Reinforcement Learning

Reinforcement Learning
- Markov Decision Process
  - Rewards
  - Policy
  - State Value Function
  - Q-function (State-Action Value)
- Bellman Equations
  - Bellman Equations for State Value
  - Bellman Equation for Q-function
- Policy Evaluation
  - Solution in Finite State Space
  - Bellman Operator
- Bellman Optimality Equations
  - Bellman Optimality Criterion
  - Solving Optimal State Values
  - Optimal Q-function
- Dynamic Programming
  - Value Iteration
  - Policy Iteration
  - Generalized Policy Iteration
- Generalization
- Appendix
  - Cascade of Expectations
  - Row Stochastic Matrix

Notation:

- upercase (e.g. R_t): random variable
- lowercase (e.g. s_t): instance of random variable
- bold straight (e.g. $\mathbf{v}$): determineistic vector

Markov Decision Process

A Markov decision process (MDP) consists of

- $\mathcal{S}$: set of states.
- $\mathcal{A}$: set of actions.
- $p(\cdot \mid s, a)$: state transition probability. i.e. the probability distribution of the new state given the current state s and current action a.
- $\gamma \in (0, 1)$: discount factor.
- $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$: reward function. bounded.

In the following, we consider stationary MDP. i.e. Both the station transition probability and reward function are time-independent.

Rewards

Given the state s_t at time t, the agent takes action a_t, which leads to

- the new state s_{t+1}, sampled from $p(\cdot \mid s_t, a_t)$
- and the reward r_t, determined by the reward function $r_t = r(s_t, a_t)$.

Continue taking actions $a_{t+1}, a_{t+2}, \ldots$, we get a state-action-reward trajectory

$$s_t \xrightarrow[r_t]{a_t} s_{t+1} \xrightarrow[r_{t+1}]{a_{t+1}} s_{t+2} \xrightarrow[r_{t+2}]{a_{t+2}} s_{t+3} \cdots$$

$\rightarrow$ Intuitive goal: Maximize the sum of all r_t.

In the stochastic setting, all states $S_t, S_{t+1}, \ldots$ are random. Since the agent takes action based on current state, the action sequence is also random. Similary, the reward sequence is also random. $\implies$ stochastic state-action-reward trajectory:

$$S_t \xrightarrow[R_t]{A_t} S_{t+1} \xrightarrow[R_{t+1}]{A_{t+1}} S_{t+2} \xrightarrow[R_{t+2}]{A_{t+2}} S_{t+3} \cdots$$

$\rightarrow$ Intuitive goal: Maximize $\mathbb{E}[\text{sum of all } R_t]$.

The total (discounted) reward is defined as the sum of (discounted) rewards starting from S_t

$$G_t = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \ldots \tag{1}$$
$$= \sum_{k=0}^{\infty} \gamma^k R_{t+k} \tag{2}$$

Remarks:

- G_t is a random quantity as the state transition is stochastic. Executing the same policy from s_t again will yield a different g_t.
- The discount factor γ serves two purposes:
  1. it ensures that the infinite sum is defined. Recall from math: If $\sum_{n=1}^{\infty} a_n$ converges absolutely and $\{b_n\}$ is bounded, then $\sum_{n=1}^{\infty} a_n b_n$ converges absolutely. Here, we have a geometric series which converges absolutely and a bounded sequence of rewards.
  2. it puts more weight on short-term rewards over long-term rewards. e.g. In finance, 100 dollar today is worth more than 100 dollar next year.

Recursive structure: The total reward G_t comprises immediate reward R_t plus a discounted future reward G_{t+1}.

$$G_t = R_t + \gamma G_{t+1} \tag{3}$$

Policy

In MDP, the agent performs actions determined by current state, according to a ***policy***

$$\pi : \mathcal{S} \to \mathcal{A}, s \mapsto a = \pi(s)$$

Remark: The policy defined here is **time-invariant** and **deterministic**, i.e. For a certain state s, the agent takes the **same** action **whenever** he arrives at s.

State Value Function

For a certain policy π, the corresponding ***state value function*** is mapping $v_\pi : \mathcal{S} \to \mathbb{R}, s \mapsto v_\pi(s)$. $v_\pi(s)$ is called ***state value***, defined as the expected total reward by executing policy π starting from state s.

$$v_\pi(s) = \mathbb{E}\left[G_t \mid S_t = s\right], \quad \forall s \in \mathcal{S} \tag{4}$$

Remarks:

- The expectation is take over $\{S_k\}_{k \geq t+1}$. The random variable G_t depends implicity on policy π as it represents the total reward by executing π.
- State value $v_\pi(s)$ is defined for **ALL** possible states in space $\mathcal{S}$.
- For a fixed policy π, $v_\pi(s)$ quantifies the goodness of state s.
- For a fixed initial state s, $v_\pi(s)$ quantifies the goodness of policy π.
- For a stationanry MDP, $v_\pi(s)$ is independent of t. i.e. The state value of s remains the same regardless of when the agents arrives at s. Hence, we can assume without loss of generality that the agent arrived at s at $t = 0$. The state value then becomes

$$v_\pi(s) = \mathbb{E}\left[G_0 \mid S_0 = s\right]$$

Q-function (State-Action Value)

The ***Q-function*** (or ***state action value***) for a certain policy π as follows

$$q_\pi(s, a) = \mathbb{E}\left[G_t \mid S_t = s, A_t = a\right], \quad \forall s \in \mathcal{S}, \forall a \in \mathcal{A} \tag{5}$$

Relation between Q-function and state value for the same π:

- Interpretation: $q_\pi(s, a)$ represents the total reward of taking action a at initial state s and then following a policy π. vs. $v_\pi(s)$ represents the total reward of following π from s onward.
- Compute $v_\pi(s)$ from $q_\pi(s, a)$: simply let $a = \pi(s)$, i.e.

$$v_\pi(s) = q_\pi(s, a)\Big|_{a=\pi(s)} \tag{6}$$

- Compute $q_\pi(s, a)$ from $v_\pi(s)$: use Bellman equation (will be proved later)

$$q_\pi(s, a) = r(s, a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)}[v_\pi(s')] \tag{7}$$

$$= r(s, a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s, a) v_\pi(s') \qquad \text{if } \mathcal{S} \text{ is finite} \tag{8}$$

Bellman Equations

Bellman Equations for State Value

Bellman Equations: The state values have the recursive structure

$$v_\pi(s) = r(s, \pi(s)) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,\pi(s))}[v_\pi(s')], \quad \forall s \in \mathcal{S} \tag{9}$$

Remarks:

- The expected total reward is the sum of immediate reward plus the (discounted) expected future reward.
  - $r(s, \pi(s))$: immediate reward at current state s by executing the policy π

    - $v_\pi(s')$: future reward from the next state s' by executing the policy π. $\mathbb{E}_{s' \sim p(\cdot \mid s, \pi(s))}[v_\pi(s')]$ is the average future reward over all possible s'.
- The current action $\pi(s)$ influences the immediate reward and the probability distribution of the next state s'.
- Bellman equations hold for all $s \in \mathcal{S}$. If $\mathcal{S}$ is finite, there are $|\mathcal{S}|$ Bellman equations.

