Markov Decision Problems

1. Action value function

2. Policy iteration
   - Policy improvement
   - Policy improvement theorem and proof
   - Policy iteration algorithm

3. History-dependent and stochastic policies
Markov Decision Problems

1. Action value function

2. Policy iteration
   - Policy improvement
   - Policy improvement theorem and proof
   - Policy iteration algorithm

3. History-dependent and stochastic policies
Action Value Function

For $\pi \in \Pi, s \in S, a \in A$:

$$Q^\pi(s, a) \overset{\text{def}}{=} \mathbb{E}[r^0 + \gamma r^1 + \gamma^2 r^2 + \ldots \mid s^0 = s; a^0 = a; a^t = \pi(s^t) \text{ for } t \geq 1].$$
Action Value Function

For $\pi \in \Pi$, $s \in S$, $a \in A$:

$$Q^\pi(s, a) \overset{\text{def}}{=} \mathbb{E}[r^0 + \gamma r^1 + \gamma^2 r^2 + \ldots | s^0 = s; a^0 = a; a^t = \pi(s^t) \text{ for } t \geq 1].$$

$Q^\pi(s, a)$ is the expected long-term reward from starting at state $s$, taking action $a$ at $t = 0$, and following policy $\pi$ for $t \geq 1$. 
Action Value Function

For $\pi \in \Pi, s \in S, a \in A$:

$$Q^\pi(s, a) \overset{\text{def}}{=} \mathbb{E}[r^0 + \gamma r^1 + \gamma^2 r^2 + \ldots | s^0 = s; a^0 = a; a^t = \pi(s^t) \text{ for } t \geq 1].$$

$Q^\pi(s, a)$ is the expected long-term reward from starting at state $s$, taking action $a$ at $t = 0$, and following policy $\pi$ for $t \geq 1$.

$Q^\pi : S \times A \rightarrow \mathbb{R}$ is called the action value function of $\pi$. 
Action Value Function

For $\pi \in \Pi$, $s \in S$, $a \in A$:

$$Q^\pi(s, a) \overset{\text{def}}{=} \mathbb{E}[r^0 + \gamma r^1 + \gamma^2 r^2 + \ldots \mid s^0 = s; a^0 = a; a^t = \pi(s^t) \text{ for } t \geq 1].$$

$Q^\pi(s, a)$ is the expected long-term reward from starting at state $s$, taking action $a$ at $t = 0$, and following policy $\pi$ for $t \geq 1$.

$Q^\pi : S \times A \to \mathbb{R}$ is called the action value function of $\pi$.

Observe that $Q^\pi$ satisfies, for $s \in S$, $a \in A$:

$$Q^\pi(s, a) = \sum_{s' \in S} T(s, a, s') \{ R(s, a, s') + \gamma V^\pi(s') \}.$$
Action Value Function

For $\pi \in \Pi, s \in S, a \in A$:

$$Q^\pi(s, a) \overset{\text{def}}{=} \mathbb{E}[r^0 + \gamma r^1 + \gamma^2 r^2 + \ldots | s^0 = s; a^0 = a; a^t = \pi(s^t) \text{ for } t \geq 1].$$

$Q^\pi(s, a)$ is the expected long-term reward from starting at state $s$, taking action $a$ at $t = 0$, and following policy $\pi$ for $t \geq 1$.

$Q^\pi : S \times A \rightarrow \mathbb{R}$ is called the action value function of $\pi$.

Observe that $Q^\pi$ satisfies, for $s \in S, a \in A$:

$$Q^\pi(s, a) = \sum_{s' \in S} T(s, a, s') \{R(s, a, s') + \gamma V^\pi(s')\}.$$

For $\pi \in \Pi, s \in S$: $Q^\pi(s, \pi(s)) = V^\pi(s)$. 

$Q^\pi$ needs $O(n^2 k)$ operations to compute if $V^\pi$ is available.

All optimal policies have the same (optimal) action value function $Q^\star$. 

