

Lecture 2: The Bellman Equation

Dynamic Programming

Thomas Jørgensen



The Simplest Consumption Model

- Model:**

$$V_t^*(M) = \max_{C_t, C_{t+1}, \dots, C_T} \left\{ C_t + \beta C_{t+1} + \dots + \beta^{T-t} C_T \right\}$$

s.t.

$$M = C_t + C_{t+1} + \dots + C_T$$

$$C_t \in \mathbb{N}$$

- Solution by backwards induction:**

$$V_T^*(M) = \sqrt{M}$$

$$V_{T-1}^*(M) = \max_{C_{T-1}} \left\{ \sqrt{C_{T-1}} + \beta V_T^*(M - C_{T-1}) \right\}$$

$$V_{T-2}^*(M) = \max_{C_{T-2}} \left\{ \sqrt{C_{T-2}} + \beta V_{T-1}^*(M - C_{T-2}) \right\}$$

$$V_{T-3}^*(M) = \max_{C_{T-3}} \left\{ \sqrt{C_{T-3}} + \beta V_{T-2}^*(M - C_{T-3}) \right\}$$

...

$$V_t^*(M) = \max_{C_t} \left\{ \sqrt{C_t} + \beta V_{t+1}^*(M - C_t) \right\}$$



Recap

Bellman's Principle
of Optimality

Stochastic shocks

Continuous choice

Wrap-Up

Until next

Algorithm 4: Find V_t^* given V_{t+1}^* [find_V]

input : M $V_{t+1}^*[\bullet]$ **output:** $V_t^*[\bullet]$ (value of optimal choice) $C_t^*[\bullet]$ (optimal choice)**1 for** $m = 0$ **to** M **do****2** $i_M = m + 1$ **3** $V_t^*[i_M] = -\infty$ **4** **for** $C_t = 0$ **to** m **do****5** $V = \sqrt{C_t} + \beta V_{t+1}^*[(m - C_t) + 1]$ **6** **if** $V > V_t^*[i_M]$ **then****7** $V_t^*[i_M] = V$ **8** $C_t^*[i_M] = C_t$



Algorithm 5: Find all V_t^* (algorithm 4 inserted)

```

input :  $M$ 
output:  $V_t^*[\bullet]$  for all  $t$ 
            $C_t^*[\bullet]$  for all  $t$ 

1   $V_T^*[m+1] = \sqrt{m}$  for all  $m = 0$  to  $M$ 
2  for  $t = T - 1$  to 1 do
3      for  $m = 0$  to  $M$  do
4           $i_M = m + 1$ 
5           $V_t^*[i_M] = -\infty$ 
6          for  $C_t = 0$  to  $m$  do
7               $V = \sqrt{C_t} + \beta V_{t+1}^*[(m - C_t) + 1]$ 
8              if  $V > V_t^*[i_M]$  then
9                   $V_t^*[i_M] = V$ 
10                  $C_t^*[i_M] = C_t$ 
  
```

[From last lecture: **b.socrative.com**, room:
DynamicProgramming (all but the last question)]



Slight reformulation I

- **Instead of:**

$$V_t^*(M) = \max_{C_t, C_{t+1}, \dots, C_T} \left\{ \sqrt{C_t} + \beta \sqrt{C_{t+1}} + \beta^2 \sqrt{C_{t+2}} + \dots + \beta^{T-t} \sqrt{C_T} \right\}$$

s.t.

$$M = C_t + C_{t+1} + \dots + C_T$$

$$C_t \in \mathbb{N}$$

- **We will write:**

$$V_t^*(M_t) = \max_{C_t, C_{t+1}, \dots, C_T} \left\{ \sqrt{C_t} + \beta \sqrt{C_{t+1}} + \beta^2 \sqrt{C_{t+2}} + \dots + \beta^{T-t} \sqrt{C_T} \right\}$$

s.t.

$$M_{t+1} = M_t - C_t$$

$$C_t \leq M_t$$

$$C_t \in \mathbb{N}$$

$$M_t = M$$



Slight reformulation II

- Start from

$$V_t^*(M_t) = \max_{C_t, C_{t+1}, \dots, C_T} \left\{ \sqrt{C_t} + \beta \sqrt{C_{t+1}} + \beta^2 \sqrt{C_{t+2}} + \dots + \beta^{T-t} \sqrt{C_T} \right\}$$

- As C_{t+1}, \dots, C_T does not affect C_t we can write

$$\begin{aligned} V_t^*(M_t) &= \max_{C_t} \left\{ \sqrt{C_t} + \beta \left\{ \max_{C_{t+1}, \dots, C_T} \sqrt{C_{t+1}} + \beta \sqrt{C_{t+2}} + \dots + \beta^{T-(t+1)} \sqrt{C_T} \right\} \right\} \\ &= \max_{C_t} \left\{ \sqrt{C_t} + \beta V_{t+1}^*(M_{t+1}) \right\} \end{aligned}$$

- What do I use in the second equality?**
- Mathematical details:** See Lucas and Stokey theorem 4.2.