Proof: Without loss of generality, assume $t = 0$. Using $G_0 = R_0 + \gamma G_1$, we get

$$\begin{aligned} v_\pi(s) &= \mathbb{E}_{S_1, S_2, \dots}[G_0 \mid S_0 = s] \\ &= \mathbb{E}_{S_1, S_2, \dots}[R_0 + \gamma G_1 \mid S_0 = s] \\ &= \mathbb{E}_{S_1, S_2, \dots}[R_0 \mid S_0 = s] + \gamma \mathbb{E}_{S_1, S_2, \dots}[G_1 \mid S_0 = s] \\ &= r(s, \pi(s)) + \gamma \mathbb{E}_{S_1, S_2, \dots}[G_1 \mid S_0 = s] \end{aligned}$$

Using the law of total expectation and the markov properties (c.f. Appendix), we get

$$v_\pi(s) = r(s, \pi(s)) + \gamma \mathbb{E}_{S_1 \sim p(\cdot \mid s, \pi(s))} \left[\underbrace{\mathbb{E}_{S_2, S_3, \dots}[G_1 \mid S_1 = s_1]}_{v_\pi(s_1)} \right]$$

The underbraced term is the expected total reward by executing policy π starting from state s_1 which is by definition exactly the state value of $s_1 s_1$. Hence, we conclude. $\square$

For finite state space, the expected future reward in Bellman equation can be expressed in a sum. The Bellman equations become

$$v_\pi(s) = r(s, \pi(s)) + \gamma \sum_{s'} p(s' \mid s, \pi(s)) \cdot v_\pi(s'), \quad \forall s \in \mathcal{S} \tag{10}$$

Example: We model the state of a stock market as $\mathcal{S} = \{\text{bull}, \text{bear}, \text{flat}\}$. The agent uses some investment policy π (not detailed here) which yields the following state transition probability (Note that each row sums to 1).

$p(s' \mid s, \pi(s))$	$s' = \text{bull}$	$s' = \text{bear}$	$s' = \text{flat}$
$s = \text{bull}$	0.8	0.1	0.1
$s = \text{bear}$	0.1	0.7	0.2
$s = \text{flat}$	0	0.1	0.9

Suppose that immediate rewards under this policy are

$$r(\text{bull}, \pi(\text{bull})) = 8, \quad r(\text{bear}, \pi(\text{bear})) = -9, \quad r(\text{flat}, \pi(\text{flat})) = 2,$$

Then, the three Bellman equations are

$$\begin{aligned} v_\pi(\text{bull}) &= 8 + \gamma \left[0.8 v_\pi(\text{bull}) + 0.1 v_\pi(\text{bear}) + 0.1 v_\pi(\text{flat}) \right] \\ v_\pi(\text{bear}) &= -9 + \gamma \left[0.1 v_\pi(\text{bull}) + 0.7 v_\pi(\text{bear}) + 0.2 v_\pi(\text{flat}) \right] \\ v_\pi(\text{flat}) &= 2 + \gamma \left[0 \cdot v_\pi(\text{bull}) + 0.1 v_\pi(\text{bear}) + 0.9 v_\pi(\text{flat}) \right] \end{aligned}$$

Reformulation in vector form:

$$\begin{bmatrix} v_\pi(\text{bull}) \\ v_\pi(\text{bear}) \\ v_\pi(\text{flat}) \end{bmatrix} = \begin{bmatrix} 8 \\ -9 \\ 2 \end{bmatrix} + \gamma \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.7 & 0.2 \\ 0 & 0.1 & 0.9 \end{bmatrix} \begin{bmatrix} v_\pi(\text{bull}) \\ v_\pi(\text{bear}) \\ v_\pi(\text{flat}) \end{bmatrix}$$

In general, let $\mathcal{S} = \{\varsigma_1, \dots, \varsigma_n\}$. (To avoid confusion, we do not use $\{s_1, \dots, s_n\}$ to denote $\mathcal{S}$ because the indices of s represent time.) Then, we can write n Bellman equations into the vector form

$$\underbrace{\begin{bmatrix} v_\pi(\varsigma_1) \\ v_\pi(\varsigma_2) \\ \vdots \\ v_\pi(\varsigma_n) \end{bmatrix}}_{\mathbf{v}_\pi} = \underbrace{\begin{bmatrix} r(\varsigma_1, \pi(\varsigma_1)) \\ r(\varsigma_2, \pi(\varsigma_2)) \\ \vdots \\ r(\varsigma_n, \pi(\varsigma_n)) \end{bmatrix}}_{\mathbf{r}_\pi} + \gamma \underbrace{\begin{bmatrix} p(\varsigma_1 \mid \varsigma_1, \pi(\varsigma_1)) & \dots & p(\varsigma_n \mid \varsigma_1, \pi(\varsigma_1)) \\ p(\varsigma_1 \mid \varsigma_2, \pi(\varsigma_1)) & \dots & p(\varsigma_n \mid \varsigma_2, \pi(\varsigma_1)) \\ \vdots & \dots & \vdots \\ p(\varsigma_1 \mid \varsigma_n, \pi(\varsigma_1)) & \dots & p(\varsigma_n \mid \varsigma_n, \pi(\varsigma_1)) \end{bmatrix}}_{\mathbf{P}_\pi} \cdot \underbrace{\begin{bmatrix} v_\pi(\varsigma_1) \\ v_\pi(\varsigma_2) \\ \vdots \\ v_\pi(\varsigma_n) \end{bmatrix}}_{\mathbf{v}_\pi}$$

> Bellman equation (vector form)
>
> $$\mathbf{v}_\pi = \mathbf{r}_\pi + \gamma \mathbf{P}_\pi \mathbf{v}_\pi \tag{11}$$

Remarks:

- $\mathbf{v}_\pi$ comprises state values for all $s \in \mathcal{S}$ under policy π
- $\mathbf{r}_\pi$ comprises immediate rewards arriving at $s \in \mathcal{S}$ under policy π
- $\mathbf{P}_\pi$ comprises all state transition probabilities under policy π
- All state values, immediate rewards and state transition probabilities depend on policy π.