Shivaram Kalyanakrishnan (2023)

CS 747, Autumn 2023
Action Value Function

- For $\pi \in \Pi, s \in S, a \in A$:
  \[ Q^\pi(s, a) \overset{\text{def}}{=} \mathbb{E}[r_0 + \gamma r_1 + \gamma^2 r_2 + \ldots | s^0 = s; a^0 = a; a^t = \pi(s^t) \text{ for } t \geq 1]. \]

$Q^\pi(s, a)$ is the expected long-term reward from starting at state $s$, taking action $a$ at $t = 0$, and following policy $\pi$ for $t \geq 1$.

$Q^\pi : S \times A \to \mathbb{R}$ is called the action value function of $\pi$.

Observe that $Q^\pi$ satisfies, for $s \in S, a \in A$:

\[ Q^\pi(s, a) = \sum_{s' \in S} T(s, a, s') \{ R(s, a, s') + \gamma V^\pi(s') \}. \]

For $\pi \in \Pi, s \in S$: $Q^\pi(s, \pi(s)) = V^\pi(s)$.

- $Q^\pi$ needs $O(n^2 k)$ operations to compute if $V^\pi$ is available.
Action Value Function

For $\pi \in \Pi$, $s \in S$, $a \in A$:

$$Q^\pi(s, a) \overset{\text{def}}{=} \mathbb{E}[r_0 + \gamma r_1 + \gamma^2 r_2 + \ldots | s^0 = s; a^0 = a; a^t = \pi(s^t) \text{ for } t \geq 1].$$

$Q^\pi(s, a)$ is the expected long-term reward from starting at state $s$, taking action $a$ at $t = 0$, and following policy $\pi$ for $t \geq 1$.

$Q^\pi : S \times A \rightarrow \mathbb{R}$ is called the action value function of $\pi$.

Observe that $Q^\pi$ satisfies, for $s \in S$, $a \in A$:

$$Q^\pi(s, a) = \sum_{s' \in S} T(s, a, s') \{ R(s, a, s') + \gamma V^\pi(s') \}.$$

For $\pi \in \Pi$, $s \in S$: $Q^\pi(s, \pi(s)) = V^\pi(s)$.

$Q^\pi$ needs $O(n^2 k)$ operations to compute if $V^\pi$ is available.

All optimal policies have the same (optimal) action value function $Q^*$. 

Shivaram Kalyanakrishnan (2023)
Markov Decision Problems

1. Action value function

2. Policy iteration
   - Policy improvement
   - Policy improvement theorem and proof
   - Policy iteration algorithm

3. History-dependent and stochastic policies
Given $\pi$, - Pick one or more improvable states, and in these states, - Switch to an arbitrary improving action. Let the resulting policy be $\pi'$. 
Policy Improvement

Given $\pi$, - Pick one or more improvable states, and in these states, - Switch to an arbitrary improving action.

Let the resulting policy be $\pi'$. 

Shivaram Kalyanakrishnan (2023)
Given $\pi$, - Pick one or more improvable states, and in these states, - Switch to an arbitrary improving action. Let the resulting policy be $\pi'$. 

$Q^\pi(s_3, \blacksquare) \leq Q^\pi(s_3, \square)$
Policy Improvement

Given $\pi$,
- Pick one or more improvable states, and in these states,
- Switch to an arbitrary improving action.

Let the resulting policy be $\pi'$.
Policy Improvement

Given $\pi$,
- Pick one or more improvable states, and in these states,
- Switch to an arbitrary improving action.

Let the resulting policy be $\pi'$. 
Policy Improvement

- Given $\pi$, 
- Pick one or more improvable states, and in these states, 
- Switch to an arbitrary improving action.

Let the resulting policy be $\pi'$. 

Improvable states

Improving actions
Policy Improvement

Given \( \pi \),
- Pick one or more improvable states, and in these states,
- Switch to an arbitrary improving action.

Let the resulting policy be \( \pi' \).
Given $\pi$, 
- Pick one or more improvable states, and in these states, 
- Switch to an arbitrary improving action.

