TOPIC 3 DYNAMIC PROGRAMMING

TEXAS
The University of Texas at Austin

Dynamic Programming

- ALL dynamic programming problems need a few common elements
- State variables
  - What information do you need to describe where you are
- Choice/Decision variables
  - What can you choose to do
- Dynamics
  - How do choice variables combine with state variables to evolve through time
- Value Function
  - Discounted value of all future payoffs
- Bellman Equation
  - Value today is immediate payoff plus discounted payoff tomorrow
- Terminal/Boundary Condition
  - Value function after the last time step

Fishing Example

What are the state variables, choice variables, dynamics, value function, Bellman equation and terminal condition?

State variable

(s, t)

tons of fish in lake

time

dynamics

$$(s,t) \longrightarrow \begin{cases} (2s, t+1) & \text{if } nf \\ (s, t+1) & \text{if } f \end{cases}$$

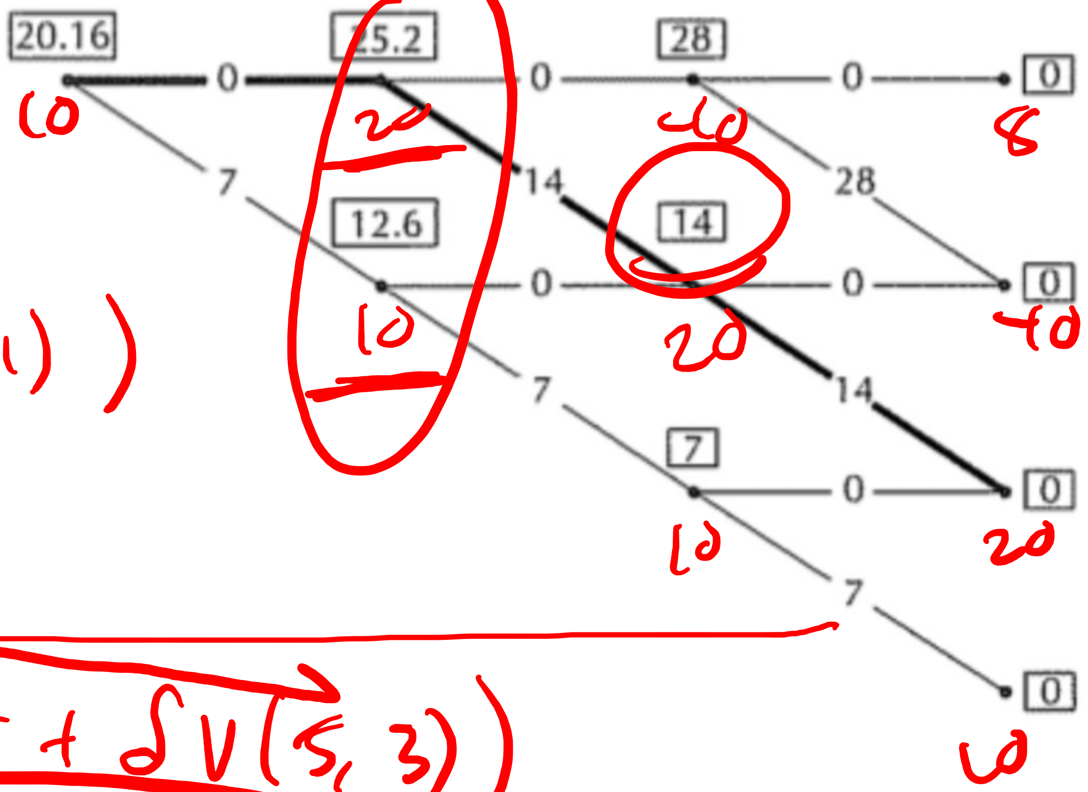
Choice

f or nf

Value function

$$V(s,t) = \sum_{i=0}^{T-t} (\text{rev} @ t+i)\, \delta^i$$

Bellman



$$V(S,t) = \max\left(\delta V(2S, t+1),\ 0.7S + \delta V(S, t+1)\right)$$

Terminal: $\boxed{V(S,3) = 0}$

$$V(S,2) = \max\left(\delta V(2S,3),\ 0.7S + \delta V(S,3)\right)$$

$$V(S,1) = \max\left(\delta V(2S,2),\ 0.7S + \delta V(S,2)\right)$$

$$V(S,0) = \max\left(\delta V(2S,1),\ 0.7S + \delta V(S,1)\right)$$

Fishing Example

- How would we put this in python to find the optimal policy?