Now, given the Bellman equations (either in element-wise or vector form), we ask two questions

1. Given the policy π, how to compute the state values? This is called ***policy evaluation***
   $\rightarrow$ analytical solution or fixed point iteration
2. Is there a policy which maximizes the state values? If so, how to find it?
   $\rightarrow$ dynamic programming

Bellman Equation for Q-function

Similarly, Q-function also has recursive structure

$$q_\pi(s, a) = r(s, a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)}[q_\pi(s', \pi(s'))]$$

Policy Evaluation

Given a policy π, computing its value function $v_\pi(\cdot)$ is called ***policy evaluation***. Effectively, we would like to evaluate how good π is for each state s. This is equivelent to solving Bellman equations.

Solution in Finite State Space

If $\mathcal{S}$ is fininte, policy evaluation boils down to solving a system of linear equations $\mathbf{v}_\pi = \mathbf{r}_\pi + \gamma \mathbf{P}_\pi \mathbf{v}_\pi$. It is easy to verify that the analytical solution to the Bellman equations is

$$\mathbf{v}_\pi = (\mathbf{I} - \gamma \mathbf{P}_\pi)^{-1} \mathbf{r}_\pi \tag{12}$$

where $\mathbf{I}$ is the $|\mathcal{S}| \times |\mathcal{S}|$ identity matrix.

Drawback of analytical solution: involves matrix inversion. High computational complexity when $|\mathcal{S}|$ is large.

Numerical solution (fixed point iteration): The state values can be obtained from the following algorithm

Initialize $\mathbf{v}^{(0)}$ arbitrarily.
For $i = 0, 1, \ldots$, run until convergence:

$$\mathbf{v}^{(i+1)} = \mathbf{r}_\pi + \gamma \mathbf{P}_\pi \mathbf{v}^{(i)} \tag{13}$$

Remarks:

- During iteration, $\mathbf{v}^{(i)}$ is not a true state value vector since $\mathbf{v}^{(i)}$ itself does not satisfy Bellman equation. There is no policy associated with $\mathbf{v}^{(i)}$.
- The sequence $\mathbf{v}^{(0)}, \mathbf{v}^{(1)}, \ldots$ obtained from above iteration converges to $\mathbf{v}_\pi$, i.e.

$$\lim_{i \to \infty} \mathbf{v}^{(i)} = \mathbf{v}_\pi = (\mathbf{I} - \gamma \mathbf{P}_\pi)^{-1} \mathbf{r}_\pi \tag{14}$$

Proof of convergence: Let $n = |\mathcal{S}|$. From Bellman equation, we know that $\mathbf{v}_\pi$ is a fixed point of the affine function

$$f : \mathbb{R}^n \to \mathbb{R}^n, \mathbf{v} \mapsto f(\mathbf{v}) = \mathbf{r}_\pi + \gamma \mathbf{P}_\pi \mathbf{v}$$

We show that $f(\cdot)$ is a contractive mapping under infinity norm.

$$\begin{aligned} \|f(\mathbf{u}) - f(\mathbf{v})\|_\infty &= \|(\mathbf{r}_\pi + \gamma \mathbf{P}_\pi \mathbf{u}) - (\mathbf{r}_\pi + \gamma \mathbf{P}_\pi \mathbf{v})\|_\infty \\ &= \gamma \|\mathbf{P}_\pi (\mathbf{u} - \mathbf{v})\|_\infty \\ &\leq \gamma \|(\mathbf{u} - \mathbf{v})\|_\infty, \end{aligned}$$

The last step follows from the fact that $\|\mathbf{P}_\pi \mathbf{x}\|_\infty \leq \|\mathbf{x}\|_\infty, \forall x \in \mathbb{R}^n$, i.e. multiplication with row stochastic matrix does not increase infnity norm. (c.f. Appendix).

By contraction mapping theorem (c.f. separate notes), we conclude that

1. $f(\cdot)$ has a unique fixed point. Since $\mathbf{v}_\pi = f(\mathbf{v}_\pi)$ by Bellman equation, $\mathbf{v}_\pi$ is the unique fixed point.
2. $\forall \mathbf{v}^{(0)} \in \mathbb{R}^n$, the sequence defined by $\mathbf{v}^{(i+1)} = f(\mathbf{v}^{(i)})$ converges to $\mathbf{v}_\pi$ in infnity norm

Since all p-norms in $\mathbb{R}^n$ are equivalent, convergence in infnity norm implies convergence in any p-norm. □

Bellman Operator

If $\mathcal{S}$ is a infinite set, there is generally no closed-form solution for $v_\pi(s)$ expect for a few special cases (not covered here). Here, we only show a theoretical study based on Bellman operator.

Let $\mathcal{V}$ be the set of all **bounded** value functions. Then, $\mathcal{V}$ with the sup norm $\|\cdot\|_\infty$ is a metric space

$$\mathcal{V} = \left\{ v : \mathcal{S} \to \mathbb{R} \;\middle|\; \|v\|_\infty = \max_{s \in \mathcal{S}} |v(s)| < \infty \right\}$$

For a certain policy π, we define the corresponding Bellman operator $\mathcal{B}_\pi$ which maps a state value function $v(\cdot)$ to another value function $\mathcal{B}_\pi v(\cdot)$.

$$\mathcal{B}_\pi : \mathcal{V} \to \mathcal{V}, v(\cdot) \mapsto \mathcal{B}_\pi v(\cdot)$$

The resulting value function is

$$\mathcal{B}_\pi v(s) = r(s, \pi(s)) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,\pi(s))}[v(s')]$$

Properties of Bellman operator:

1. $\mathcal{B}_\pi$ is monotonic, i.e.

$$u(s) \leq v(s), \forall s \in \mathcal{S} \implies \mathcal{B}_\pi u(s) \leq \mathcal{B}_\pi v(s), \forall s \in \mathcal{S}$$

2. $\mathcal{B}_\pi$ is a contractive mapping. i.e.

$$\forall u, v : \mathcal{S} \to \mathbb{R},\ \|\mathcal{B}_\pi u - \mathcal{B}_\pi v\|_\infty \leq \|u - v\|_\infty$$

3. $v_\pi(\cdot)$ is the unique fixed point of $\mathcal{B}_\pi$, i.e.

$$\mathcal{B}_\pi v_\pi(s) = v_\pi(s), \forall s \in \mathcal{S}$$

Proof 1: By the monotonicity of expectation
$\mathbb{E}_{s' \sim p(\cdot|s,\pi(s))}[u(s')] \leq \mathbb{E}_{s' \sim p(\cdot|s,\pi(s))}[v(s')]$, we conclude. $\square$