Recap

Bellman's Principle
of Optimality

Stochastic shocks

Continuous choice

Wrap-Up

Until next

Bellman's Principle of Optimality

***Principle of Optimality:** An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision. (See Bellman, 1957, Chap. III.3.)*



Model with stochastic shocks

- Consider the following **Bellman equation**

$$V_t(M_t) = \max_{C_t} \sqrt{C_t} + \beta \mathbb{E}_t[V_{t+1}(M_{t+1})]$$

s.t.

$$C_t \leq M_t$$

$$M_{t+1} = \begin{cases} M_t - C_t + 1 & \text{with probability } \pi \in (0, 1) \\ M_t - C_t & \text{else} \end{cases}$$

where $\mathbb{E}_t[\bullet] \equiv \mathbb{E}[\bullet | M_t, C_t]$

- Question 1:** Can we try out all combinations of C_t, C_{t+1}, \dots, C_T ?
- Question 2:** Can we write down a formula for $\mathbb{E}_t[V_{t+1}(M_{t+1})]$?

Go to **b.socrative.com**, room name **DynamicProgramming**
[Q1 only]



Formula for $\mathbb{E}_t[V_{t+1}(M_{t+1})]$

- **We have:**

$$M_{t+1} = \begin{cases} M_t - C_t + 1 & \text{with probability } \pi \in (0, 1) \\ M_t - C_t & \text{else} \end{cases}$$

- **Possibility 1:**

$$\mathbb{E}_t[V_{t+1}(M_{t+1})] = \pi V_{t+1}(M_t - C_t + 1)$$

- **Possibility 2:**

$$\mathbb{E}_t[V_{t+1}(M_{t+1})] = V_{t+1}(\pi(M_t - C_t + 1) + (1 - \pi)(M_t - C_t))$$

- **Possibility 3:**

$$\mathbb{E}_t[V_{t+1}(M_{t+1})] = \pi V_{t+1}(M_t - C_t + 1) + (1 - \pi)V_{t+1}(M_t - C_t)$$



Algorithm 6: Find V_t^* given V_{t+1}^* [find_V]

input : M $V_{t+1}^*[\bullet]$ **output:** $V_t^*[\bullet]$ (value of optimal choice) $C_t^*[\bullet]$ (optimal choice)

```

1 for  $m = 0$  to  $M + t$  do
2    $i_M = m + 1$ 
3    $V_t^*[i_M] = -\infty$ 
4   for  $C_t = 0$  to  $m$  do
5      $i\_next = (m - C_t) + 1$ 
6      $V = \sqrt{C_t} + \beta V_{t+1}^*[i\_next]$  (if deterministic)
7      $i\_next\_plus = (m - C_t) + 1 + 1$ 
8      $EV_{t+1} = (1 - \pi)V_{t+1}^*[i\_next] + \pi V_{t+1}^*[i\_next\_plus]$ 
9      $V = \sqrt{C_t} + \beta EV_{t+1}$  (if stochastic)
10    if  $V > V_t^*[i_M]$  then
11       $V_t^*[i_M] = V$ 
12       $C_t^*[i_M] = C_t$ 

```



Model with continuous consumption

- Consider the following **Bellman equation**

$$V_t(M_t) = \max_{C_t} \sqrt{C_t} + \beta \mathbb{E}_t[V_{t+1}(M_{t+1})]$$

s.t.

$$C_t \leq M_t$$

$$M_{t+1} = M_t - C_t$$

where we now allow C_t to be continuous,
i.e. $C_t \in \mathbb{R}$ and not just $C_t \in \mathbb{N}$.

- Remember:** We only know $V_{t+1}(M_{t+1})$ at some *grid*,
 $M_{t+1} \in \{0, 1, 2, 3, \dots, M\} \equiv \mathcal{G}$
- Question:** What is $V_{t+1}(0.5)$?



Recap

Bellman's Principle
of Optimality

Stochastic shocks

Continuous choice

Wrap-Up

Until next

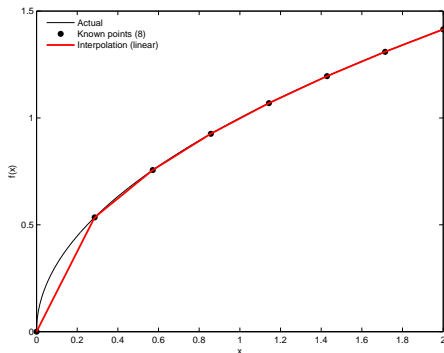
Linear Interpolation

- **Information:**

- ① $f(x)$ is known for x in some set \mathcal{G}
- ② \mathcal{G} is ordered *low to high*

- **Task:** linear interpolation at $x \notin \mathcal{G}$:

Figure: Linear interpolation



Linear Interpolation

- **Information:**

- ① $f(x)$ is known for x in some set \mathcal{G}

- ② \mathcal{G} is ordered *low to high*

- **Task:** linear interpolation at $x \notin \mathcal{G}$:

