PhD Macro Core Part I:
Lecture 8 – Dynamic Programming IV

Min Fang
University of Florida

Fall 2024

Today

- Practical stochastic dynamic programming
- Numerical integration to help compute expectations
- Using interpolations & approximations to solve the stochastic optimal growth model

Goal: Solve the Stochastic Growth Model

- Let's solve the Bellman equation

$$v(k,z) = \max_{k'} \{u\left(f(k,z) - k'\right) + \beta \mathbb{E}\left[v\left(k',z'\right) \mid z\right]\}$$

with the usual specification

$$f(k,z) = zk^{\alpha} + (1-\delta)k$$

$$u(c) = \frac{c^{1-\sigma} - 1}{1-\sigma}$$

- And let's suppose that z' is an $\mathrm{AR}(1)$ in logs ($\mu = 0$)

$$\log z' = \rho_z \log z + \varepsilon, \quad \varepsilon \sim \mathrm{IID}\, N\left(0, \sigma_z^2\right)$$

Steps: Value Function Iteration

- Step 1: Create grids for both states and transition matrix for z

$$k_{\min} < \ldots < k_i < \ldots < k_{\max}, \quad i = 1, \ldots, n$$

$$z_{\min} < \ldots < z_j < \ldots < z_{\max}, \quad j = 1, \ldots, m; \quad \text{an } m \times m \text{ matrix } P_z$$

- Step 2: Create $n \times m$ matrices for value function $v(k, z)$ and policy function $k'(k, z)$

- Step 3: Make an initial guess of value function $v_0(k, z)$

- Step 4: For each pair today (z_j, k'), calculate the expected value $\mathbb{E}\left[v\left(k', z'\right) \mid z_j\right]$

- Step 5: Given $\mathbb{E}\left[v\left(k', z'\right) \mid z_j\right]$ for any k', find $k_1'(k_i, z_j)$ for each pair (k_i, z_j) and $v_0(k, z)$

- Step 6: Calculate $v_1(k, z) \Rightarrow$ Now we are done with one iteration! Repeat until convergence!

How to Decide the Grids of z: Tauchen (1986)

- z' is an AR(1) in logs ($\mu = 0$)

$$\log z' = \rho_z \log z + \varepsilon, \quad \varepsilon \sim \text{IID}\, N\left(0, \sigma_z^2\right)$$

- Conditional on z, $\log z'$ is normally distributed with mean $\rho_z \log z$ and standard deviation σ_z

- Decisions to make: number of points m, grid range ($\log z_m - \log z_1$), probability matrix P_z

$$\log z_1 < \ldots < \log z_j < \ldots < \log z_m, \quad j = 1, \ldots, m; \quad \text{an } m \times m \text{ matrix } P_z$$

- Choose $\log z_1 = -\tau\sigma_z / \sqrt{1 - \rho_z^2}$ and $\log z_m = +\tau\sigma_z / \sqrt{1 - \rho_z^2}$ where τ is a number

- We can then denote distance $w \equiv \log z_2 - \log z_1 = \log z_j - \log z_{j-1}$

How to Decide the Grids of z: Tauchen (1986)

- For any j between 2 and $m-1$:

$$p_{i,j} = \text{Prob}\left[\log z_j - w/2 \leqslant \rho_z \log z_i + \varepsilon \leqslant \log z_j + w/2\right]$$

- If $j = 1$ then set

$$p_{i,1} = \text{Prob}\left[\rho_z \log z_i + \varepsilon \leqslant \log z_1 + w/2\right]$$

- And if $j = m$ set

$$p_{i,m} = 1 - \text{Prob}\left[\log z_m - w/2 \leqslant \rho_z \log z_i + \varepsilon\right]$$

- Equivalently, (Φ is the standard normal c.d.f.)

$$p_{ij} = \begin{cases} \Phi\left(\frac{\log z_1 + w/2 - \rho_z \log z}{\sigma_z}\right) & \text{for } j = 1 \\ \Phi\left(\frac{\log z_m - w/2 - \rho_z \log z}{\sigma_z}\right) - \Phi\left(\frac{\log z_1 + w/2 - \rho_z \log z}{\sigma_z}\right) & \text{for } 1 < j < n \\ 1 - \Phi\left(\frac{\log z_m - w/2 - \rho_z \log z}{\sigma_z}\right) & \text{for } j = n \end{cases}$$

How to Decide the Grids of z: Tauchen (1986)

- For any (i, j) pair, we have now:

$$p_{ij} = \text{Prob}\left(z' = logz_j \mid z = logz_i\right)$$

- Now, computing the conditional expectation no longer involves an integral since

$$\mathbb{E}\left[v\left(k', z'\right) \mid z_i\right] = \sum_{j=1}^{m} p_{ij} v\left(k', z'\right)$$

- The most difficult step 5 is nailed!

VFI with Discrete Choices of k'

- Almost the same as Lecture 6 without stochastic component z

- But everything is two dimensional now: value function and policy function

- Just more effort of the computer!

