Peter Bartlett

1. Recall: MDPs.
2. Value iteration.
3. Policy iteration.
4. Linear programming formulation.
5. Q: state-action utility function.
Definition: A Markov Decision Process (MDP) consists of

1. A state space $\mathcal{X}$,
2. An action space $\mathcal{A}$,
3. A set of Markov chains, $\mathcal{M} = (\mathcal{X}, P_a)$, one for each $a \in \mathcal{A}$,
4. A reward distribution $R: \mathcal{X} \times \mathcal{A} \rightarrow \Delta(\mathbb{R})$.

A policy is a sequence of functions $\pi_t: \mathcal{X} \rightarrow \Delta(\mathcal{A})$, one for each time $t$. (A stationary policy is constant with $t$.)
Recall: Value iteration

**Definition:** Define the operator $T : \mathbb{R}^X \rightarrow \mathbb{R}^X$ by

$$(TJ)(x) = \max_{a \in \mathcal{A}} \mathbb{E} \left[ r_0 + \alpha J(x_1) \mid x_0 = x, a_0 = a \right].$$

**Theorem:** For any $\alpha < 1$, there is a vector $J^* \in \mathbb{R}^X$ such that

1. For all $J \in \mathbb{R}^X$, $J^* = \lim_{k \to \infty} T^k J$.
2. $J^*$ is the unique solution to $J = TJ$.
3. $J^* = \max_{\pi} J^\pi$, where the max is over stationary (or non-stationary) policies $\pi$.
4. $J^* = J^{\pi^*}$, where

$$\pi^*(x) = \arg \max_{a \in \mathcal{A}} \mathbb{E} \left[ r_0 + \alpha J^*(x_1) \mid x_0 = x, a_0 = a \right].$$
Greedy policy

Notice that $\pi^*$ is the greedy choice with respect to the value function $J^*$.

**Definition:** For a value function estimate $\hat{J} \in \mathbb{R}^X$, the corresponding greedy policy is $\pi = G\hat{J}$, where we define the greedy operator $G : \mathbb{R}^X \rightarrow \mathcal{A}^X$:

$$(G\hat{J})(x) := \arg \max_{a \in \mathcal{A}} \mathbb{E} \left[ r_0 + \alpha \hat{J}(x_1) \middle| x_0 = x, a_0 = a \right].$$

It’s easy to show:

**Lemma:** For a value function estimate $\hat{J} \in \mathbb{R}^X$, if $\pi = G\hat{J}$, $J^*$ and $\hat{J}$ are bounded by

$$\| J^* - J^\pi \|_\infty \leq \frac{2\alpha}{1 - \alpha} \| J^* - \hat{J} \|_\infty.$$
Value iteration and (generalized) policy iteration

Value iteration:

\[ \hat{J}_{k+1} := T \hat{J}_k, \quad \pi_{k+1} := G \hat{J}_{k+1}. \]

Policy iteration:

\[ \pi_{k+1} := G J^{\pi_k}. \]

Generalized policy iteration:

\[ J_{k+1} := T_{\pi_k}^l J_k, \quad \pi_{k+1} := G J_{k+1}. \]
Theorem:

Policy iteration generates a sequence of policies with distinct, increasing values, terminating after a finite number of iterations with an optimal policy, that is, for some $k$,

$$J^\pi_0 \leq J^\pi_1 \leq \ldots \leq J^\pi_k = J^*.$$ 

Generalized policy iteration generates a sequence of policies with $J_k \to J^*$. 

(Generalized) policy iteration
Bellman equations:

\[ J = T \cdot J. \]

Linear programming formulation:
Fix a probability distribution \( p \) with support \( \mathcal{X} \).

\[
\begin{align*}
\min_{J} & \quad p^T J \\
\text{s.t.} & \quad J \geq T \cdot J.
\end{align*}
\]
Proof. Uses monotonicity: \( J \geq J' \) implies \( TJ \geq TJ' \). So \( J \geq TJ \) implies \( J \geq T^k J \rightarrow J^* \). Minimizing \( \mu^T J \) sets \( J = J^* \). \( \square \)
Dual linear program

\[
\max_\mu \sum_{x \in \mathcal{X}} \sum_{a \in \mathcal{A}} \mu(x, a) \mathbb{E}[r_0 | x_0 = x, a_0 = a]
\]

s.t. \(\forall x' \in \mathcal{X}, \sum_{a \in \mathcal{A}} \mu(x', a) = p(x)\)

\[+ \alpha \sum_{x \in \mathcal{X}} \sum_{a \in \mathcal{A}} \mu(x, a) P[x_1 = x' | x_0 = x, a_0 = a].\]

View \(\lambda\) as discounted expected number of state-action visits, starting from the distribution \(p\). So criterion is expected discounted reward.

Primal-dual are related via optimal policy: \(\pi^*(x) = \arg \max_{a \in \mathcal{A}} \lambda(x, a)\).
Analogous to $J^*$:

$$Q^*(x, a) := \mathbb{E} \left[ r_0 + \alpha \max_{a' \in \mathcal{A}} Q^*(x', a') \right| x_0 = x, a_0 = a],$$

$$\pi^*(x) := \arg \max_{a \in \mathcal{A}} Q^*(x, a).$$

**Q iteration:**

$$\hat{Q}_{k+1}(x, a) := \mathbb{E} \left[ r_0 + \alpha \max_{a' \in \mathcal{A}} \hat{Q}_k(x', a') \right| x_0 = x, a_0 = a],$$

$$\pi_{k+1}(x) := \arg \max_{a \in \mathcal{A}} \hat{Q}_{k+1}(x, a).$$