V[:, 3] = 0

for t in 2, 1, 0
    for all possible s @ t
        V[s, t] = max (δ V(2s, t+1),
            0.75 + δ V(s, t+1))

Fishing Example

- Let's rearrange the triangle upside down into a box!

	0	1	2	3
0	10	10	10	10
1	0	20	20	20
2	0	0	40	40
3	0	0	0	80

Dynamic Programming

- Value Function
  - $v(S_t, t) = \max_x \sum_{i=0}^{T-t} \delta^i \, r_{t+i}$
  - $v(S_t, t) = \max_x r_t + \delta \sum_{i=0}^{T-(t+1)} \delta^i \, r_{(t+1)+i}$
  - $v(S_t, t) = \max_x r_t + \delta v(S_{t+1}, t+1)$
- If I know the value function for all possible values of s tomorrow, then I can calculate it for all possible values of s today!
- In general, tomorrow’s state is dependent on today’s state and our choice today

Mining Example

- You must decide how much ore to extract from a mine that will be shut down and abandoned after T years of operation.
- The sales price of extracted ore is p dollars per ton, and the total cost of extracting x tons of ore in any year, given that the mine contains s tons at the beginning of the year, is $x^2/(1+s)$ dollars.
- The mine currently contains M tons of ore
- This discount factor is δ
- Assuming the amount of ore extracted in any year must be an integer number of tons, what extraction schedule maximizes profits?

$x_0 + x_1 + x_2 + \cdots x_{T-1} \le M$

Mining Example – Class Participation

- Can we pose this as a traditional optimization problem (not a DP)?

$$\max_{x_0, x_1, \ldots, x_{T-1}} \; P x_0 - \frac{x_0^2}{1+M} + \delta\left(P x_1 - \frac{x_1^2}{1+M-x_0}\right) +$$

$$\delta^2\left(P x_2 - \frac{x_2^2}{1+M-x_0-x_1}\right) + \cdots + \delta^{T-1}\left(P x_{T-1} - \frac{x_{T-1}^2}{1+M-x_0-x_1-\cdots-x_{T-2}}\right)$$

$x \ge 0$ $\quad \sum_{t=0}^{T-1} x_i \le M$ $\quad \sum_{t=1}^{T-1} x_i \le M - x_0$

Mining Example - DP

What are the state variables, choice variables, dynamics, value function, Bellman equation and terminal condition?

State

(s, t)

ore in mine

time

dynamics

$(s, t) \rightarrow (s - x, t + 1)$

Choice

x_t = tons to extract @ t

value function

$V(s, t) = \max \sum_{i=0}^{T-t} (\text{Profit} @ t+i)\,\delta^i$

Bellman

$$V(s,t) = \max_{0 \le x \le s} \; px - \frac{x^2}{1+s} + \delta V(s-x, t+1)$$

Terminal : $V(s,T) = 0$

$$V(s, T-1) = \max_{0 \le x \le s} \; px - \frac{x^2}{1+s} + \delta V(s-x, T)$$

$$V(s, T-2) = \max_{0 \le x \le s} \; px - \frac{x^2}{1+s} + \delta V(s-x, T-1)$$

$$V(s, T-3) = \max_{0 \le x \le s} \; px - \frac{x^2}{1+s} + \delta V(s-x, T-2)$$

Dynamic Programming

- The general Bellman Eq is
  - $v(S_t, t) = \max_x r_t + \delta v(S_{t+1}, t+1)$
  - S_t is a dynamic variable that changes through time
- For the mining example this is
  - $v(s, t) = \max_{0 \le x \le s} px - \frac{x^2}{1+s} + \delta v(s - x, t+1)$
  - s is one particular value that variable could take on
  - If S_t takes on the value s, then S_{t+1} takes on the value $s - x$

Mining Example

- How would we code it in python?

$V[:, T] = 0$

for t in $T-1, T-2, \ldots, 0$

for s in $0, 1, \ldots, M$

$x =$ range$(s+1)$

possible $= px - \frac{x^2}{1+s} + \delta V(s-x, t+1)$

$V(s,t) =$ max(possible)

$U(s,t] =$ argmax(possible)