VFI with Discrete Choices of k': S1-S4

- Step 1: Create grids for both states and transition matrix for z

$$k_{\min} < \ldots < k_i < \ldots < k_{\max}, \quad i = 1, \ldots, n$$

$$z_{\min} < \ldots < z_j < \ldots < z_{\max}, \quad j = 1, \ldots, m; \quad \text{an } m \times m \text{ matrix } P_z$$

- Step 2: Create $n \times m$ matrices for value function $v(k, z)$ and policy function $k'(k, z)$

- Step 3: Make an initial guess of value function $v_0(k, z) = 0$

- Step 4: For each pair today (z_i, k'), calculate the expected value

$$\mathbb{E}\left[v_0\left(k', z'\right) \mid z_i\right] = \sum_{j=1}^{m} p_{ij} v_0\left(k', z'\right)$$

VFI with Discrete Choices of k': S5

- Let c_{ijp} denote consumption if current productivity is $z = z_i$, capital is $k = k_j$ and capital chosen for next period is $k' = k_p$
$$c_{ijp} = z_i f\left(k_j\right) - k_p, \quad i, j = 1, \ldots, n$$
We will need to be careful to respect the feasibility constraints
$$0 \leqslant k_p \leqslant f\left(k_j\right), \quad i, j = 1, \ldots, n$$
- Let u_{ijp} denote the flow utility associated with c_{ijp}
$$u_{ijp} = u\left(c_{ijp}\right), \quad i = 1, \ldots, m; j, p = 1, \ldots, n$$
- So u is an $m \times n \times n$ matrix

VFI with Discrete Choices of k': S5

- In this notation, our Bellman equation can be written

$$v_{ij} = \max_p \left[u_{ijp} + \beta \mathbb{E}[v_{i'p}]\right]$$

- Associated with this is the policy function

$$k'_{ij} = g_{ij} = \operatorname*{argmax}_p \left[u_{ijp} + \beta \mathbb{E}[v_{i'p}]\right]$$

VFI with Discrete Choices of k': S6

- Start with an initial guess v_{ij}^0 and then calculate

$$v_{ij}^1 = Tv_{ij}^0 = \max_p \left[u_{ijp} + \beta \mathbb{E}[v_{i'j}^0]\right]$$

and compute the error

$$\left\|Tv^0 - v^0\right\| = \max_i \left[\left|Tv_i^0 - v_i^0\right|\right]$$

- If this error is less than some pre-specified tolerance $\varepsilon > 0$, stop. Otherwise, update to

$$v_{ij}^2 = Tv_{ij}^1$$

until the error is small enough

VFI with Discrete Choices of k': Performance

- To achieve a high-accuracy solution, discrete choices of k' need a huge grid on k
- Say, $k_{min} = k_1 = 0.01$ and $k_{max} = k_{10000} = 100.00$
- If your productivity grid points are 10
- Then u_{ijp} is a $10 \times 10000 \times 10000$ matrix with 10^9 elements!
- Evaluation of the below step will be extremely costly

$$v_{ij}^1 = Tv_{ij}^0 = \max_p \left[u_{ijp} + \beta \mathbb{E}[v_{i'j}^0] \right]$$

- Can we improve on it? Certainly, we can!

VFI with Continuous Choices of k': Setup

- The reason that we need so many grid points (nodes) of k is because of accuracy need
- If true $k'(k_j, z_i) = 0.521$, but the nearest nodes are $k_- = 0.515$ and $k_+ = 0.530$
- Then, we would choose $\hat{k}'(k_j, z_i) = k_- = 0.515$, $error = \hat{k}'/k' - 1 = 1.16\%!$ (huge)
- To reduce the error to $10e^{-}7$, we need 15,000 nodes in between $k_- = 0.515$ and $k_+ = 0.530$
- So, ideally, we want to choose true $k'(k_j, z_i) = 0.521$, but we do not know $v(0.521, z_i)$, any i
- To achieve this, we need interpolation

VFI with Continuous Choices of k': Interpolation

- Interpolation: a type of estimation, a method of constructing (finding) new data points based on the range of a discrete set of known data points.

- We will do linear interpolation for today (simple)

- Given z_i, $v(k_-, z_i)$ and $v(k_+, z_i)$ are known, now we want to know $v(k', z_i)$ for any $k' \in [k_-, k_+]$

- We construct an interpolation function

$$Interp(v(k, z_i)) = v(k_j, z_i) + (v(k_{j+1}, z_i) - v(k_j, z_i))\frac{k' - k_j}{k_{j+1} - k_j}$$

for any k_j from k_{min} to k_{max-1}

VFI with Continuous Choices of k': Only Change is in S5

- In this notation, our Bellman equation can be written

$$v_{ij} = \max_{p} \left[u_{ijp} + \beta Interp(\mathbb{E}[v_{i'p}])\right]$$

- Associated with this is the policy function

$$k'_{ij} = g_{ij} = \operatorname*{argmax}_{p} \left[u_{ijp} + \beta Interp(\mathbb{E}[v_{i'p}])\right]$$

- In practice, instead of choosing $k_{min} = k_1 = 0.01$ and $k_{max} = k_{10000} = 100.00$

- You can now choose $k_{min} = k_1 = 0.01$ and $k_{max} = k_{30} = 100.00$ (30 instead of 10,000 nodes)

VFI in Stochastic Growth: Practice

- Next homework: Solve the same model with different methods
- Compute speed, error, etc