in \mathcal{A}} \pi(s,a) \left[r(s,a) + \sum_{s' \in \mathcal{S}} P(s,a,s') \sum_{a' \in \mathcal{A}} \pi(s',a') \left(\gamma^1 r(s',a') + \sum_{k=2}^{\infty} \gamma^k \mathbb{E}^\pi \left[r_{t+k} \mid S_0 = s \right] \right) \right] \\
=& \gamma^0 \sum_{a \in \mathcal{A}} \pi(s,a) r(s,a) + \gamma^1 \sum_{a \in \mathcal{A}} \pi(s,a) \sum_{s' \in \mathcal{S}} P(s,a,s') \sum_{a' \in \mathcal{A}} \pi(s',a') r(s',a') \\
& + \gamma^2 \sum_{a \in \mathcal{A}} \pi(s,a) \sum_{s' \in \mathcal{S}} P(s,a,s') \sum_{a' \in \mathcal{A}} \pi(s',a') \sum_{s'' \in \mathcal{S}} P(s',a',s'') \sum_{a'' \in \mathcal{A}} \pi(s'',a'') r(s'',a'') + \dots \\
=& \gamma^0 \sum_{a \in \mathcal{A}} \pi(s,a) r_0(s,a) + \gamma^1 \sum_{a \in \mathcal{A}} \pi(s,a) \sum_{s' \in \mathcal{S}} P_0(s,a,s') \sum_{a' \in \mathcal{A}} \pi(s',a') r_0(s',a') \\
& + \gamma^2 \sum_{a \in \mathcal{A}} \pi(s,a) \sum_{s' \in \mathcal{S}} P_0(s,a,s') \sum_{a' \in \mathcal{A}} \pi(s',a') \sum_{s'' \in \mathcal{S}} P_0(s',a',s'') \sum_{a'' \in \mathcal{A}} \pi(s'',a'') r'(s'',a'') + \dots \\
=& \sum_{a \in \mathcal{A}} \pi(s,a) \left[r_0(s,a) + \sum_{s' \in \mathcal{S}} P_0(s,a,s') \sum_{a' \in \mathcal{A}} \pi(s',a') \left(\gamma^1 r(s',a') + \sum_{k=2}^{\infty} \gamma^k \mathbb{E}^\pi \left[r_{t+k} \mid S_0 = s \right] \right) \right] \\
=& \sum_{a \in \mathcal{A}} \pi(s,a) \left(r_0(s,a) + \sum_{k=1}^{\infty} \gamma^k \mathbb{E}^\pi \left[r_{t+k} \mid S_0 = s \right] \right) \\
=& v_{M_0}^\pi(s)
\end{aligned}
$$

3.2 Multiplying the Rewards!

Let π^* be an optimal policy for MDP M. Now, let's define a new MDP $M' = (S, A, P, R', \gamma)$, where R' represents the rewards of MDP M multiplied by a positive scalar $\alpha > 0$, i.e., $R'(s,a,s') = \alpha \cdot R(s,a,s')$.

We want to show that π^* is also an optimal policy for MDP M'. Let $J(\pi)$ denote the expected return of policy π in MDP M, and let $J'(\pi)$ denote the expected return of policy π in MDP M'.

Since π^* is optimal for MDP M, we have:

$$J(\pi^*) \geq J(\pi) \quad \forall \pi$$

Now, let's consider $J'(\pi^*)$ for MDP M':

$$\begin{aligned} J'(\pi^*) =& \sum_{s \in S} d_{\pi^*}(s) \sum_{a \in A} \pi^*(a|s) \sum_{s' \in S} P(s'|s,a) \alpha R(s,a,s') \\ =& \alpha \sum_{s \in S} d_{\pi^*}(s) \sum_{a \in A} \pi^*(a|s) \sum_{s' \in S} P(s'|s,a) R(s,a,s') \\ =& \alpha \sum_{s \in S} d_{\pi^*}(s) \sum_{a \in A} \pi^*(a|s) \sum_{s' \in S} P(s'|s,a) R'(s,a,s') \\ =& \alpha J(\pi^*) \\ \implies & \alpha J(\pi^*) \geq \alpha J(\pi) \quad \forall \pi \rightarrow J'(\pi^*) \geq J'(\pi) \quad \forall \pi \end{aligned}$$