Proof 2: Recall that the infinity norm of a function f is defined as

$$\|f\|_\infty \triangleq \max_x |f(x)|$$

Consider $|\mathcal{B}_\pi u(s) - \mathcal{B}_\pi v(s)|$ for all $s \in \mathcal{S}$.

$$\begin{aligned}
|\mathcal{B}_\pi u(s) - \mathcal{B}_\pi v(s)| &= \left| r(s,\pi(s)) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,\pi(s))}[u(s')] - r(s,\pi(s)) - \gamma \mathbb{E}_{s' \sim p(\cdot|s,\pi(s))}[v(s')] \right| \\
&= |\gamma| \cdot \left| \mathbb{E}_{s' \sim p(\cdot|s,\pi(s))}[u(s') - v(s')] \right| \\
&\leq \gamma \cdot \mathbb{E}_{s' \sim p(\cdot|s,\pi(s))} \left[|u(s') - v(s')| \right] \\
&\leq \gamma \cdot \mathbb{E}_{s' \sim p(\cdot|s,\pi(s))} \left[\|u - v\|_\infty \right] \\
&\leq \gamma \cdot \|u - v\|_\infty
\end{aligned}$$

Hence, we conclude

$$\|\mathcal{B}_\pi u - \mathcal{B}_\pi v\|_\infty = \max_{s \in \mathcal{S}} |\mathcal{B}_\pi u(s) - \mathcal{B}_\pi v(s)| \leq \gamma \cdot \|u - v\|_\infty$$

Proof 3: By Bellman equation, we know that v_π is a fixed point of $\mathcal{B}_\pi$. By contraction mapping theorem, we conclude that v_π is the unique fixed point.

Followed by contraction mapping theorem, the state value function can be obtained through fixed point iteration

Starting from any $v^{(0)}(\cdot) \in \mathcal{V}$
For $i = 0, 1, \ldots$, run until $v^{(i)}(\cdot)$ converges
  For each $s \in \mathcal{S}$, do

$$v^{(i+1)}(s) = \mathcal{B}_\pi v(s) = r(s,\pi(s)) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,\pi(s))}[v(s')]$$

Mathematically, the resulting function sequence $\{v^{(i)}\}_{i \geq 0}$ converges to v_π in the sup norm (and thus converges pointwise)

$$\lim_{i \to \infty} \left\| v^{(i)} - v_\pi \right\|_\infty = 0 \implies \lim_{i \to \infty} v^{(i)}(s) = v_\pi(s), \forall s \in \mathcal{S}$$

However, above algorithm can not be directly implementend since we can not evaluate $v^{(i)}(s)$ for infinitely many s. In practice, we use approximation techniques to estimate $v_\pi(\cdot)$. (Not detailed here.)

Bellman Optimality Equations

A policy π outperforms another policy $\tilde{\pi}$ iff the state values $v_\pi(s)$ outperforms $v_{\tilde{\pi}}(s)$ for **ALL** states $s \in \mathcal{S}$. i.e.

$$\forall s \in \mathcal{S},\, v_\pi(s) \geq v_{\tilde{\pi}}(s)$$

A policy π^* is optimal if it outperforms any other policies, i.e.

$$\forall \pi, \forall s \in \mathcal{S},\, v_{\pi^*}(s) \geq v_\pi(s)$$

The optimal state value $v^*(s)$ is the state value under π^*

$$v^*(s) \triangleq v_{\pi^*}(s) = \max_\pi v_\pi(s) \tag{15}$$

We haven't proved the existence of π^*. For now, let's assume its existence and discover what conditions have to be met for π^* and $v^*(s)$. This will lead us to Bellman optimality equations, from which we will derive an algorithm to find π^* (and thus prove its existence).

Bellman Optimality Criterion

Recall the Bellman equation for $v_\pi(s)$ holds for any policy. In particular, Bellman equations also hold for π^*:

$$v^*(s) = r(s, \pi^*(s)) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,\pi^*(s))}[v^*(s')], \quad \forall s \in \mathcal{S}$$

Yet, we are unable to solve the optimal state value since the optimal current action $\pi^*(s)$ is unknown. However, we can reformulate $v^*(s)$ without explicit reference to $\pi^*(s)$.

Trick: Note that $\pi^*(s)$ is the best action to take on current state s. $\implies$ Taking $\pi^*(s)$ now followed by executing π^* yields a state value no less than taking any other $a \in \mathcal{A}$ followed by exectuting π^*, as illustrated below.

- Optimal: Executing the optimal policy from now onward.

$$s \xrightarrow[r(s,\,\pi^*(s))]{\pi^*(s)} S_1 \xrightarrow[R_1]{\pi^*(S_1)} S_2 \xrightarrow[R_2]{\pi^*(S_2)} S_3 \cdots$$

- (Sub)optimal: Taking any other $a \in \mathcal{A} \setminus \{\pi^*(s)\}$ now, followed by executing π^* onward.

$$s \xrightarrow[r(s,\,a)]{a} S_1 \xrightarrow[R_1]{\pi^*(S_1)} S_2 \xrightarrow[R_2]{\pi^*(S_2)} S_3 \cdots$$

Formally, this means

$$\forall s \in \mathcal{S}, \forall a \in \mathcal{A}:\ v^*(s) \geq r(s,a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)}[v^*(s')]$$

where the equality holds iff $a = \pi^*(s)$. Namely, the optimal action at state s should maximize the sum of immediate reward and the (discounted) expected future reward

$$\pi^*(s) = \arg\max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)}[v^*(s')] \right\}, \quad \forall s \in \mathcal{S} \tag{16}$$

Remarks:

- This equation holds for all $s \in \mathcal{S}$.

- Suppose we solved all optimal state values $\{v^*(s') \mid s' \in \mathcal{S}\}$, plugging them into this equation yields the optimal policy.

The optimal state value $v^*(s)$ thus satisfies the ***Bellman optimality equations (BOE)***:

$$v^*(s) = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)}[v^*(s')] \right\}, \quad \forall s \in \mathcal{S} \tag{17}$$

Remarks:

- Previously, computing $v^*(s)$ requires knowledge of $r(s, \pi^*(s))$ and $p(\cdot \mid s, \pi^*(s))$, but since $\pi^*(s)$ is unknown, solving $v^*(s)$ is challenging.
- Now with the BOE, we bypass the need to know $\pi^*(s)$ explicitly. Instead, we evaluate $r(s,a)$ and $p(\cdot \mid s, a)$ (which are provided by MDP) for all $a \in \mathcal{A}$. Then, $v^*(s)$ can be solved by solving the optimization problem (detailed later).
- Just like Bellman equation, BOE holds for all $s \in \mathcal{S}$.

