Dynamic Programming Exercise Class 2

Jacob Røpke

February 2025

Ex. 3: Cake-eating problem: Continuous choice

Last time:

$$V_t(W_t) = \max_{c_t \in \{0,1,\ldots,W_t\}} \sqrt{c_t} + \beta V_{t+1}(W_t + 1)$$

$$\text{s.t}$$

$$W_{t+1} = W_t - c_t$$

$$c_t \in \{0, 1, \ldots, W_t\}$$

Today:

$$V_t(W_t) = \max_{c_t \in \{0,1,\ldots,W_t\}} \sqrt{c_t} + \beta V_{t+1}(W_t + 1)$$

$$\text{s.t}$$

$$W_{t+1} = W_t - c_t$$

$$c_t \in \mathbb{R}_+$$

$$W_t - c_t \geq 0$$

Solving on a grid

- In Dynamic Programming we typically solve problems on a grid:
  - Grid of whole cake-slices: $\{0,1,2,3,4,5\}$
  - Grid of half slices of cake: $\{0,0.5,1.0,1.5. \ldots ...,5.0\}$
- We solve the problem at each grid point.
- With continuous choice models we will often end up having to evaluate the value function between grid points $\Longrightarrow$ We need to interpolate!
  - Linear interpolation: Drawing lines between points.

Ex. 4: Cake-eating problem: with discrete stochastic shocks

$$V_t(W_t) = \max_{c_t \in \{0,1,\ldots,W_t\}} \sqrt{c_t} + \beta E_t[V_{t+1}(W_t + 1)]$$

$$\text{s.t}$$

$$W_{t+1} = W_t - c_t + \varepsilon_t$$

$$\varepsilon_i \in \mathbb{N} \quad \text{Following some probability distribution}$$

$$c_t \in \mathbb{R}_+$$

$$W_t - c_t \geq 0$$

There is now an expectation in the bellman equation:
$E_t[V_{t+1}(W_{t+1})] = \sum_{\varepsilon_i \in \mathbb{N}} [V_{t+1}(W_t - c_t + \epsilon_i)]$

Ex. 5: Expectations over continuous shocks (probability densities)

- Ex. 4: Discrete shocks.
- Can be easily handled using a sum on a computer
- What if we had continuous shocks in the cake-eating problem?
  - $E_t[V_{t+1}(W_{t+1})] = \int V_{t+1}(W_t - c_t + \varepsilon)g(\varepsilon)d\varepsilon$
  - Everything on a computer must be discretized: $\Longrightarrow$ Integral must be turned into a sum.
- Ex. 5 Two main approaches.
  - Monte Carlo: Approximate expectations as sample average $E_t[V_{t+1}(W_{t+1})] \approx \frac{1}{M}\sum_{i=1}^{M} V_{t+1}(W_t - c_t + \varepsilon_i)$
  - Gaussian quadrature: Similar but pick shocks in a smart way.