- **Step 1: Neighbors.** Find $x_n \in \mathcal{G}$ such that

$$x_n \leq x < x_{n+1}.$$

- **Step 2: Relative distance.** Compute

$$\omega = \frac{x - x_n}{x_{n+1} - x_n}$$

- **Step 3: Interpolation value.** Compute

$$\begin{aligned} f(x) &\approx (1 - \omega)f(x_n) + \omega f(x_{n+1}) \\ &= f(x_n) + \omega(f(x_{n+1}) - f(x_n)) \end{aligned}$$



Algorithm 7: Linear interpolation [called "interp" later]

input : x (point to be interpolated)
 \mathcal{G} (grid of length $\#$ where the function is known)
 $f[\bullet]$ (function values at grid points)
output: y (interpolated value)

```

1 if  $x < \mathcal{G}[1]$  then
2   |    $n = 1$  (extrapolate below)
3 else if  $x > \mathcal{G}[\#]$  then
4   |    $n = \# - 1$  (extrapolate above)
5 else
6   |    $n = 1$ 
7     while  $x ? \mathcal{G}[n + 1]$  do
8       |    $n = n + 1$ 
9    $\omega = \frac{x - x[n]}{x[n + 1] - x[n]}$ 
10   $y = f[n] + \omega(f[n + 1] - f[n])$ 
  
```

- What should be instead of “?” on line 7: \leq, \geq ?
 - \geq because when this requirement is violated for the first time, x is just above $\mathcal{G}[n + 1]$ and below $\mathcal{G}[n + 2]$



Algorithm 8: Find V_t^* given V_{t+1}^* [find_V]

input : \mathcal{G}_M (grid for M with $\#_M$ elements) $V_{t+1}^*[\bullet]$ **output:** $V_t^*[\bullet]$ (value of optimal choice) $C_t^*[\bullet]$ (optimal choice)1 **for** $i_M = 1$ **to** $\#_M$ **do**2 $V_t^*[i_M] = -\infty$ 3 $M_t = \mathcal{G}_M[i_M]$ 4 **for** $C_t = 0$ **to** M_t **do**5 $V = \sqrt{C_t} + \beta \cdot \text{interp}(M_t - C_t, \mathcal{G}_M, V_{t+1}^*)$ 6 **if** $V > V_t^*[i_M]$ **then**7 $V_t^*[i_M] = V$ 8 $C_t^*[i_M] = C_t$

-
- Or is something not quite right?



Summary

- **The Bellman equation** was introduced. In general notation

$$V_t(M_t) = \max_{C_t \in \mathcal{C}(M_t)} u(C_t) + \beta \mathbb{E}_t[V_{t+1}(\Gamma(M_t, C_t))]$$

- You can now solve models with:
 - ① Stochastic shocks
 - ② Continuous choices and/or states
- You can also do both at once...



Algorithm 9: Find V_t^* given V_{t+1}^* [find_V]

input : $V_{t+1}^*[\bullet]$
 \mathcal{G}_M (grid for M with $\#_M$ elements)
 \mathcal{G}_C (grid for C (as a share of M) with $\#_C$ elements in $(0, 1)$)
output: $V_t^*[\bullet]$ (value of optimal choice)
 $C_t^*[\bullet]$ (optimal choice)

```

1  for  $i_M = 1$  to  $\#_M$  do
2       $V_t^*[i_M] = -\infty$ 
3       $M_t = \mathcal{G}_M[i_M]$ 
4      for  $i_c = 1$  to  $\#_C$  do
5           $C_t = \mathcal{G}_C[i_c]M_t$ 
6           $EV_{t+1} = \pi \text{interp}(M_t - C_t + 1, \mathcal{G}_M, V_{t+1}^*)$ 
               $+ (1 - \pi) \text{interp}(M_t - C_t, \mathcal{G}_M, V_{t+1}^*)$ 
7           $V = \sqrt{C_t} + \beta EV_{t+1}$ 
8          if  $V > V_t^*[i_M]$  then
9               $V_t^*[i_M] = V$ 
10              $C_t^*[i_M] = C_t$ 

```

- Could you do something different?



Until next

- **Ensure that you understand:**

- ① Algorithm 7-9
- ② And why we had

$$\mathbb{E}_t[V_{t+1}(M_{t+1})] = \pi V_{t+1}(M_t - C_t + 1) + (1 - \pi) V_{t+1}(M_t - C_t)$$

- Go to **PadLet** and ask or answer a question
(<https://padlet.com/tjo3/DP>)

- **Think about:**

- ① How can we compute $\mathbb{E}_t[V_{t+1}(M_{t+1})]$ if:

$$M_{t+1} = R(M_t - C_t) + Y_{t+1}$$

$$Y_{t+1} = \exp(\xi_{t+1})$$

$$\xi_{t+1} \sim \mathcal{N}(0, \sigma_\xi^2)$$

- **Now:** go to **b.socratic.com** room name
“**DynamicProgramming**” and answer a few questions