Since $\alpha > 0$ and $\gamma < 1$, multiplying the rewards by α doesn't affect the policy's optimality because the relative ordering of policies' expected returns remains the same. Therefore, π^* remains an optimal policy for MDP M', as required to be proved.

3.3 Adding to the Rewards!

To prove that adding a positive constant to all rewards of a finite MDP can change which policies are optimal, consider a finite Markov Decision Process (MDP) $M = (S, A, P, R, \gamma)$ where:
- S is the set of states. - A is the set of actions. - P is the state transition probability function. - R is the reward function with bounded rewards. - γ is the discount factor, such that $0 \leq \gamma < 1$.
Let π^* be an optimal policy for MDP M, achieving the maximum expected return among all policies.

Now, let's define a new MDP $M' = (S, A, P, R', \gamma)$, where R' represents the rewards of MDP M with a positive constant c added to each reward, i.e., $R'(s,a,s') = R(s,a,s') + c$.

We want to show that π^* may not be an optimal policy for MDP M'. Let $J(\pi)$ denote the expected return of policy π in MDP M, and let $J'(\pi)$ denote the expected return of policy π in MDP M'.

Since π^* is optimal for MDP M, we have:

$$J(\pi^*) \geq J(\pi) \quad \forall \pi$$

Now, let's consider $J'(\pi^*)$ for MDP M':

$$\begin{aligned} J'(\pi^*) &= \sum_{s \in S} d_{\pi^*}(s) \sum_{a \in A} \pi^*(a|s) \sum_{s' \in S} P(s'|s,a)(R(s,a,s') + c) \\ &= \sum_{s \in S} d_{\pi^*}(s) \sum_{a \in A} \pi^*(a|s) \sum_{s' \in S} P(s'|s,a)R(s,a,s') + c \\ &= J(\pi^*) + c \end{aligned}$$

Since $c > 0$ and $\gamma < 1$, adding a positive constant to all rewards shifts the relative ordering of policies' expected returns. Therefore, π^* may not necessarily be an optimal policy for MDP M', which demonstrates that adding a positive constant to all rewards can change which policies are optimal. As an example, consider an MDP with three states, s_0, s_1, and s_2, where three actions are possible in each state: transitioning to the other states or taking no action. In the initial scenario, each action incurs a -1 reward, and starting from s_0, there is a +2 reward at s_2. Consequently, the optimal action in s_0 would be to move directly to s_2. However, in an alternate scenario where we add +2 to all rewards, the optimal action shifts to transitioning from s_0 to s_1 and then to s_2.

3.4 Adding to the Rewards when there is no end!

False. Adding a constant positive value to the rewards can indeed change the optimal policy, even in scenarios without terminating states. Consider the previous example: in the initial setup where each action incurs a -1 reward, the optimal action is to choose the "no action" option. However, upon adding +2 to each reward, every transition yields a positive reward, altering the optimal policy.

3.5 New MDP!

For all $s \in \mathcal{S}$ and $a \in \mathcal{A}$, $r(s,a) \geq r'(s,a)$, where r' denotes the reward function of M'. This inequality arises because $r(s,a) = r'(s,a)$ when the optimal policy π^* selects action a in state s,

and $r(s,a) > r'(s,a)$ otherwise.

