CS 520: Value Iteration and Contraction Mapping

16:198:520
Instructor: Wes Cowan

We can formulate a Markov Decision Process in the following way: for a finite state space S, for every state $s \in S$ we have a finite action set $A(s)$, where taking action $a \in A(s)$ while in state s will earn reward $r_{s,a}$ (or potentially cost, if the reward is negative), and transition the system to a new state. From state s, taking action a will transition to state s' with probability $p^a_{s,s'}$. Note, $p^a_{s,s'} = 0$ if it is impossible to go from s to s' under that action, and over all,

$$\sum_{s' \in S} p^a_{s,s'} = 1. \tag{1}$$

In every time step, based on the current state (starting at some initial state), the controller must decide which of the available actions to take, will collect the corresponding reward, and transition to the next state. A policy π specifies for each state an action to take in that state, $\pi(s) \in A(s)$. If the sequence of states (necessarily random) is denoted $S_0, S_1, S_2, \ldots$, the *infinite horizon expected discounted utility* starting from state s under policy π is denoted by

$$U_\pi(s) = E\left[\sum_{t=0}^{\infty} \beta^t r_{S_t,\pi(S_t)} | S_0 = s_0\right]. \tag{2}$$

That is, at every time step, starting from state s, the 'value' of policy π is the immediate reward of the action taken under π, plus the expected future reward based on whatever state the system transitions to, discounted by a factor of $0 < \beta < 1$. Note the following relation based on the recursive structure of the utility, for every state s:

$$U_\pi(s) = r_{s,\pi(s)} + \beta + \sum_{s'} p^{\pi(s)}_{s,s'} U_\pi(s'). \tag{3}$$

Note, for a given policy π, the above system of linear equations may be solved to yield $U_\pi(s)$ for each state $s \in S$.

The discounting is introduced both to model certain economic realities (inflation, etc), and to render the 'utility' of a given policy finite over an infinite horizon. Note we have the following bound: if $r_{s,a} \leq R$ for all s, a,

$$U_\pi(s) \leq E\left[\sum_{t=0}^{\infty} \beta^t R | S_0 = s_0\right] \leq \frac{R}{1-\beta}. \tag{4}$$

The fact that the utility of a policy is now finite means that the value of policies can be compared. We can define an 'optimal' utility in the following way:

$$U^*(s) = \max_\pi U_\pi(s). \tag{5}$$

For any state s, the quantity $U^*(s)$ represents the greatest achievable utility starting in state s under any policy. Given the optimal utility, we may define the optimal policy in the following way:

$$\pi^*(s) = \text{argmax}_{a \in A(s)} \left[r_{s,a} + \beta \sum_{s'} p^a_{s,s'} U^*(s')\right]. \tag{6}$$

If $U^*(s')$ represents the value of performing 'optimally' from state s' onwards, the optimal action can be determined by comparing the immediate value of any available action, plus the discounted expected 'optimal' utility from that point. Similarly, given the optimal policy, we can recover the optimal utility function, based on:

$$U_{\pi^*}(s) = r_{s,\pi^*(s)} + \beta + \sum_{s'} p^{\pi^*(s)}_{s,s'} U_{\pi^*}(s'). \tag{7}$$

This is a nice sort of duality - solving for U^* can give π^*, and solving for π^* can give U^* - but how to solve for either of these things? One key point is the so called **Bellman's Equations**, a system of recurrence equations that specify the optimal utility function: for each $s \in S$,

$$U^*(s) = \max_{a \in A(s)} \left[r_{s,a} + \beta \sum_{s'} p^a_{s,s'} U^*(s')\right]. \tag{8}$$

If this system could be solved for U^* for each state, we could recover the optimal policy as indicated above. This system has a very natural interpretation - the optimal utility at any state s can be determined by comparing the value of taking any given action a and then performing optimally past that point (thus earning $U^*(s')$). However, solving this system is difficult, as the max function introduces significant nonlinearities into the system.

Value Iteration

Under the technique of Value Iteration, we construct an initial estimate or guess for U^*, i.e., select some value $U_0^*(s)$ for each state $s \in S$. We iteratively improve this estimate with the following iteration:

$$U_{k+1}^*(s) = \max_{a \in A(s)} \left[r_{s,a} + \beta \sum_{s'} p_{s,s'}^a U_k^*(s') \right]. \tag{9}$$

This can be viewed as an iterated update version of Bellman's equations above. The significance of this scheme is that the successive approximations U_k^* converge to U^* in the following way:

$$\max_{s \in S} |U_k^*(s) - U^*(s)| \to 0 \text{ as } k \to \infty. \tag{10}$$

Once this scheme has been utilized, to compute U^* to sufficient accuracy, the optimal policy can be recovered as previously discussed.

Three important questions are as follows: a) is convergence guaranteed, b) what is the rate of convergence, and c) how should the initial estimates U_0^* be chosen? These are all connected, based on the notion of *contraction mappings*. The iteration scheme above is what's known as a contraction, effectively 'squeezing' the estimate U_k^* down onto the true value U^*. To see this, consider defining the following error measure:

$$\epsilon_k = \max_{s \in S} |U_k^*(s) - U^*(s)|. \tag{11}$$

We can relate the error at time $k+1$ to the error at time k in the following way: for any state s, we have

$$\begin{aligned}
U_{k+1}^*(s) &= \max_{a \in A(s)} \left[r_{s,a} + \beta \sum_{s'} p_{s,s'}^a U_k^*(s') \right] \\
&\leq \max_{a \in A(s)} \left[r_{s,a} + \beta \sum_{s'} p_{s,s'}^a \left(U^*(s') + \epsilon_k \right) \right] \\
&= \max_{a \in A(s)} \left[r_{s,a} + \beta \sum_{s'} p_{s,s'}^a U^*(s') + \beta \sum_{s'} p_{s,s'}^a \epsilon_k \right] \\
&= \max_{a \in A(s)} \left[r_{s,a} + \beta \sum_{s'} p_{s,s'}^a U^*(s') + \beta \epsilon_k \right] \\
&= \max_{a \in A(s)} \left[r_{s,a} + \beta \sum_{s'} p_{s,s'}^a U^*(s') \right] + \beta \epsilon_k \\
&= U^*(s) + \beta \epsilon_k
\end{aligned} \tag{12}$$

Hence,

$$|U_{k+1}^*(s) - U^*(s)| \leq \beta \epsilon_k. \tag{13}$$

Note, we get immediately from this that

$$\epsilon_{k+1} \leq \beta \epsilon_k. \tag{14}$$

Substituting backwards, we ultimately get that

$$\max_{s \in S} |U_k^*(s) - U^*(s)| \leq \beta^k \max_{s \in S} |U_0^*(s) - U^*(s)| \tag{15}$$

What this demonstrates is that in every step of the iteration, the overall error is contracted, reduced by at least a factor of $\beta < 1$, *regardless* of the initial error on U_0^*. We get from this that a) the error on successive estimates converges to 0, b) convergence is exponential in terms of β (which may or may not be good, depending on the relative size of β), and c) *any* initial estimate U_0^* will result in convergence.