Let the resulting policy be $\pi'$. 
Markov Decision Problems

1. Action value function

2. Policy iteration
   - Policy improvement
   - Policy improvement theorem and proof
   - Policy iteration algorithm

3. History-dependent and stochastic policies
Policy Improvement Theorem

For $\pi \in \Pi$, $s \in S$,

$$IA(\pi, s) \overset{\text{def}}{=} \{ a \in A : Q^\pi(s, a) > V^\pi(s) \}.$$
Policy Improvement Theorem

For $\pi \in \Pi, s \in S$,

$$\text{IA}(\pi, s) \overset{\text{def}}{=} \{ a \in A : Q^\pi(s, a) > V^\pi(s) \}.$$ 

For $\pi \in \Pi$,

$$\text{IS}(\pi) \overset{\text{def}}{=} \{ s \in S : |\text{IA}(\pi, s)| \geq 1 \}.$$
For $\pi \in \Pi, s \in S$, 

$$IA(\pi, s) \overset{\text{def}}{=} \{a \in A : Q^{\pi}(s, a) > V^{\pi}(s)\}.$$ 

For $\pi \in \Pi$, 

$$IS(\pi) \overset{\text{def}}{=} \{s \in S : |IA(\pi, s)| \geq 1\}.$$ 

Suppose $IS(\pi) \neq \emptyset$ and $\pi' \in \Pi$ is obtained by policy improvement on $\pi$. Thus, $\pi'$ satisfies 

$$\forall s \in S : [\pi'(s) = \pi(s) \text{ or } \pi'(s) \in IA(\pi, s)] \text{ and } \exists s \in S : \pi'(s) \in IA(\pi, s).$$
Policy Improvement Theorem

For $\pi \in \Pi, s \in S$,

$$IA(\pi, s) \overset{\text{def}}{=} \{a \in A : Q^\pi(s, a) > V^\pi(s)\}.$$ 

For $\pi \in \Pi$,

$$IS(\pi) \overset{\text{def}}{=} \{s \in S : |IA(\pi, s)| \geq 1\}.$$

Suppose $IS(\pi) \neq \emptyset$ and $\pi' \in \Pi$ is obtained by policy improvement on $\pi$. Thus, $\pi'$ satisfies

$$\forall s \in S : [\pi'(s) = \pi(s) \text{ or } \pi'(s) \in IA(\pi, s)] \text{ and } \exists s \in S : \pi'(s) \in IA(\pi, s).$$

Policy Improvement Theorem:
(1) If $IS(\pi) = \emptyset$, then $\pi$ is an optimal policy, else
(2) if $\pi'$ is obtained by policy improvement on $\pi$, then $\pi' \succ \pi$. 

Shivaram Kalyanakrishnan (2023)
Implication of Policy Improvement Theorem

Policy Improvement Theorem:
(1) If $\text{IS}(\pi) = \emptyset$, then $\pi$ is an optimal policy, else
(2) if $\pi'$ is obtained by policy improvement on $\pi$, then $\pi' \succ \pi$. 
Implication of Policy Improvement Theorem

**Policy Improvement Theorem:**

1. If $\text{IS}(\pi) = \emptyset$, then $\pi$ is an optimal policy, else
2. if $\pi'$ is obtained by policy improvement on $\pi$, then $\pi' \succ \pi$.

- If $\pi \in \Pi$ is such that $\text{IS}(\pi) \neq \emptyset$, then there exists $\pi' \in \Pi$ such that $\pi' \succ \pi$. 
Implication of Policy Improvement Theorem

**Policy Improvement Theorem:**

(1) If \( IS(\pi) = \emptyset \), then \( \pi \) is an optimal policy, else

(2) if \( \pi' \) is obtained by policy improvement on \( \pi \), then \( \pi' \succ \pi \).

- If \( \pi \in \Pi \) is such that \( IS(\pi) \neq \emptyset \), then there exists \( \pi' \in \Pi \) such that \( \pi' \succ \pi \).
- But \( \Pi \) has a finite number of policies \( (k^n) \).
Implication of Policy Improvement Theorem

**Policy Improvement Theorem:**

1. If $\text{IS}(\pi) = \emptyset$, then $\pi$ is an optimal policy, else
2. if $\pi'$ is obtained by policy improvement on $\pi$, then $\pi' \succ \pi$.