- **Case 1:** $r(s,a) = r'(s,a)$

$$v_M^{\pi^*}(s) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k} \mid S_t = s, \pi^*, M\right] = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k} \mid S_t = s, \pi^*, M'\right] = v_{M'}^{\pi^*}(s) \tag{1}$$

- **Case 2:** $r(s,a) > r'(s,a)$

$$\begin{aligned} v_M^{\pi}(s) &= \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k} \mid S_t = s, \pi, M\right] = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r\left(S_{t+k}, A_{t+k}\right) \mid S_t = s, \pi, M\right] \\ &\geq \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r'\left(S_{t+k}, A_{t+k}\right) \mid S_t = s, \pi, M'\right] \\ &= v_{M'}^{\pi}(s) \end{aligned} \tag{2}$$

Also:

$$v_M^{\pi^*}(s) \geq v_M^{\pi}(s) \tag{3}$$

Using (1), (2), and (3):

$$v_{M'}^{\pi^*}(s) = v_M^{\pi^*}(s) \geq v_M^{\pi}(s) \geq V_{M'}^{\pi}(s),$$

Thus $\pi^* \geq \pi$ and π^* is an optimal policy for M'.

4 Bellman Equation

4.1 Bellman Inverse!

Given equation is incorrect. Let's consider an MDP where there exists a state s_0 that cannot be reached from any other state, yet $v^\pi(s_0) \neq 0$. In this scenario, applying the new equation $v^\pi(s_0) = 0$ would imply that every $P(s, a, s_0)$ term is zero, which contradicts the actual behavior of the MDP. Therefore, this equation cannot be valid.

4.2 Equivalent Expressions!

$$
\begin{aligned}
v_t^\pi(s) =& \mathbb{E}\left[G_t \mid S_t = s, \pi\right] = \sum_{k=0}^{\infty} \mathbb{E}\left[\gamma^k R_{t+k} \mid S_t = s, \pi\right] \\
=& \sum_{a\in\mathcal{A}} \pi(s,a) \left(R(s,a) + \sum_{k=1}^{\infty} \gamma^k \mathbb{E}\left[R_{t+k} \mid S_t = s, \pi\right] \right) \\
=& \sum_{a\in\mathcal{A}} \pi(s,a) \left[R(s,a) + \sum_{s'\in\mathcal{S}} P\left(s,a,s'\right) \sum_{a'\in\mathcal{A}} \pi\left(s',a'\right) \left(\gamma^1 R\left(s',a'\right) + \sum_{k=2}^{\infty} \gamma^k \mathbb{E}\left[R_{t+k} \mid S_t = s, \pi\right] \right) \right] \\
=& \gamma^0 \sum_{a\in\mathcal{A}} \pi(s,a) R(s,a) + \gamma^1 \sum_{a\in\mathcal{A}} \pi(s,a) \sum_{s'\in\mathcal{S}} P\left(s,a,s'\right) \sum_{a'\in\mathcal{A}} \pi\left(s',a'\right) R\left(s',a'\right) \\
& + \gamma^2 \sum_{a\in\mathcal{A}} \pi(s,a) \sum_{s'\in\mathcal{S}} P\left(s,a,s'\right) \sum_{a'\in\mathcal{A}} \pi\left(s',a'\right) \sum_{s''\in\mathcal{S}} P\left(s',a',s''\right) \sum_{a''\in\mathcal{A}} \pi\left(s'',a''\right) R\left(s'',a''\right) + \dots \\
=& \gamma^0 \sum_{a\in\mathcal{A}} P\left(A_0 = a \mid S_0 = s\right) R(s,a) \\
& + \gamma^1 \sum_{a\in\mathcal{A}} P\left(A_0 = a \mid S_0 = s\right) \sum_{s'\in\mathcal{S}} P\left(S_1 = s' \mid A_0 = a, S_0 = s\right) \sum_{a'\in\mathcal{A}} P\left(A_1 = a' \mid S_1 = s'\right) R\left(s',a'\right) + . \\
=& \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_t \mid S_0 = s, \pi \right] = \mathbb{E}\left[G \mid S_0 = s, \pi\right] = v_0^\pi(s).
\end{aligned}
$$

