EE 223: Stochastic Estimation and Control **Spring 2007**

Lecture 5 — January 30

Lecturer: Venkat Anantharam *Scribe: Maryam Kamgarpour*

5.1 Secretary Problem

The problem set-up is explained in Lecture 4. We review the notation and then study the optimal solution.

Notation
Let M be the total number of secretaries. The set-up is over the duration of time 0 through N, where $N = M + 1$.
For the state space we have $x_0 \in \{0\}$, a dummy state,
$x_k \in \{(\Delta, k, 1), (\Delta, k, 0), (k, 1), (k, 0)\}$,
$x_N \in \{T\}$ a terminal state.

In the above, Δ indicates that the secretary was picked earlier, k refers to the index of the secretary currently being considered, $(\Delta, k, 1)$ (resp. $(\Delta, k, 0)$) means the secretary picked is the best (resp. not the best) of the k secretaries so far, and $(k, 1)$ (resp. $(k, 0)$) means that the secretary currently being considered is the best (resp. not the best) of the k secretaries so far. The possible control actions at each non-terminal state are: $u = 0$, which for non-Δ states means pick the current secretary and which leads to a Δ state; and $u = 1$, which for non-Δ states means don't pick the current secretary and leads to a non-Δ state. In Δ states, the control action is irrelevant. The problem can be put into our canonical framework via independent $\{0, 1\}$-valued random variables $w_0, w_1, \ldots, w_{N-1}$, as discussed in Lecture 4.

DP Recursion evaluates the reward-to-go function:

$J_N(T) = 0$
$J_k(\Delta, k, 1) = \frac{k}{M}, \quad k = 1, \ldots, N-1$. This is the probability that the secretary who was picked, and who happens to be the best among the first k secretaries (this is what it means to be in state $(\Delta, k, 1)$) is actually the best overall.
$J_k(\Delta, k, 0) = 0, \quad k = 1, \ldots, N-1$. This is because if the secretary that was picked is not the best among the first k secretaries, he or she cannot possibly be the best overall.
$J_k(k, 1) = max\{\frac{k}{M}, \frac{1}{k+1}J_{k+1}(k+1, 1) + \frac{k}{k+1}J_{k+1}(k+1, 0)\}$ where $\frac{1}{k+1}$ corresponds to probability that the secretary at time $k+1$ is better than current best secretary at time k and hence better than all previous ones. In this maximum, the first term corresponds to the choice $u = 0$ of picking the current secretary, and the second term corresponds to the choice $u = 1$ of deciding to keep interviewing secretaries.
$J_k(k, 0) = max\{0, \frac{1}{k+1}J_{k+1}(k+1, 1) + \frac{k}{k+1}J_{k+1}(k+1, 0)\}$. Here again in the maximum, the first term corresponds to the choice $u = 0$ of picking the current secretary, and the second

term corresponds to the choice $u = 1$ of deciding to keep interviewing secretaries.
$J_0(0,0) = max_{\{}0, J_1(1,1)\}$. To understand the second term in the max, note that the first secretary seen will always be the best so far.

Observations

1. In state $(k,0)$ reward $u = 1$ is an optimizer. This can seen from the update equation for $J(k,0)$ by noting that the reward-to-go functions are nonnegative. The intuitive meaning of this observation is that if the current secretary is not the best so far, you won't gain anything by choosing this person, but you may have a chance of choosing the best one if you play along. In fact $u = 1$ can be seen to be the unique optimizer in state $(k,0)$ for $0 \leq k \leq N-2$, while in state $(N-1,0)$ either control action is an optimizer.

2. if $J_k(k,1) > \frac{k}{M}$ then $J_{k-1}(k-1,1) > \frac{k-1}{M}$.
Derivation of the above:
$J_k(k,1) > \frac{k}{M} \Leftrightarrow \frac{1}{k+1}J_{k+1}(k+1,1) + \frac{k}{k+1}J_{k+1}(k+1,0) > \frac{k}{M}$
$\Leftrightarrow J_k(k,0) > \frac{k}{M} \Rightarrow J_{k-1}(k-1,1) = max\{\frac{k-1}{M}, \frac{1}{k}J_k(k,1) + \frac{k-1}{k}J_k(k,0)\}$
$> \frac{1}{M} + \frac{k-1}{M} \Rightarrow J_{k-1}(k-1,1) > \frac{k}{M} > \frac{k-1}{M}$.