- If $\pi \in \Pi$ is such that $\text{IS}(\pi) \neq \emptyset$, then there exists $\pi' \in \Pi$ such that $\pi' \succ \pi$.
- But $\Pi$ has a finite number of policies ($k^n$).
- Hence, there must exist a policy $\pi^* \in \Pi$ such that $\text{IS}(\pi^*) = \emptyset$. 
Implication of Policy Improvement Theorem

**Policy Improvement Theorem:**
(1) If $\text{IS}(\pi) = \emptyset$, then $\pi$ is an optimal policy, else
(2) if $\pi'$ is obtained by policy improvement on $\pi$, then $\pi' \succ \pi$.

- If $\pi \in \Pi$ is such that $\text{IS}(\pi) \neq \emptyset$, then there exists $\pi' \in \Pi$ such that $\pi' \succ \pi$.
- But $\Pi$ has a finite number of policies ($k^n$).
- Hence, there must exist a policy $\pi^* \in \Pi$ such that $\text{IS}(\pi^*) = \emptyset$.
- The theorem itself also tells us that $\pi^*$ must be optimal.
Implication of Policy Improvement Theorem

**Policy Improvement Theorem:**
(1) If $\text{IS}(\pi) = \emptyset$, then $\pi$ is an optimal policy, else
(2) if $\pi'$ is obtained by policy improvement on $\pi$, then $\pi' \succ \pi$.

- If $\pi \in \Pi$ is such that $\text{IS}(\pi) \neq \emptyset$, then there exists $\pi' \in \Pi$ such that $\pi' \succ \pi$.
- But $\Pi$ has a finite number of policies ($k^n$).
- Hence, there must exist a policy $\pi^* \in \Pi$ such that $\text{IS}(\pi^*) = \emptyset$.
- The theorem itself also tells us that $\pi^*$ must be optimal.
- Observe that $\text{IS}(\pi^*) = \emptyset \iff B^*(V^{\pi^*}) = V^{\pi^*}$.
Implication of Policy Improvement Theorem

**Policy Improvement Theorem:**
1. If \( IS(\pi) = \emptyset \), then \( \pi \) is an optimal policy, else
2. if \( \pi' \) is obtained by policy improvement on \( \pi \), then \( \pi' \succ \pi \).

- If \( \pi \in \Pi \) is such that \( IS(\pi) \neq \emptyset \), then there exists \( \pi' \in \Pi \) such that \( \pi' \succ \pi \).
- But \( \Pi \) has a finite number of policies \( (k^n) \).
- Hence, there must exist a policy \( \pi^* \in \Pi \) such that \( IS(\pi^*) = \emptyset \).
- The theorem itself also tells us that \( \pi^* \) must be optimal.
- Observe that \( IS(\pi^*) = \emptyset \iff B^*(V^{\pi^*}) = V^{\pi^*} \).
- In other words, \( V^{\pi^*} \) satisfies the Bellman optimality equations—which we know has a unique solution. It is a convention to denote \( V^{\pi^*} \) by \( V^* \).
Bellman Operator $B^\pi$

For $\pi \in \Pi$, we define $B^\pi : \mathbb{R}^n \to \mathbb{R}^n$ as follows.

For $X : S \to \mathbb{R}$ and for $s \in S$,

$$(B^\pi(X))(s) \overset{\text{def}}{=} \sum_{s' \in S} T(s, \pi(s), s') (R(s, \pi(s), s') + \gamma X(s')).$$
Bellman Operator $B_{\pi}$

- For $\pi \in \Pi$, we define $B_{\pi} : \mathbb{R}^n \to \mathbb{R}^n$ as follows.
  
  For $X : S \to \mathbb{R}$ and for $s \in S$,
  
  $$(B_{\pi}(X))(s) \overset{\text{def}}{=} \sum_{s' \in S} T(s, \pi(s), s') (R(s, \pi(s), s') + \gamma X(s')) .$$

- One Bellman operator for each $\pi \in \Pi$. No “max” like $B^*$. 

Some facts about $B_{\pi}$ for all $\pi \in \Pi$. Similar proofs as for $B^*$. 