For finite state space, the BOE becomes

$$v^*(s) = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s'} p(s' \mid s, a) \cdot v^*(s') \right\}, \quad \forall s \in \mathcal{S} \tag{18}$$

Again, let $\mathcal{S} = \{\varsigma_1, \dots, \varsigma_n\}$. Then, we can write n BOEs into vector form

$$\begin{bmatrix} v^*(\varsigma_1) \\ v^*(\varsigma_2) \\ \vdots \\ v^*(\varsigma_n) \end{bmatrix} = \begin{bmatrix} \max_{a \in \mathcal{A}} \{r(\varsigma_1, a) + \gamma [p(\varsigma_1 \mid \varsigma_1, a) v^*(\varsigma_1) + \cdots + p(\varsigma_n \mid \varsigma_1, a) + v^*(\varsigma_n)]\} \\ \max_{a \in \mathcal{A}} \{r(\varsigma_2, a) + \gamma [p(\varsigma_1 \mid \varsigma_2, a) v^*(\varsigma_1) + \cdots + p(\varsigma_n \mid \varsigma_2, a) + v^*(\varsigma_n)]\} \\ \vdots \\ \max_{a \in \mathcal{A}} \{r(\varsigma_n, a) + \gamma [p(\varsigma_1 \mid \varsigma_n, a) v^*(\varsigma_1) + \cdots + p(\varsigma_n \mid \varsigma_n, a) + v^*(\varsigma_n)]\} \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} v^*(\varsigma_1) \\ v^*(\varsigma_2) \\ \vdots \\ v^*(\varsigma_n) \end{bmatrix}}_{\mathbf{v}^*} = \max_{a \in \mathcal{A}} \left\{ \underbrace{\begin{bmatrix} r(\varsigma_1, a) \\ r(\varsigma_2, a) \\ \vdots \\ r(\varsigma_n, a) \end{bmatrix}}_{\mathbf{r}_a} + \gamma \underbrace{\begin{bmatrix} p(\varsigma_1 \mid \varsigma_1, a) & \dots & p(\varsigma_n \mid \varsigma_1, a) \\ p(\varsigma_1 \mid \varsigma_2, a) & \dots & p(\varsigma_n \mid \varsigma_2, a) \\ \vdots & \dots & \vdots \\ p(\varsigma_1 \mid \varsigma_n, a) & \dots & p(\varsigma_n \mid \varsigma_n, a) \end{bmatrix}}_{\mathbf{P}_a} \cdot \underbrace{\begin{bmatrix} v^*(\varsigma_1) \\ v^*(\varsigma_2) \\ \vdots \\ v^*(\varsigma_n) \end{bmatrix}}_{\mathbf{v}^*} \right\}$$

Bellman Optimality Equation (vector form)

$$\mathbf{v}^* = \max_{a \in \mathcal{A}} \{\mathbf{r}_a + \gamma \mathbf{P}_a \mathbf{v}^*\} \tag{19}$$

where $\max$ acting on a vector is taken element-wise.

$$\max_x \mathbf{w}(x) = [\max_x w_1(x) \quad \cdots \quad \max_x w_n(x)]^\top$$

Solving Optimal State Values

Similar to the algorithm to solve the state values $\mathbf{v}_\pi$ for any given policy, we introduce the following algorithm to solve the optimal state values $\mathbf{v}^*$.

Init $\mathbf{v}^{(0)}$ arbitrarily
For $i = 0, 1, \ldots$, run until convergence

$$\mathbf{v}^{(i+1)} = \max_{a \in \mathcal{A}} \left\{ \mathbf{r}_a + \gamma \mathbf{P}_a \mathbf{v}^{(i)} \right\} \tag{20}$$

Remarks:

- The sequence $\mathbf{v}^{(0)}, \mathbf{v}^{(1)}, \ldots$ obtained from above iteration converges to $\mathbf{v}^*$, i.e.

$$\lim_{i \to \infty} \mathbf{v}^{(i)} = \mathbf{v}^* \tag{21}$$

- Having computed the optimal state values $\mathbf{v}^*$, the optimal policy is obtained from

$$\pi^*(s) = \arg\max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s'} p(s' \mid s, a) \cdot v^*(s') \right\}, \quad \forall s \in \mathcal{S} \tag{22}$$

- Reminder: The iteration does **not** ensure that the intermediate result $\mathbf{v}^{(i)}$ satisfy Bellman equation **for any policy**. However, the limit of $\mathbf{v}^{(i)}$ satisfies BOEs, i.e. the Bellman equation for the optimal policy.

Proof of convergence: In the following, all $\max(\cdot)$, $|\cdot|$ and inequalities are taken element-wise when acting on vectors. Let $\mathcal{S} = \{\varsigma_1, \ldots, \varsigma_n\}$ and

$$f : \mathbb{R}^n \to \mathbb{R}^n, \mathbf{v} \mapsto f(\mathbf{v}) = \max_{a \in \mathcal{A}} \{ \mathbf{r}_a + \gamma \mathbf{P}_a \mathbf{v} \}$$

By BOE, $\mathbf{v}^*$ is a fixed point of $f(\cdot)$. To prove the convergence, it is sufficient to show that $f(\cdot)$ is contractive. For any $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$, we have two optimization problems w.r.t. a. (The maximization is taken element-wise)

$$f(\mathbf{u}) = \max_{a \in \mathcal{A}} \{ \mathbf{r}_a + \gamma \mathbf{P}_a \mathbf{u} \}$$
$$f(\mathbf{v}) = \max_{a \in \mathcal{A}} \{ \mathbf{r}_a + \gamma \mathbf{P}_a \mathbf{v} \}$$

For $f(\mathbf{u})$, let $\hat{a}_k$ be the optimizer at k-row of $\mathbf{r}_a + \gamma \mathbf{P}_a \mathbf{u}$. (Note: $\hat{a}_k$ depends on $\mathbf{u}$)

$$\hat{a}_k = \arg\max_{a \in \mathcal{A}} \left\{ r(\varsigma_k, a) + \gamma \sum_j p(\varsigma_j \mid \varsigma_k, a) \cdot u_j \right\}$$

Then, we can express $f(\mathbf{u})$ in with $\mathbf{r}_{\hat{a}}$ and $\mathbf{P}_{\hat{a}}$, defined as follows.

$$\mathbf{r}_{\hat{a}} \triangleq \begin{bmatrix} r(\varsigma_1, \hat{a}_1) \\ \vdots \\ r(\varsigma_n, \hat{a}_n) \end{bmatrix}, \ \mathbf{P}_{\hat{a}} \triangleq \begin{bmatrix} p(\varsigma_1 \mid \varsigma_1, \hat{a}_1) & \cdots & p(\varsigma_n \mid \varsigma_1, \hat{a}_1) \\ \vdots & \ddots & \vdots \\ p(\varsigma_1 \mid \varsigma_n, \hat{a}_n) & \cdots & p(\varsigma_n \mid \varsigma_n, \hat{a}_n) \end{bmatrix} \implies f(\mathbf{u}) = \mathbf{r}_{\hat{a}} + \gamma \mathbf{P}_{\hat{a}} \mathbf{u}$$

