

ECON 210: Section 3

Marcelo Sena

Plan for today

- ▶ Code optimization in the change example
- ▶ Infinite-horizon consumption-savings
- ▶ Infinite horizon value function iteration algorithm

Infinite-Horizon Consumption-Savings

We are interested in maximizing:

$$\lim_{T \rightarrow \infty} \sum_{t=0}^T \beta^t u(c_t) \quad (1)$$

$$\text{s.t.} \quad \begin{cases} c_t + R^{-1}a_t = a_{t-1} \\ c_t \geq 0, a_t \geq 0, \forall t \end{cases} \quad (2)$$

The objective function is typically written more compactly as

$$\sum_{t=0}^{\infty} \beta^t u(c_t) \quad (3)$$

Infinite-Horizon Consumption-Savings

The recursive formulation now reads;

$$V(a) = \max_{c, a'} u(c) + \beta V(a') \quad (4)$$

$$\text{s.t. } \begin{cases} c + R^{-1}a' = a \\ c, a' \geq 0 \end{cases} \quad (5)$$

- ▶ no time now $\implies$ one less state variable!
- ▶ “the world tomorrow looks the same as today”
- ▶ problem is parallelizable in each state (not the case with time backward iteration)
- ▶ a is the **state**
- ▶ (c, a') are determined by **policy functions**
- ▶ value function iteration algorithm: “discretization” of the Bellman operator

Infinite-Horizon Consumption-Savings

Value function iteration algorithm (justified on the grounds of the Contraction-Mapping, Theorem 4.6 SLP)

- i) Initialize the assets grid a and $V^{(0)}(a) = 0$ (what could be another more “clever” choice?)
- ii) For every a_i in a we will compute optimal policy:
 - ii) a. For every a_j in a , compute:

$$T_{ij} = u(a_i - a_j/R) + \beta V(a_j) \quad (6)$$

- ii) b. If $c = a_i - a_j/R < 0$, set T_{ij} to a large negative value. This imposes the fact that the agent cannot pick negative consumption.
 - ii) c. Set $V^{(1)}(a_i) = \max_j T_{ij}$.
- iii) Compute then the distance between the two value functions, the guess and the updated value function:

$$d = \|V^{(1)} - V^{(0)}\| \quad (7)$$

Typically we use the max norm, so $\|v\| = \max_{i \in \mathcal{V}} |v_i|$.

- iv) If the distance is smaller than some pre-determined tolerance threshold ϵ , then we claim convergence.

Infinite-Horizon Consumption-Savings

Points worth noting:

- ▶ as a sanity check, we can see if the VFI algorithm converges to the finite-horizon solution when we set T very large, thus approximating $T \rightarrow \infty$.
- ▶ the tolerance level will generally depend on the units of the problem at hand and the utility function, so it is useful to check these when setting the tolerance. The final solution must be insensitive to marginal changes in the tolerance level
- ▶ similarly, it is useful to print the distance between value functions during the algorithm (monotonic convergence is a sign things are going well, but may not occur for more complicated models or algorithmic variations)
- ▶ as always, need to be careful with the bounds (should be large enough) and size (should be large enough but not too expensive computationally) of the a grid.