2 Some Examples of Dynamic Programming

We illustrate the method of dynamic programming and some useful 'tricks'.

2.1 Example: managing spending and savings

An investor receives annual income from a building society of x_t pounds in year t. He consumes u_t and adds $x_t - u_t$ to his capital, $0 \le u_t \le x_t$. The capital is invested at interest rate $\theta \times 100\%$, and so his income in year $t+1$ increases to

$$x_{t+1} = a(x_t, u_t) = x_t + \theta(x_t - u_t).$$

He desires to maximize his total consumption over h years, $\mathbf{C} = \sum_{t=0}^{h-1} u_t$.

Solution. In the notation we have been using, $c(x_t, u_t, t) = u_t$, $\mathbf{C}_h(x_h) = 0$. This is a **time-homogeneous** model, in which neither costs nor dynamics depend on t. It is easiest to work in terms of '**time to go**', $s = h - t$. Let $F_s(x)$ denote the maximal reward obtainable, starting in state x and when there is time s to go. The dynamic programming equation is

$$F_s(x) = \max_{0 \le u \le x} \left[u + F_{s-1}(x + \theta(x-u))\right],$$

where $F_0(x) = 0$, (since no more can be obtained once time h is reached.) Here, x and u are generic values for x_s and u_s.

We can substitute backwards and soon guess the form of the solution. First,

$$F_1(x) = \max_{0 \le u \le x} \left[u + F_0(u + \theta(x-u))\right] = \max_{0 \le u \le x} [u + 0] = x\,.$$

Next,

$$F_2(x) = \max_{0 \le u \le x} \left[u + F_1(x + \theta(x-u))\right] = \max_{0 \le u \le x} \left[u + x + \theta(x-u)\right].$$

Since $u + x + \theta(x-u)$ linear in u, its maximum occurs at $u = 0$ or $u = x$, and so

$$F_2(x) = \max[(1+\theta)x, 2x] = \max[1+\theta, 2]x = \rho_2 x\,.$$

This motivates the guess $F_{s-1}(x) = \rho_{s-1}x$. Trying this, we find

$$F_s(x) = \max_{0 \le u \le x} \left[u + \rho_{s-1}(x + \theta(x-u))\right] = \max[(1+\theta)\rho_{s-1}, 1 + \rho_{s-1}]x = \rho_s x\,.$$

Thus our guess is verified and $F_s(x) = \rho_s x$, where ρ_s obeys the recursion implicit in the above, and i.e., $\rho_s = \rho_{s-1} + \max[\theta\rho_{s-1}, 1]$. This gives

$$\rho_s = \begin{cases} s & s \le s^* \\ (1+\theta)^{s-s^*} s^* & s \ge s^* \end{cases},$$

where s^* is the least integer such that $s^* \ge 1/\theta$, i.e., $s^* = \lceil 1/\theta \rceil$. The optimal strategy is to invest the whole of the income in years $0, \dots, h - s^* - 1$, (to build up capital) and then consume the whole of the income in years $h - s^*, \dots, h-1$.

There are several things worth remembering from this example. (i) It is often useful to frame things in terms of time to go, s. (ii) Although the form of the dynamic programming equation can sometimes look messy, try working backwards from $F_0(x)$ (which is known). Often a pattern will emerge from which we can piece together a solution. (iii) When the dynamics are linear, the optimal control lies at an extreme point of the set of feasible controls. This form of policy, which either consumes nothing or consumes everything, is known as **bang-bang control**.

2.2 Example: exercising a stock option

The owner of a call option has the option to buy a share at fixed 'striking price' p. The option must be exercised by day h. If he exercises the option on day t and then immediately sells the share at the current price x_t, he can make a profit of $x_t - p$. Suppose the price sequence obeys the equation $x_{t+1} = x_t + \epsilon_t$, where the ϵ_t are i.i.d. random variables for which $E|\epsilon| < \infty$. The aim is to exercise the option optimally.

Let $F_s(x)$ be the value function (maximal expected profit) when the share price is x and there are s days to go. Show that (i) $F_s(x)$ is non-decreasing in s, (ii) $F_s(x) - x$ is non-increasing in x and (iii) $F_s(x)$ is continuous in x. Deduce that the optimal policy can be characterised as follows.

There exists a non-decreasing sequence $\{a_s\}$ such that an optimal policy is to exercise the option the first time that $x \ge a_s$, where x is the current price and s is the number of days to go before expiry of the option.

Solution. The state variable at time t is, strictly speaking, x_t plus a variable which indicates whether the option has been exercised or not. However, it is only the latter case which is of interest, so x is the effective state variable. Since dynamic programming makes its calculations backwards, from the termination point, it is often advantageous to write things in terms of the time to go, $s = h - t$. So if we let $F_s(x)$ be the value function (maximal expected profit) with s days to go then

$$F_0(x) = \max\{x - p, 0\},$$

and so the dynamic programming equation is

$$F_s(x) = \max\{x - p, E[F_{s-1}(x + \epsilon)]\}, \quad s = 1, 2, \dots$$

Note that the expectation operator comes *outside*, not inside, $F_{s-1}(\cdot)$.