Likewise, for $f(\mathbf{v})$, let $\hat{b}_k$ be the optimizer at k-row of $\mathbf{r}_a + \gamma \mathbf{P}_a \mathbf{v}$. (Note: $\hat{b}_k$ depends on $\mathbf{v}$. Hence, $\hat{a}_k \neq \hat{b}_k$ in general)

$$\hat{b}_k = \arg\max_{a \in \mathcal{A}} \left\{ r(\varsigma_k, a) + \gamma \sum_j p(\varsigma_j \mid \varsigma_k, a) \cdot v_j \right\}$$

Define $\mathbf{r}_{\hat{b}}$ and $\mathbf{P}_{\hat{b}}$ in the same way. $\implies f(\mathbf{v}) = \mathbf{r}_{\hat{b}} + \gamma \mathbf{P}_{\hat{b}} \mathbf{v}$.

By the optimality of $\{\hat{a}_1, \ldots, \hat{a}_n\}$ and $\{\hat{b}_1, \ldots, \hat{b}_n\}$,

$$f(\mathbf{u}) = \mathbf{r}_{\hat{a}} + \gamma \mathbf{P}_{\hat{a}}\mathbf{u} \geq \mathbf{r}_{\hat{b}} + \gamma \mathbf{P}_{\hat{b}}\mathbf{u}$$
$$f(\mathbf{v}) = \mathbf{r}_{\hat{b}} + \gamma \mathbf{P}_{\hat{b}}\mathbf{v} \geq \mathbf{r}_{\hat{a}} + \gamma \mathbf{P}_{\hat{a}}\mathbf{v}$$

Hence, $f(\mathbf{u}) - f(\mathbf{v})$ is element-wise bounded as follows

$$\begin{aligned} f(\mathbf{u}) - f(\mathbf{v}) &= (\mathbf{r}_{\hat{a}} + \gamma \mathbf{P}_{\hat{a}}\mathbf{u}) - (\mathbf{r}_{\hat{b}} + \gamma \mathbf{P}_{\hat{b}}\mathbf{v}) \\ &\geq (\mathbf{r}_{\hat{b}} + \gamma \mathbf{P}_{\hat{b}}\mathbf{u}) - (\mathbf{r}_{\hat{b}} + \gamma \mathbf{P}_{\hat{b}}\mathbf{v}) = \gamma \mathbf{P}_{\hat{b}}(\mathbf{u} - \mathbf{v}) \\ f(\mathbf{u}) - f(\mathbf{v}) &= (\mathbf{r}_{\hat{a}} + \gamma \mathbf{P}_{\hat{a}}\mathbf{u}) - (\mathbf{r}_{\hat{b}} + \gamma \mathbf{P}_{\hat{b}}\mathbf{v}) \\ &\leq (\mathbf{r}_{\hat{a}} + \gamma \mathbf{P}_{\hat{a}}\mathbf{u}) - (\mathbf{r}_{\hat{a}} + \gamma \mathbf{P}_{\hat{a}}\mathbf{v}) = \gamma \mathbf{P}_{\hat{a}}(\mathbf{u} - \mathbf{v}) \\ \implies \gamma \mathbf{P}_{\hat{b}}(\mathbf{u} - \mathbf{v}) &\leq f(\mathbf{u}) - f(\mathbf{v}) \leq \gamma \mathbf{P}_{\hat{a}}(\mathbf{u} - \mathbf{v}) \end{aligned}$$

Taking the absolute values of $f(\mathbf{u}) - f(\mathbf{v})$ element-wise yields

$$|f(\mathbf{u}) - f(\mathbf{v})| \leq \max\left\{\gamma\left|\mathbf{P}_{\hat{b}}(\mathbf{u} - \mathbf{v})\right|, \gamma\left|\mathbf{P}_{\hat{a}}(\mathbf{u} - \mathbf{v})\right|\right\} \leq \gamma \|\mathbf{u} - \mathbf{v}\|_\infty \cdot \mathbf{1}$$

- ⓘ The last inequality follows from the property of row-stochastic matrix (c.f. Appendix): If $\mathbf{P} \in \mathbb{R}^{n \times n}$ is a row-stochastic matrix, then

$$\forall \mathbf{x} \in \mathbb{R}^n : \left|(\mathbf{P}\mathbf{x})_i\right| \leq \|\mathbf{x}\|_\infty \iff \left|(\mathbf{P}\mathbf{x})\right| \leq \|\mathbf{x}\|_\infty \cdot \mathbf{1}$$

Namely, all elements of $|f(\mathbf{u}) - f(\mathbf{v})|$ is boudned by $\gamma\|\mathbf{u} - \mathbf{v}\|_\infty$. Hence,

$$\|f(\mathbf{u}) - f(\mathbf{v})\|_\infty \leq \gamma\|\mathbf{u} - \mathbf{v}\|_\infty \iff f(\cdot) \text{ is contractive} \quad \square$$

Proof of optimal policy: When deriving BOEs, we showed that

$$\pi^*(s) = \arg\max_{a \in \mathcal{A}} \left\{r(s,a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)}[v^*(s')]\right\}, \quad \forall s \in \mathcal{S}$$

Expanding the expectation into a sum, we conclude. $\square$

Optimal Q-function

Similary, the optimal Q-function is defined as

$$q^*(s,a) = q_{\pi^*}(s,a)$$

Relation between optimal Q-function and optimal state value:

- Compute $v^*(s)$ from $q^*(s,a)$: simply let $a = \pi^*(s)$, i.e.

$$v^*(s) = q^*(s,a)\Big|_{a=\pi^*(s)} \tag{23}$$

- Compute $q^*(s,a)$ from $v^*(s)$: use recurisve structure

$$\begin{aligned} q^*(s,a) &= r(s,a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)}[v^*(s')] && (24) \\ &= r(s,a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s,a) v^*(s') \qquad \text{if } \mathcal{S} \text{ is finite} && (25) \end{aligned}$$

The Bellman optimality criterion can also be formulated in terms of Q-function:

$$v^*(s) = \max_{a \in \mathcal{A}} q^*(s,a) \tag{26}$$
$$\pi^*(s) = \arg\max_{a \in \mathcal{A}} q^*(s,a) \tag{27}$$

Dynamic Programming

Throughout this section, we assume that both state space and action space are discrete. Given the parameters of an MDP:

- state transition probabilities $p(s' \mid s, a)$ for all $s, s' \in \mathcal{S}, a \in \mathcal{A}$
- state-action-rewards $r(s, a)$ for all $s \in \mathcal{S}, a \in \mathcal{A}$

From previous sections, we knew that an optimal policy π^* exists. Now, we focus on designing algorithms to compute π^*.

