Dynamic Programming

There are two types of Bellman equations:
First, one is called Bellman Expectation Equation:

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s',r'} p(s', r|s, a)[r + \gamma v_\pi(s')]$$

Or written in iterative fashion:

$$v_{i+1}(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + \gamma v_i(s')]$$

Note that the value function for next state s' comes from the previous iteration, aka, *sweep*. This becomes a system of $|S|$ simultaneous linear equations in $|S|$ unknowns. This gives us a value function for an arbitrary policy π as per the policy evaluation. We may then want to know if there is a policy π' that is better than our current policy.
Depending on the type of policy and reward function definition, we have different Bellman expectation equations, as summarized in this picture:

Summary of the Equations

1. **Deterministic Policy with $R(s,a)$:**
   $V^\pi(s) = R(s, \pi(s)) + \gamma \sum_{s'} P(s' \mid s, \pi(s)) V^\pi(s')$
2. **Stochastic Policy with $R(s,a)$:**
   $V^\pi(s) = \sum_a \pi(a \mid s) \left[R(s,a) + \gamma \sum_{s'} P(s' \mid s, a) V^\pi(s')\right]$
3. **Deterministic Policy with $R(s,a,s')$:**
   $V^\pi(s) = \sum_{s'} P(s' \mid s, \pi(s)) \left[R(s, \pi(s), s') + \gamma V^\pi(s')\right]$
4. **Stochastic Policy with $R(s,a,s')$:**
   $V^\pi(s) = \sum_a \pi(a \mid s) \sum_{s'} P(s' \mid s, a) \left[R(s,a,s') + \gamma V^\pi(s')\right]$

Optimal Policies and Value Functions

A way of evaluating this is by taking a new action a in state s that is not in our current policy, running our policy thereafter and seeing how the value function changes. Formally that looks like:

$$q_\pi(s, a) = \sum_{s',r} p(s', r|s, a)[r + max_a \gamma v_\pi(s')]$$

If taking this new action in state s produces a value function that is greater than or equal to the previous value function for all states then we say the policy π' is an improvement over π:

$$v'_\pi(s) \geq v_\pi \forall s \in S$$

This is known as **Policy improvement theorem**. One way of choosing such new actions for policy improvement is by acting greedily w.r.t the value function.

Bellman Optimality Equation for V:

$$v^{\pi^*}(s) = max_{a \in A(s)} \sum_{s',r} p(s',r|s,a)[r + \gamma v^{\pi^*}(s')]$$

or alternatively, if you take the immediate reward outside the sum(if it doesn't spend on next state)

$$v^{\pi^*}(s) = max_{a \in A(s)}(r(s,a) + \gamma \sum_{s',r} p(s',r|s,a) v^{\pi^*}(s'))$$

Generalized Policy Iteration

There are two components that make up both policy iteration and value iteration:

1. Policy evaluation (model-free prediction)
   Evaluate policy π using Bellman Expectation Equation
2. Policy improvement (model-free control)
   Improve the policy by acting greedily with respect to V^π:

   $$\pi' = greedy(V^\pi)$$

   Then, we have Policy Iteration and Value Iteration:

Policy Iteration

1. Evaluate policy π to obtain value function V_π until convergence.
2. Improve policy π by acting greedily with respect to V_π to obtain new policy π'
3. Evaluate new policy π' to obtain new value function $V_{\pi'}$
4. Repeat until new policy is no longer better than old policy.

Value Iteration

Instead of doing infinite sweeps of the state space to approach the true value function, stop after one *sweep* and do policy improvement. Value iteration is achieved by turning the Bellman optimality equation into an update rule:

$$v_{i+1}(s) = argmax_a \sum_{s',r} p(s'r|s,a)[r + \gamma v_k(s')]$$

for all $s \in S$. Value iteration effectively combines, in each sweep, one sweep of policy evaluation and one sweep of policy improvement.

DP methods are guaranteed to find optimal solutions for Q and V in polynomial time and are exponentially faster than direct search.

Questions

Q1. Given an optimal state-value function V_π and without a dynamics model P, can we derive an optimal policy?
A1: Look at the policy improvement (control) update rule: we need access to the dynamics model to do greedy policy search !
Q2. Imagine, instead, we had state-action value function Q_π, without a dynamic model, can we derive an optimal policy?
A2: Yes, we can just do greedy policy improvement by

$$\pi^* = argmax_a Q(s, a)$$

Q3: Can we have multiple optimal policies for a given MDP?
A3: There can be more than one optimal policy for a given value function: this only happens when two actions have the same value in a given state. Nevertheless, both policies lead to the same **expected return**
Q4: What is the relationship between Q and V functions?
A4: $V(s) = E_{a \sim \pi} q_\pi(s, a)$. If the agent acts according to a given policy, the expected value of q-values should sum up to value of that state.

$$V^\pi(s) = \sum_a \pi(a|s) Q^\pi(s, a)$$

Alternatively,

$$Q^\pi(s, a) = \sum_{s' \in S} P(s'|s, a)[R(s, a, s') + \gamma V^\pi(s')]$$