- $B_{\pi}$ is a contraction mapping with contraction factor $\gamma$.
- For $X : S \to \mathbb{R}$: \( \lim_{l \to \infty} (B_{\pi})^l(X) = V_{\pi} \).
- For $X : S \to \mathbb{R}$, $Y : S \to \mathbb{R}$: $X \succeq Y \Rightarrow B_{\pi}(X) \succeq B_{\pi}(Y)$.
Bellman Operator $B^\pi$

- For $\pi \in \Pi$, we define $B^\pi : \mathbb{R}^n \rightarrow \mathbb{R}^n$ as follows. For $X : S \rightarrow \mathbb{R}$ and for $s \in S$,

$$
(B^\pi(X))(s) \overset{\text{def}}{=} \sum_{s' \in S} T(s, \pi(s), s') (R(s, \pi(s), s') + \gamma X(s')).
$$

- One Bellman operator for each $\pi \in \Pi$. No “max” like $B^\star$.

- Some facts about $B^\pi$ for all $\pi \in \Pi$. Similar proofs as for $B^\star$.
  - $B^\pi$ is a contraction mapping with contraction factor $\gamma$.
  - For $X : S \rightarrow \mathbb{R}$:
    $$
    \lim_{l \rightarrow \infty} (B^\pi)^l(X) = V^\pi.
    $$
  - For $X : S \rightarrow \mathbb{R}, Y : S \rightarrow \mathbb{R}$: $X \succeq Y \implies B^\pi(X) \succeq B^\pi(Y)$. 

Observe that for $\pi, \pi' \in \Pi$, $\forall s \in S$:
$$
B^\pi(V^\pi)(s) = Q^\pi(s, \pi'(s)).
$$
**Bellman Operator** $B^\pi$

- For $\pi \in \Pi$, we define $B^\pi : \mathbb{R}^n \rightarrow \mathbb{R}^n$ as follows. For $X : S \rightarrow \mathbb{R}$ and for $s \in S$,
  $$(B^\pi(X))(s) \overset{\text{def}}{=} \sum_{s' \in S} T(s, \pi(s), s') (R(s, \pi(s), s') + \gamma X(s')).$$

- One Bellman operator for each $\pi \in \Pi$. No “max” like $B^\star$.

- Some facts about $B^\pi$ for all $\pi \in \Pi$. Similar proofs as for $B^\star$.
  - $B^\pi$ is a contraction mapping with contraction factor $\gamma$.
  - For $X : S \rightarrow \mathbb{R}$:
    $$\lim_{l \rightarrow \infty} (B^\pi)^l(X) = V^\pi.$$
  - For $X : S \rightarrow \mathbb{R}$, $Y : S \rightarrow \mathbb{R}$: $X \succeq Y \implies B^\pi(X) \succeq B^\pi(Y)$.

- Observe that for $\pi, \pi' \in \Pi$, $\forall s \in S$: $B^\pi'(V^\pi)(s) = Q^\pi(s, \pi'(s))$. 

Shivaram Kalyanakrishnan (2023)
Proof of Policy Improvement Theorem

\[ \text{IS}(\pi) = \emptyset \]
Proof of Policy Improvement Theorem

$$\text{IS}(\pi) = \emptyset \implies \forall \pi' \in \Pi : V^\pi \succeq B^\pi'(V^\pi)$$
Proof of Policy Improvement Theorem

\[ IS(\pi) = \emptyset \implies \forall \pi' \in \Pi : V^\pi \succeq B^\pi'(V^\pi) \]

\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^\pi'(V^\pi) \succeq (B^\pi')^2(V^\pi) \]
Proof of Policy Improvement Theorem

\[ \text{IS}(\pi) = \emptyset \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \succeq \cdots \succeq \lim_{l \to \infty} (B^{\pi'})^l(V^\pi) \]