4.3 Negative Rewards!

$$
\begin{aligned}
v^{\pi}(S_t) &= \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k} \mid S_t = s_t, \pi\right] = \mathbb{E}\left[\sum_{k=0}^{\infty} R_{t+k} \mid S_t = s_t, \pi\right] \\
&= \sum_{k=0}^{\infty} \mathbb{E}\left[R_{t+k} \mid S_t = s_t, \pi\right] = \sum_{k=0}^{\infty} \mathbb{E}\left[R_{t+k} \mid \pi\right] \\
&= \mathbb{E}\left[R_t \mid \pi^{\star}\right] + \sum_{k=0}^{\infty} \mathbb{E}\left[R_{t+k+1} \mid \pi\right] = \mathbb{E}\left[R_t \mid \pi\right] + \sum_{k=0}^{\infty} \mathbb{E}\left[R_{t+k+1} \mid S_{t+1} = s_{t+1}, \pi\right] \\
&= \mathbb{E}\left[R_t \mid \pi\right] + v^{\pi}(s_{t+1}) = \mathbb{E}\left[R_t \mid \pi\right] + v^{\pi}(S_{t+1}) < v^{\pi}(S_{t+1})
\end{aligned}
$$

The last inequality holds because $R_t < 0$. Therefore the given sequence is strictly increasing.

5 Temporal Difference and Eligibility Traces

5.1 Value Updates

Given a sequence of states $S_t, S_{t+1}, \ldots, S_{t+n}$, the update rule for TD(n) is defined as:

$$V(S_t) \leftarrow V(S_t) + \alpha \left(G_t^{(n)} - V(S_t) \right)$$

where $G_t^{(n)} = R_{t+1} + \gamma R_{t+2} + \ldots + \gamma^{n-1} R_{t+n} + \gamma^n V(S_{t+n})$ Therefore:

- $n = 1$: Only the value of the state E would be updated.
- $n = 2$: Values for state E and D would be updated.
- $n = 3$: Values for state E, D, and C would be updated.
- $n = 4$: Values for state E, D, C, and B would be updated.
- $n = 4$: Values for all states would be updated.

5.2 Effect of α

The α parameter denotes the learning rate. It determines the step size of the updates to the value function or policy parameters during learning. A higher value of α implies that the agent updates its estimates more aggressively based on new information, while a lower value means that the agent updates its estimates more conservatively.

Now, regarding the graph comparing the average RMS error for different values of n and α, the results suggest that there is an optimal value for α for each n. Initially, increasing the value of α tends to decrease the RMS error, indicating faster learning. However, beyond a certain threshold, further increasing α leads to instability or overshooting, causing the RMS error to start growing again.

As the value of n increases, the optimal value for α tends to be smaller. This trend can be attributed to the fact that larger n values involve longer time horizons for computing the returns.

In such cases, larger α values can lead to overly aggressive updates, causing the learning process to become unstable. Therefore, smaller α values are preferred for larger n to ensure more stable and accurate learning over longer time horizons.

5.3 Reducing the RMSE!

Among the options provided, (b) increasing the number of episodes over which error is calculated would likely result in a decrease in the RMS errors shown in the graphs.

By increasing the number of episodes, the algorithm has more opportunities to explore and learn about the environment. This increased exposure to different scenarios allows the agent to refine its value function estimates, leading to potentially lower RMS errors. Since graphs are averaged over the episode, more episodes provide a better representation of the true performance of the learning algorithm, leading to more reliable estimates of the RMS error. Therefore, increasing the number of episodes tends to improve the learning process and decrease RMS errors.

5.4 Eligibility Traces

Maximum increase in eligibility traces occurs when a state is seen through the episode. In this case:

$$e_t(s) = \gamma\lambda e_{t-1}(s) + 1$$

When the eligibility trace reaches its maximum, $e_t(s) = e_{t-1}(s)$, resulting in $e_t(s) = e_{t-1}(s) = \frac{1}{1-\gamma\lambda} = \frac{1}{1-0.8\times0.25} = 1.25$.