This result confirms the intuition that if $u = 1$ (don't pick current secretary) is an optimizer in state $(k,1)$, it must have also been an optimizer in states $(l,1)$ for all $0 \leq l \leq k$.

3. Based on above, it is seen that the optimal strategy is that there exists some threshold time L that one would let go of the first L-1 secretaries and pick the first best one afterward. Hence, the optimal Markov strategy is of the following type:
1. If the state is 0 chose $u = 1$.
2. If current state is $(k,0)$ choose $u = 1\ \forall k = 1, \ldots, N-1$.
3. If current state is $(k,1)$ and $k < L$ pick $u = 1$. If current state is $(k,1)$ and $k \geq L$ choose $u = 0$.

Evaluating the Threshold
We look for L to maximize the following:
$\sum_{k=L}^{M}$ P(k^{th} secretary is the best and you have selected this person)
$= \sum_{k=L}^{M} \frac{1}{M}\frac{L-1}{k-1} = \frac{L-1}{M}(\frac{1}{L-1} + \cdots + \frac{1}{M-1})$.

To understand the expression $\frac{1}{M}\frac{L-1}{k-1}$ that is the k-th term in the summation above, $L \leq k \leq M$, note that $\frac{1}{M}$ is the probability that the k^{th} secretary is the absolute best, and that if we condition on this event then the relative ordering of all the other secretaries is uniformly distributed. Now, with this threshold strategy we will end up picking the absolute best secretary precisely if at times L through $k-1$ we are not fooled into picking the current best secretary. Since $\frac{L-1}{k-1}$ is the probability that the best among secretaries $1 \ldots L-1$ occurred at one of the times $1 \ldots k-1$, this is precisely the conditional probability that we are not fooled.

Now consider $M \to \infty$. Define $x := \frac{L-1}{M}$. The above summation approaches $x\int_x^1 \frac{1}{t}dt = -xlog_e x$ which is maximized at $x = \frac{1}{e}$. Hence, as number of secretaries increases, the optimal

strategy is to let $\frac{1}{e}$ fraction of them go by and then pick the first best one.

Summary
This problem indicates how to set up a problem as a DP problem. It illustrates that among all strategies optimal ones can be found among a small class of strategies (i.e. threshold type) and once you determine this class, it is relative easy to find an actual optimal strategy. This is typical of how dynamic programming is used in practice. Here the optimal strategy within the identified class of strategies was also found analytically, but in practice you may be able to use simulation and numerical techniques to find the best strategy within this class (after having identified which class of strategies to work with through analysis of the dynamic programming recursion).

We now turn to another example. The point is to illustrate the importance of correctly modeling a real world problem.

5.2 Asset Selling Problem

This problem is discussed in the textbook, section 4.4. The set-up is:
1. You have an asset that you would like to sell: e.g. a house with a Bay view
2. You have N offers, $w_0, \ldots, w_{N-1}$ one after another, modeled as i.i.d with a known distribution.
3. If you get an offer, you invest the cash at an interest rate r till the end of the process, at time N. If you reject an offer, it's gone once and for all.
Objective: maximize the expected reward at end of the process.
Note that this problem can be solved directly without using DP, but we will use a DP approach.

State Space
$x_0 = \{0\}$, a dummy state,
$x_1 = \{w_0\}$,
$x_k = \{w_{k-1}, T\} \quad k = 2 \ldots N.$

At time 0 you move from the dummy state to x_1. At each time $1 \leq k \leq N-1$, there are two control actions: either pick the current offer w_{k-1} and move to the terminal state or keep going. If you reach a nonterminal state at time N you are looking at the last offer $x_N = w_{N-1}$ and you *have to* accept it (this is not treated as a control action).

Note that in contrast to our discussion in the secretary problem, we are abusing notation by not carrying the notion of time in the terminal state. We will attribute the reward of terminating (including investment gain) at the time that we choose to accept an offer thereby making the movement between terminal states from one time to another have zero reward, so there is no point distinguishing between terminal states at different times.

DP Recursion

$J_k(T) = 0$ for all $1 \leq k \leq N$,

$J_k(x_k) = max\{(1+r)^{N-k}x_k, E\{J_{k+1}(w_k)\}\}$ for $0 \leq k \leq N-1$, where $x_k \neq T$. Here the maximization is taken over the two possible control actions. To understand this equation, note that for $1 \leq k \leq N-1$ the decision to accept the offer $x_k = w_{k-1}$ allows you to invest it for $N-k$ time steps; this reward is paid up front and you move to the terminal state whose reward-to-go is 0. The decision to reject the offer moves you to state w_k at time $k+1$, you get no immediate reward, and the expected reward-to-go is now $E[J_{k+1}(w_k)]$.