Shivaram Kalyanakrishnan (2023)
CS 747, Autumn 2023
Proof of Policy Improvement Theorem

\[ IS(\pi) = \emptyset \implies \forall \pi' \in \Pi : V^\pi \succeq B^\pi'(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^\pi'(V^\pi) \succeq (B^\pi')^2(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^\pi'(V^\pi) \succeq (B^\pi')^2(V^\pi) \succeq \cdots \succeq \lim_{l \to \infty} (B^{\pi'})^l(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq V^{\pi'} . \]
Proof of Policy Improvement Theorem

\[ IS(\pi) = \emptyset \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \succeq \cdots \succeq \lim_{l \to \infty} (B^{\pi'})^l(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq V^{\pi'} . \]

\[ IS(\pi) \neq \emptyset ; \pi \xrightarrow{\text{P.I.}} \pi' \]}
Proof of Policy Improvement Theorem

\[ \text{IS}(\pi) = \emptyset \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \succeq \cdots \succeq \lim_{l \to \infty} (B^{\pi'})^l(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq V^{\pi'} \]

\[ \text{IS}(\pi) \neq \emptyset; \pi \xrightarrow{\text{P.I.}} \pi' \implies B^{\pi'}(V^\pi) \succ V^\pi \]
Proof of Policy Improvement Theorem

\[ IS(\pi) = \emptyset \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \]

\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \]

\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \succeq \cdots \succeq \lim_{l \to \infty} (B^{\pi'})^l(V^\pi) \]

\[ \implies \forall \pi' \in \Pi : V^\pi \succeq V^{\pi'} \]

\[ IS(\pi) \neq \emptyset ; \pi \xrightarrow{P,I.} \pi' \implies B^{\pi'}(V^\pi) \succ V^\pi \]

\[ \implies (B^{\pi'})^2(V^\pi) \succeq B^{\pi'}(V^\pi) \succ V^\pi \]
Proof of Policy Improvement Theorem

\[ IS(\pi) = \emptyset \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \succeq \cdots \succeq \lim_{l \to \infty} (B^{\pi'})^l(V^\pi) \]
\[ \implies \forall \pi' \in \Pi : V^\pi \succeq B^{\pi'}(V^\pi) \succeq (B^{\pi'})^2(V^\pi) \succeq \cdots \succeq \lim_{l \to \infty} (B^{\pi'})^l(V^\pi) \succeq B^{\pi'}(V^\pi) \succeq V^\pi \]

\[ IS(\pi) \neq \emptyset ; \pi \xrightarrow{P.I.} \pi' \implies B^{\pi'}(V^\pi) \succ V^\pi \]
\[ \implies (B^{\pi'})^2(V^\pi) \succeq B^{\pi'}(V^\pi) \succ V^\pi \]
\[ \implies \lim_{l \to \infty} (B^{\pi'})^l(V^\pi) \succeq \cdots \succeq (B^{\pi'})^2(V^\pi) \succeq B^{\pi'}(V^\pi) \succ V^\pi \]
Proof of Policy Improvement Theorem

\[
\text{IS}(\pi) = \emptyset \implies \forall \pi' \in \Pi : V^\pi \succeq B^\pi'(V^\pi) \\
\implies \forall \pi' \in \Pi : V^\pi \succeq B^\pi'(V^\pi) \succeq (B^\pi')^2(V^\pi) \\
\implies \forall \pi' \in \Pi : V^\pi \succeq B^\pi'(V^\pi) \succeq (B^\pi')^2(V^\pi) \succeq \cdots \succeq \lim_{l \to \infty} (B^\pi')^l(V^\pi) \\
\implies \forall \pi' \in \Pi : V^\pi \succeq V^\pi'.
\]

\[
\text{IS}(\pi) \neq \emptyset; \pi \xrightarrow{\text{P.I.}} \pi' \implies B^\pi'(V^\pi) \succ V^\pi \\
\implies (B^\pi')^2(V^\pi) \succeq B^\pi'(V^\pi) \succ V^\pi \\
\implies \lim_{l \to \infty} (B^\pi')^l(V^\pi) \succeq \cdots \succeq (B^\pi')^2(V^\pi) \succeq B^\pi'(V^\pi) \succ V^\pi \\
\implies V^\pi' \succ V^\pi.
\]
Markov Decision Problems

1. Action value function

2. Policy iteration
   - Policy improvement
   - Policy improvement theorem and proof
   - Policy iteration algorithm

3. History-dependent and stochastic policies
Policy Iteration Algorithm

\( \pi \leftarrow \text{Arbitrary policy.} \)

\textbf{While} \( \pi \) has improvable states:

\( \pi' \leftarrow \text{PolicyImprovement}(\pi) \).