One can use induction to show (i), (ii) and (iii). For example, (i) is obvious, since increasing s means we have more time over which to exercise the option. However, for a formal proof

$$F_1(x) = \max\{x - p, E[F_0(x + \epsilon)]\} \ge \max\{x - p, 0\} = F_0(x).$$

Now suppose, inductively, that $F_{s-1} \ge F_{s-2}$. Then

$$F_s(x) = \max\{x - p, E[F_{s-1}(x + \epsilon)]\} \ge \max\{x - p, E[F_{s-2}(x + \epsilon)]\} = F_{s-1}(x),$$

whence F_s is non-decreasing in s. Similarly, an inductive proof of (ii) follows from

$$\underbrace{F_s(x) - x} = \max\{-p, E[\underbrace{F_{s-1}(x+\epsilon) - (x+\epsilon)}] + E(\epsilon)\},$$

since the left hand underbraced term inherits the non-increasing character of the right hand underbraced term. Thus the optimal policy can be characterized as stated. For from (ii), (iii) and the fact that $F_s(x) \geq x - p$ it follows that there exists an a_s such that $F_s(x)$ is greater that $x - p$ if $x < a_s$ and equals $x - p$ if $x \geq a_s$. It follows from (i) that a_s is non-decreasing in s. The constant a_s is the smallest x for which $F_s(x) = x - p$.

2.3 Example: accepting the best offer

We are to interview h candidates for a job. At the end of each interview we must either hire or reject the candidate we have just seen, and may not change this decision later. Candidates are seen in random order and can be ranked against those seen previously. The aim is to maximize the probability of choosing the candidate of greatest rank.

Solution. Let W_t be the history of observations up to time t, i.e., after we have interviewed the tth candidate. All that matters are the value of t and whether the tth candidate is better than all her predecessors: let $x_t = 1$ if this is true and $x_t = 0$ if it is not. In the case $x_t = 1$, the probability she is the best of all h candidates is

$$P(\text{best of } h \mid \text{best of first } t) = \frac{P(\text{best of } h)}{P(\text{best of first } t)} = \frac{1/h}{1/t} = \frac{t}{h}.$$

Now the fact that the tth candidate is the best of the t candidates seen so far places no restriction on the relative ranks of the first $t-1$ candidates; thus $x_t = 1$ and W_{t-1} are statistically independent and we have

$$P(x_t = 1 \mid W_{t-1}) = \frac{P(W_{t-1} \mid x_t = 1)}{P(W_{t-1})} P(x_t = 1) = P(x_t = 1) = \frac{1}{t}.$$

Let $F(t-1)$ be the probability that under an optimal policy we select the best candidate, given that we have passed over the first $t-1$ candidates.Dynamic programming gives

$$F(t-1) = \frac{t-1}{t} F(t) + \frac{1}{t} \max\left(\frac{t}{h}, F(t)\right) = \max\left(\frac{t-1}{t} F(t) + \frac{1}{h}\,,\; F(0,t)\right)$$

The first term deals with what happens when the tth candidate is not the best so far; we should certainly pass over her. The second term deals with what happens when it is. In that case we have a choice: accept that candidate (which will turn out to be best with probability t/h, or pass over that candidate).

These imply $F(t-1) \geq F(t)$ for all $t \leq h$. Therefore, since t/h and $F(t)$ are respectively increasing and non-increasing in t, it must be that for small t we have $F(t) > t/h$ and for large t we have $F(t) \leq t/h$. Let t_0 be the smallest t such that $F(t) \leq t/h$. Then

$$F(t-1) = \begin{cases} F(t_0)\,, & t < t_0\,, \\ \dfrac{t-1}{t} F(t) + \dfrac{1}{h}\,, & t \geq t_0\,. \end{cases}$$

Solving the second of these backwards from the point $t = h$, $F(h) = 0$, we obtain

$$\frac{F(t-1)}{t-1} = \frac{1}{h(t-1)} + \frac{F(t)}{t} = \cdots = \frac{1}{h(t-1)} + \frac{1}{ht} + \cdots + \frac{1}{h(h-1)}\,,$$

whence

$$F(t-1) = \frac{t-1}{h} \sum_{\tau=t-1}^{h-1} \frac{1}{\tau}, \quad t \geq t_0\,.$$

Since we require $F(t_0) \leq t_0/h$, it must be that t_0 is the smallest integer satisfying

$$\sum_{\tau=t_0}^{h-1} \frac{1}{\tau} \leq 1\,.$$

For large h the sum on the left above is about $\log(h/t_0)$, so $\log(h/t_0) \approx 1$ and we find $t_0 \approx h/e$. The optimal policy is to interview $\approx h/e$ candidates, but without selecting any of these, and then select the first one thereafter that is the best of all those seen so far. The probability of success is $F(t_0) \sim t_0/h \sim 1/e = 0.3679$. It is surprising that the probability of success is so large for arbitrarily large h.

There are a couple lessons in this example. (i) It is often useful to try to establish the fact that terms over which a maximum is being taken are monotone in opposite directions, as we did with t/h and $F(t)$. (ii) A typical approach is to first determine the form of the solution, then find the optimal cost (reward) function by backward recursion from the terminal point, where its value is known.