$J_N(x_N) = x_N$ for $x_N \neq T$. To understand this equation note that we assume that you *have to* accept the last offer if you have not yet accepted any offers, so we just treat the reward due to this (no investment gain since there is no time left to invest) as being a reward in the final state.

Observations

1. An optimal strategy is given by a moving threshold. The strategy is given for $1 \leq k \leq N-1$ by:

accept the offer x_k if $x_k > \alpha_k$

reject the offer x_k if $x_k < \alpha_k$,

where $\alpha_k = \frac{E\{J_{k+1}(w_k)\}}{(1+r)^{N-k}}$. In case $x_k = \alpha_k$ both decisions result in same reward.

Note that α_k is decreasing with k. This requires proof, and the proof is in the book, but the intuition is that as k increases, there is less chance to see an offer that becomes better. Hence, if the offer is good enough to be accepted at time k it should also be acceptable at time $k+1$.

2. Why did we bother to discuss this example in class? Let's compare this problem to the secretary problem. In many ways it refers to the same kind of situation (you have a problem of picking one of N options which are offered to you in sequence, and if you reject an offer you can never go back to it). However the nature of the optimal strategy in the asset selling problem (moving threshold) is very different from than in the secretary problem (allow a fraction roughly $\frac{1}{e}$ of the offers to go by and then pick the next best). This seems odd. The reason is that the model is different in the two cases. Contrary to the secretary problem, here we know the distribution of the offers, hence we have some absolute notion of how good they are. Moreover, there is reward associated with accepting each offer and not just the best offer.

3. The message is that the model is very important. Unless you model the problem well, you don't know what you are getting. As in all engineering: **Junk in $\Rightarrow$ Junk out**

5.3 Warehouse Restocking Problem

This problem is also in the book in section 4.2. The importance of it is that it illustrates another general, widely used methodology for deriving qualitative properties of optimal strategies in problems amenable to the DP approach.

The set-up is: You have a warehouse. At each time k you get a random demand w_k and you have to make a restocking order u_k. We assume that $u_k \geq 0$. Let x_k denote the amount of supplies in warehouse at time k. Then: $x_{k+1} = x_k + u_k - w_k \quad k = 0 \ldots N-1$. Here we allow x_k to be arbitrary real valued, with the convention that $x_k < 0$ denotes borrowing from somebody else.
The objective is to minimize the cost function: $E\{\sum_{k=0}^{N-1}(r(x_k) + cu_k) + R(x_N))\}$,
where $r(x_k)$ will be taken to be a piecewise linear function such that when $x_k > 0$ it comes from to a penalty per unit amount for keeping supplies in the warehouse and for $x_k < 0$ it comes from a penalty per unit amount of borrowing from someone else. $R(x_N)$ is similar. Thus, we consider $r(x_k) = pmax(0, -x_k) + hmax(0, x_k)$, i.e. piecewise linear cost, with slope h when we have positive supply and slope p when we have negative cost, and the same for $R(x_N)$. Further, c denotes the cost per unit amount of restocking.

DP Recursion
$J_N(N) = 0$.
$J_k(x_k) = min_{u \geq 0}\{E\{J_{k+1}(x_k + u - w_k) + cu_k + r(x_k + u - w_k)\}\}$,
where the minimization is taken over all possible controls at time k.

In this problem, we will show by induction that $J_k(x_k)$ is a nonnegative convex function which approaches ∞ as $\mid x_k \mid \rightarrow \pm\infty$. From this the optimal solution is derived. This property of $J_k(x_k)$ will be proved by backwards induction, starting with $J_{N-1}(x_{N-1})$. We will look at this example in more detail in the next lecture.

Observation
Often one can identify qualitative properties of the optimal cost-to-go functions, for example: convexity, monotonicity, multimodularity, etc., proving that these hold by backwards induction. Such properties can then indicate that the optimizing control strategies are in some class of strategies, for example: threshold strategies, time-varying threshold strategies, strategies based on some index rule, strategies based on some threshold function, etc., and hence one can determine optimal strategies for the problem at hand.