\( \pi \leftarrow \pi' \).

\textbf{Return} \( \pi \).
Policy Iteration Algorithm

\[ \pi \leftarrow \text{Arbitrary policy.} \]

\textbf{While} \( \pi \) has improvable states:

\[ \pi' \leftarrow \text{PolicyImprovement}(\pi). \]

\[ \pi \leftarrow \pi'. \]

\textbf{Return} \( \pi \).
Policy Iteration Algorithm

\( \pi \leftarrow \text{Arbitrary policy.} \)

\textbf{While} \( \pi \) has improvable states:

\( \pi' \leftarrow \text{PolicyImprovement}(\pi). \)

\( \pi \leftarrow \pi'. \)

\textbf{Return} \( \pi. \)
\( \pi \leftarrow \text{Arbitrary policy.} \)

**While** \( \pi \) has improvable states:

\[ \pi' \leftarrow \text{PolicyImprovement}(\pi). \]

\[ \pi \leftarrow \pi'. \]

**Return** \( \pi \).
Policy Iteration Algorithm

\[ \pi \leftarrow \text{Arbitrary policy.} \]

**While** \( \pi \) has improvable states:

\[ \pi' \leftarrow \text{PolicyImprovement}(\pi). \]

\[ \pi \leftarrow \pi'. \]

**Return** \( \pi \).
Policy Iteration Algorithm

\[ \pi \leftarrow \text{Arbitrary policy.} \]

**While** \( \pi \) has improvable states:

\[ \pi' \leftarrow \text{PolicyImprovement}(\pi). \]

\[ \pi \leftarrow \pi'. \]

**Return** \( \pi. \)
Policy Iteration Algorithm

\[ \pi \leftarrow \text{Arbitrary policy.} \]

\textbf{While} \( \pi \) has improvable states:

\[ \pi' \leftarrow \text{PolicyImprovement}(\pi). \]

\[ \pi \leftarrow \pi'. \]

\textbf{Return} \( \pi \).
Policy Iteration Algorithm

$$\pi \leftarrow \text{Arbitrary policy.}$$

**While** $$\pi$$ has improvable states:

$$\pi' \leftarrow \text{PolicyImprovement}(\pi).$$

$$\pi \leftarrow \pi'.$$

**Return** $$\pi.$$  

Path taken (and hence the number of iterations) in general depends on the **switching strategy**.
Markov Decision Problems

1. Action value function

2. Policy iteration
   - Policy improvement
   - Policy improvement theorem and proof
   - Policy iteration algorithm

3. History-dependent and stochastic policies
A More General Class of Policies

- In principle, an agent can follow a policy $\lambda$ that maps every possible history $s^0, a^0, r^0, s^1, a^1, r^1, \ldots, s^t$ for $t \geq 0$ to a probability distribution over $A$.

- Let $\Lambda$ be the set of such policies $\lambda$ (which are in general non-Markovian, non-stationary, and stochastic).
A More General Class of Policies

- In principle, an agent can follow a policy $\lambda$ that maps every possible history $s^0, a^0, r^0, s^1, a^1, r^1, \ldots, s^t$ for $t \geq 0$ to a probability distribution over $A$.

- Let $\Lambda$ be the set of such policies $\lambda$ (which are in general non-Markovian, non-stationary, and stochastic).

- Recall that we only considered $\Pi$, the set of all policies $\pi : S \rightarrow A$ (which are Markovian, stationary, and deterministic). Observe that $\Pi \subset \Lambda$.

- We have shown that there exists $\pi^* \in \Pi$ such that for all $\pi \in \Pi$, $\pi^* \succeq \pi$. 
A More General Class of Policies

- In principle, an agent can follow a policy $\lambda$ that maps every possible history $s^0, a^0, r^0, s^1, a^1, r^1, \ldots, s^t$ for $t \geq 0$ to a probability distribution over $A$.
- Let $\Lambda$ be the set of such policies $\lambda$ (which are in general non-Markovian, non-stationary, and stochastic).
- Recall that we only considered $\Pi$, the set of all policies $\pi : S \rightarrow A$ (which are Markovian, stationary, and deterministic). Observe that $\Pi \subset \Lambda$.
- We have shown that there exists $\pi^* \in \Pi$ such that for all $\pi \in \Pi$, $\pi^* \succeq \pi$.