Value Iteration

The algorithm introduced earlier to solve BOEs is called value iteration. For the sake of implementation, the algorithm can be unfolded element-wise as follows

> Init $v^{(0)}(s)$ for all $s \in \mathcal{S}$ by random guessing
> For $i = 0, 1, \ldots$, do
> For each $s \in \mathcal{S}$, do
> For each $a \in \mathcal{A}$, compute Q-function
> $$q^{(i)}(s, a) = r(s, a) + \gamma \sum_{s'} p(s' \mid s, a) \cdot v^{(i)}(s')$$
> **Policy update**: $\pi^{(i+1)}(s) = \arg\max_{a \in \mathcal{A}} q^{(i)}(s, a)$
> **Value update**: $v^{(i+1)}(s) = \max_{a \in \mathcal{A}} q^{(i)}(s, a)$
> until $\left\| \mathbf{v}^{(i+1)} - \mathbf{v}^{(i)} \right\| \leq$ some threshold ϵ. (i.e. $\mathbf{v}^{(i)}$ converges)
> Return $v^{(i_{\text{stop}})}(s)$ and $\pi^{(i_{\text{stop}})}(s)$ for all $s \in \mathcal{S}$

Remarks:

- In stopping condition, $\mathbf{v}^{(i)}$ is the vector containing all $v^{(i)}(s), s \in \mathcal{S}$. The norm can be infinity norm or any other norms.
- Although $v^{(i)}(s)$ converges to $v^*(s)$ for all $s \in \mathcal{S}$, the intermediate values $v^{(i)}(s)$ do **not** generally satisfy Bellman equation for **any** policy. We interpret $v^{(i)}(s)$ as the estimate of $v^*(s)$ at i-th iteration rather than the state values under $\pi^{(i)}$ or $\pi^{(i+1)}$, i.e.
$$v^{(i)}(s) \neq r(s, \pi^{(i)}(s)) + \gamma \sum_{s'} p(s' \mid s, \pi^{(i)}(s)) v^{(i)}(s)$$
$$v^{(i)}(s) \neq r(s, \pi^{(i+1)}(s)) + \gamma \sum_{s'} p(s' \mid s, \pi^{(i+1)}(s)) v^{(i)}(s)$$
- Likewise, $q^{(i)}(s, a)$ represents the estimate of $q^*(s, a)$ at i-th iteration.

In policy update step, the new policy $\pi^{(i+1)}(s)$ always picks the action maximizing the current estimtae of Q-function $q^{(i)}(s, a)$. Hence, it is called ***greedy*** policy update.

Policy Iteration

Policy iteration is another algorithm to compute optimal policy. It starts with abitrary policy and iteratively improves it. Formally:

> Init $\pi^{(0)}$ by random guessing
> For $i = 0, 1, 2, \ldots$, run until convergence
> **Policy evaluation**: Solve $\mathbf{v}_{\pi^{(i)}} = \mathbf{r}_{\pi^{(i)}} + \gamma \mathbf{P}_{\pi^{(i)}} \mathbf{v}_{\pi^{(i)}}$ for state values $\mathbf{v}_{\pi^{(i)}}$
> **Policy improvement**: Solve $\pi^{(i+1)} = \arg\max_{\pi} \left\{ \mathbf{r}_\pi + \gamma \mathbf{P}_\pi \mathbf{v}_{\pi^{(i)}} \right\}$ for new policy $\pi^{(i+1)}$

Remark:

- The convergence will be proved later.
- In policy evaluation, the state values can be computed either analytically using $\mathbf{v}_\pi = (\mathbf{I} - \gamma \mathbf{P}_\pi)^{-1}\mathbf{r}_\pi$ or numerically using fixed point iteration.
- In policy improvement, the new policy maximizes the immediate reward plus the discounted future reward by following $\pi^{(i)}$. Element-wise, this breaks down to maximizing the Q-functions:

$$\pi^{(i+1)}(s) = \arg\max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s,a)\, v_{\pi^{(i)}}(s') \right\}, \quad \forall s \in \mathcal{S} \tag{28}$$

$$= \arg\max_{a \in \mathcal{A}} q_{\pi^{(i)}}(s,a) \tag{29}$$

- Each intermediate $\mathbf{v}_{\pi^{(i)}}$ satisfies the Bellman equation for policy $\pi^{(i)}$. vs. In value iteration, $\mathbf{v}^{(i)}$ does not generally satisfy Bellman equation for any policy!

Element-wise formulation of policy iteration:

> Init $\pi^{(0)}(s)$ for all $s \in \mathcal{S}$ by random guessing
> For $i = 0, 1, 2, \ldots$, do
> **Policy evaluation**: compute $v_{\pi^{(i)}}(s)$ for all $s \in \mathcal{S}$ by solving linear equations
> For each $s \in \mathcal{S}$, do
> For each $a \in \mathcal{A}$, compute
> Q-function: $q_{\pi^{(i)}}(s,a) = r(s,a) + \gamma \sum_{s' \in \mathcal{S}} p(s' \mid s,a)\, v_{\pi^{(i)}}(s')$
> **Policy improvement**: $\pi^{(i+1)}(s) = \arg\max_{a \in \mathcal{A}} q_{\pi^{(i)}}(s,a)$
> until $\| \mathbf{v}_{\pi^{(i+1)}} - \mathbf{v}_{\pi^{(i)}} \| < \epsilon$
> Return $v_{\pi^{(i)}}(s)$ and $\pi^{(i)}(s)$ for all $s \in \mathcal{S}$

Policy iteratoin works because of following facts

1. Policy improvement theorem: the policy is indeed improved iteratively.

$$\pi^{(i+1)}(s) = \arg\max_{a \in \mathcal{A}} q_{\pi^{(i)}}(s,a) \implies v_{\pi^{(i+1)}}(s) \geq v_{\pi^{(i)}}(s),\ \forall s \in \mathcal{S}$$

2. The sequence of state values generated by policy iteration indeed converges to the optimal state values.

$$\lim_{i \to \infty} v_{\pi^{(i)}}(s) = v^*(s),\ \forall s \in \mathcal{S}$$

Proof 1: For each $s \in \mathcal{S}$, $\pi^{(i+1)}(s)$ is the maximizer of the Q-function $q_{\pi^{(i)}}(s,a)$. In particular,

$$q_{\pi^{(i)}}(s, \pi^{(i+1)}(s)) \geq q_{\pi^{(i)}}(s, \pi^{(i)}(s)) = v_{\pi^{(i)}}(s) \tag{$\star$}$$