Could there exist $\lambda \in \Lambda \setminus \Pi$ such that $\neg (\pi^* \succeq \lambda)$?
A More General Class of Policies

- In principle, an agent can follow a policy $\lambda$ that maps every possible history $s^0, a^0, r^0, s^1, a^1, r^1, \ldots, s^t$ for $t \geq 0$ to a probability distribution over $A$.

- Let $\Lambda$ be the set of such policies $\lambda$ (which are in general non-Markovian, non-stationary, and stochastic).

- Recall that we only considered $\Pi$, the set of all policies $\pi : S \rightarrow A$ (which are Markovian, stationary, and deterministic). Observe that $\Pi \subset \Lambda$.

- We have shown that there exists $\pi^* \in \Pi$ such that for all $\pi \in \Pi$, $\pi^* \succeq \pi$.

Could there exist $\lambda \in \Lambda \setminus \Pi$ such that $\neg(\pi^* \succeq \lambda)$? No.
History and Stochasticity

- In MDPs, the agent can sense state, and the consequence of each action depends solely on state.

- We are maximizing an infinite sum of expected discounted rewards—the challenge at each time step is to maximize the expected infinite discounted reward starting from the current state!

- History and stochasticity can help if the agent is unable to sense state perfectly. Such a situation arises in an abstraction called the Partially Observable MDP (POMDP).

- Optimal policies for the finite horizon reward setting are in general non-stationary (time-dependent).

- Optimal policies ("strategies") in many types of multi-player games are in general stochastic ("mixed") because the next state depends on all the players' actions, but each player chooses only their own.
History and Stochasticity

- In MDPs, the agent can sense state, and the consequence of each action depends solely on state.
- We are maximising an infinite sum of expected discounted rewards—the challenge at each time step is the same: to maximise the expected infinite discounted reward starting from the current state!
History and Stochasticity

- In MDPs, the agent can sense state, and the consequence of each action depends solely on state.

- We are maximising an infinite sum of expected discounted rewards—the challenge at each time step is the same: to maximise the expected infinite discounted reward starting from the current state!

- History and stochasticity can help if the agent is unable to sense state perfectly. Such a situation arises in an abstraction called the Partially Observable MDP (POMDP).
History and Stochasticity

- In MDPs, the agent can sense state, and the consequence of each action depends solely on state.
- We are maximising an infinite sum of expected discounted rewards—the challenge at each time step is the same: to maximise the expected infinite discounted reward starting from the current state!

- History and stochasticity can help if the agent is unable to sense state perfectly. Such a situation arises in an abstraction called the Partially Observable MDP (POMDP).
- Optimal policies for the finite horizon reward setting are in general non-stationary (time-dependent).
History and Stochasticity

- In MDPs, the agent can sense **state**, and the consequence of each action depends solely on state.
- We are maximising an **infinite** sum of **expected** discounted rewards—the challenge at each time step is the same: to maximise the expected infinite discounted reward starting from the current state!

- History and stochasticity can help if the agent is unable to sense state perfectly. Such a situation arises in an abstraction called the Partially Observable MDP (POMDP).
- Optimal policies for the **finite horizon** reward setting are in general non-stationary (time-dependent).
- Optimal policies (“strategies”) in many types of **multi-player games** are in general stochastic (“mixed”) because the next state depends on all the players’ actions, but each player chooses only their own.
Markov Decision Problems

1. Action value function

2. Policy iteration
   - Policy improvement
   - Policy improvement theorem and proof
   - Policy iteration algorithm

3. History-dependent and stochastic policies
Markov Decision Problems

1. Action value function

2. Policy iteration
   - Policy improvement
   - Policy improvement theorem and proof
   - Policy iteration algorithm

3. History-dependent and stochastic policies

Next class: Running time of policy iteration, review of MDP planning.