Hence, $\forall s \in \mathcal{S}$:

$$\begin{aligned} v_{\pi^{(i+1)}}(s) - v_{\pi^{(i)}}(s) &\overset{(\star)}{\geq} v_{\pi^{(i+1)}}(s) - q_{\pi^{(i)}}(s, \pi^{(i+1)}(s)) \\ &= r(s, \pi^{(i+1)}(s)) + \gamma \mathbb{E}[v_{\pi^{(i+1)}}(S')] - \Big(r(s, \pi^{(i+1)}(s)) + \gamma \mathbb{E}[v_{\pi^{(i)}}(S')] \Big) \\ &= \gamma \mathbb{E}\left[v_{\pi^{(i+1)}}(S') - v_{\pi^{(i)}}(S') \right] \\ &\geq \gamma \min_{s' \in \mathcal{S}} \left\{ v_{\pi^{(i+1)}}(s') - v_{\pi^{(i)}}(s') \right\} \end{aligned} \tag{$\star\star$}$$

The last step follows by the property of expectation: $\mathbb{E}[X] \geq x_{\min}$ with $X = v_{\pi^{(i+1)}}(S') - v_{\pi^{(i)}}(S')$.

Taking $\min_{s \in \mathcal{S}}$ on the LHS, we get

$$\min_{s \in \mathcal{S}} \left\{ v_{\pi^{(i+1)}}(s) - v_{\pi^{(i)}}(s) \right\} \geq \gamma \min_{s' \in \mathcal{S}} \left\{ v_{\pi^{(i+1)}}(s') - v_{\pi^{(i)}}(s') \right\}$$

Note that the optimization problem on both sides are the same. Hence,

$$\begin{aligned} (1-\gamma) \min_{s \in \mathcal{S}} \left\{ v_{\pi^{(i+1)}}(s) - v_{\pi^{(i)}}(s) \right\} &\geq 0 \\ \min_{s \in \mathcal{S}} \left\{ v_{\pi^{(i+1)}}(s) - v_{\pi^{(i)}}(s) \right\} &\geq 0 \qquad \text{since } 0 < \gamma < 1 \\ \forall s \in \mathcal{S},\ v_{\pi^{(i+1)}}(s) - v_{\pi^{(i)}}(s) &\geq 0 \qquad \text{by } (\star\star) \end{aligned}$$

We concluded that the policy improvement does not decrease state value. □

Generalized Policy Iteration

Generalization

Stochastic policy and stochastic rewards.

A policy can also be stochastic. i.e. instead of mapping the state to a fixed action, there are multiple possible actions with different probability. A stochastic policy is described by the conditional distribution

$$\pi(a \mid s), a \in \mathcal{A}, s \in \mathcal{S}$$

Remarks:

- The determinstic policy can be seen as a special of stochastic policy by assigning $\pi(\hat{a} \mid s)$ to 1 for some $\hat{a}$.

$$\pi(a \mid s) = \begin{cases} 1 & a = \hat{a} \\ 0 & \text{else} \end{cases}$$

- For stochastic policy, it holds that

$$\sum_{a} \pi(a \mid s) = 1$$

- Both deterministic and stochastic policy are time-invariant. i.e. The distribution of a given s is always the same, regardless when we arrived at s.

From now on, we stick to deterministic policy and we will answer the following quesitons

1. How to quantify the goodness of a policy?
   $\rightarrow$ state values, Bellman equations
2. Which criterion should the optimal policy satisfy?
   $\rightarrow$ Bellman optimality equations.
3. How to find the optimal policy?
   $\rightarrow$ value iteration, policy iteration

Appendix

Cascade of Expectations

Suppose $U - X - Y$ forms a markov chain, then

$$\begin{aligned}\mathbb{E}_{XY}[g(x,y) \mid u] &\triangleq \mathbb{E}_{XY\sim p(x,y|u)}[g(x,y)] \\ &= \mathbb{E}_{X\sim p(x|u)}\left[\mathbb{E}_{Y\sim p(y|x)}[g(x,y)]\right]\end{aligned}$$

Row Stochastic Matrix

A square matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ is called a ***row stochastic matrix*** iff each row of $\mathbf{A}$ is a probability vector. i.e.

- all its entires are nonnegative: $a_{ij} \geq 0, \forall i,j \in \{1,\dots,n\}$
- and each row sums to 1. $\sum_{j=1}^{n} a_{ij} = 1, \forall i \in \{1,\dots,n\}$

Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be a state transition matrix. Then,

1. 1 is always an eigenvalue of $\mathbf{A}$.
2. Multiplication with $\mathbf{A}$ does not increase the infinity norm. i.e.

$$\|\mathbf{A}\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_\infty, \forall \mathbf{x} \in \mathbb{R}^n \tag{30}$$

3. The eigenvalues of $\mathbf{A}$ are at most 1. i.e.

$$|\lambda| \leq 1, \forall \lambda \in \mathrm{spec}(A) \tag{31}$$

Proof 1: Let $\mathbf{u} = [1,\dots,1]^\top \in \mathbb{R}^n$ be all-one vector. It is easy to verify that $\mathbf{A}\mathbf{u} = \mathbf{u}$. Hence, $\mathbf{u}$ is an eigenvector of $\mathbf{A}$ with eigen value 1. $\square$

Proof 2: Recall the infinity norm is defined by

$$\|\mathbf{x}\|_\infty \triangleq \max_{i=1,\dots,n} |x_i|$$

Let $\mathbf{y} = \mathbf{A}\mathbf{x}$. Consider the abs of i-th element of $\mathbf{y}$:

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

$\implies$ All elements of $\mathbf{y}$ are upper-bounded by $\|\mathbf{x}\|_\infty$ in abs. Hence,

$$\|\mathbf{y}\|_\infty = \max_{i=1,\dots,n} |y_i| \leq \|\mathbf{x}\|_\infty \qquad \square$$

Proof 3: Let λ be any eigenvalue of $\mathbf{A}$ and $\mathbf{v}$ be the corresponding eigenvector. Using the fact that $\|\mathbf{A}\mathbf{v}\|_\infty \leq \|\mathbf{v}\|_\infty$, we conclude

$$\|\mathbf{A}\mathbf{v}\|_\infty = \|\lambda\mathbf{v}\|_\infty = |\lambda| \cdot \|\mathbf{v}\|_\infty \leq |\mathbf{v}\|_\infty$$

Eigenvector $\mathbf{v}$ is nonzero $\implies |\lambda| \leq 1$. $\square$