LECTURES 7

Solving models with occasionally binding constraints
Part I

Macroeconomics 4

A.Y. 2014-15

Income fluctuations again

- Consider the standard income fluc. problem in recursive form, when $T \to \infty$:

$$V(a,s) = \max_{\{c,a'\}} u(c) + \beta \mathbb{E}\left[V(a',s') \mid s\right],$$
$$\text{s.t. } a' = (1+r)\,a + ws - c,$$
$$a' \geq 0.$$

  for given $r > 0$ and $w > 0$. Assume the usual regularity conditions on u, and that $\beta(1+r) < 1$.

- Suppose that s is an idiosyncratic shock to efficiency of labor that follows an AR process of the form:

$$\ln(s') = (1-\rho)\ln(\mu_s) + \rho \ln(s) + \epsilon,$$

  where $\epsilon \sim N\left(0, \sigma_\epsilon^2\right)$.

- Given the AR nature of the process, the "cash in hand" trick does not apply, so we are left with two state variables: asset holdings and exogenous labor productivity.

Income fluctuations again

- Note that s' is log-normally distributed:

$$s' = \exp\left[(1-\rho)\ln(\mu_s) + \rho\ln(s) + \epsilon\right].$$

- Hence:

$$\mathbb{E}(s' \mid s) = \exp\left[(1-\rho)\ln(\mu_s) + \rho\ln(s) + \sigma_\varepsilon^2/2\right],$$
$$\operatorname{var}(s' \mid s) = \left[\exp\left(\sigma_\epsilon^2\right) - 1\right]\mathbb{E}(s' \mid s)^2.$$

- Note furthermore that $\mathbb{E}[\ln(s)] = \ln(\mu_s)$, so that $\mathbb{E}(s) > \mu_s$ because of *Jensen's inequality*.

Income fluctuations again

- Let us now discretize the $AR(1)$ process using one of the methods previously described.

- Hence, the problem can be rewritten in the following way:

$$V\left(a, s = \mathrm{s}_i\right) = \max_{\{c, a'\}} u\left(c\right) + \beta \sum_{j=1}^{n} \Pi_{ij} V\left(a', s' = \mathrm{s}_j\right),$$

$$\text{s.t. } a' = \left(1 + r\right) a + w\mathrm{s}_i - c,$$

$$a' \geq 0.$$

- This however suggests that instead of a bivariate value function, we could equivalently use a set of univariate ones:

$$V_i\left(a\right) = \max_{\{c, a'\}} u\left(c\right) + \beta \sum_{j=1}^{n} \Pi_{ij} V_j\left(a'\right),$$

$$\text{s.t. } a' = \left(1 + r\right) a + w\mathrm{s}_i - c,$$

$$a' \geq 0.$$

Constrained asset holdings

- Let us completely discretize the state space, and constrain asset holdings on this finite-dimensional grid:

$$\mathcal{A} = \{0 < \mathrm{a}_1 < \mathrm{a}_2 < ... < \mathrm{a}_m\}.$$

- The problem boils down to:

$$V_i(\mathrm{a}_z) = \max_{a' \in \mathcal{A}} \left\{ u\left[(1+r)\mathrm{a}_z + w\mathrm{s}_i - a'\right] + \beta \sum_{j=1}^{n} \Pi_{ij} V_j(a') \right\}.$$

- Being the obj. function concave and the constraint set convex, there is one and only one optimal a' for each current state $(\mathrm{a}_z, \mathrm{s}_i)$, i.e. the policy function is a *deterministic* single-value function.

- This implies that we can define a single-valued *indicator function* such that:

$$\mathcal{I}(a_j, a_z, s_i) = \begin{cases} 1 & \text{if } g(\mathrm{a}_z, \mathrm{s}_i) = \mathrm{a}_j \\ 0 & \text{if } g(\mathrm{a}_z, \mathrm{s}_i) \neq \mathrm{a}_j \end{cases}.$$

Constrained asset holdings

- The operator implied by the r.h.s. of our problem turns out to be a *contraction*.
- Hence, taking advantage of **Banach's theorem**, we iterate until convergence on the following scheme, given an initial guess for $V_{0,i}$:

$$V_{k+1,i}\left(\mathrm{a}_z\right) = \max_{a' \in \mathcal{A}} \left\{ u\left[\left(1+r\right)\mathrm{a}_z + w\mathrm{s}_i - a'\right] + \beta \sum_{j=1}^{n} \Pi_{ij} V_{k,j}\left(a'\right) \right\}.$$

- Convergence is achieved when $\|V_{k+1,i} - V_{k,i}\| \leq \varepsilon > 0$ for all i.
- This solution method is known as *Value Function Iteration* (*VFI*).

Constrained asset holdings

- Define a set of $m \times 1$ vectors $\mathbf{v}_i$ and $m \times m$ matrices $\mathbf{R}_i$, with $i = 1, 2, ..., n$, s.t.:

$$\begin{aligned} \mathbf{v}_i(z) &= V_i(\mathrm{a}_z), \\ \mathbf{R}_i(z, j) &= u\left[(1+r)\,\mathrm{a}_z + w\mathrm{s}_i - \mathrm{a}_j\right], \end{aligned}$$

  for all $z, j = 1, 2, ...m$.

- Furthermore, define:

$$\underset{(nm)\times 1}{\mathbf{v}} \equiv \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_n \end{bmatrix}, \quad \underset{(nm)\times m}{\mathbf{R}} \equiv \begin{bmatrix} \mathbf{R}_1 \\ \vdots \\ \mathbf{R}_n \end{bmatrix}.$$

Constrained asset holdings

- Thus, *VFI* can be represented in matrix notation as:

$$\mathbf{v}_{k+1} = \max \left[\mathbf{R} + \beta \left(\Pi \otimes \mathbf{1}_m \right) \begin{bmatrix} \mathbf{v}_{k,1}^T \\ \vdots \\ \mathbf{v}_{k,n}^T \end{bmatrix} \right].$$

- The pol. function, and the indic. function $\mathcal{I}\left(a_z, a_j, s_i\right)$, can be represented by a set of $m \times m$ matrices $\mathbf{G}_i$, with $i = 1, 2, ..., n$, s.t.:

$$\mathbf{G}_i\left(z, j\right) = \begin{cases} 1 & \text{if } g\left(\mathrm{a}_z, \mathrm{s}_i\right) = \mathrm{a}_j \\ 0 & \text{if } g\left(\mathrm{a}_z, \mathrm{s}_i\right) \neq \mathrm{a}_j \end{cases}.$$

- This method is slow and not particularly accurate for reasonable grid sizes, but at least it converges for sure.

Explicit optimization

- A more (numerically) elegant approach would be the following.
- As before, discretize the state space by specifying a grid for current asset holdings, $\mathcal{A} = \{0 < \mathrm{a}_1 < \mathrm{a}_2 < ... < \mathrm{a}_m\}$.
- Approximate the value function, i.e. the functions $V_i\left(a'\right)$, via interpolation (possibly using *shape-preserving cubic splines*).
- Given initial guesses $V_{0,i}$, iterate on the following scheme:

$$V_{k+1,i}\left(\mathrm{a}_z\right) = \max_{a'} u\left[\left(1+r\right)\mathrm{a}_z + w\mathrm{s}_i - a'\right] + \beta \sum_{j=1}^{n} \Pi_{ij} V_{k,j}\left(a'\right),$$

$$\text{s.t. } a' \geq 0.$$

where the optimization on the r.h.s. is performed numerically via some robust algorithm that takes inequalities into account.

Explicit optimization

- The policy function is obtained as a by-product, and, again, can be approx. via interpolation:

$$a'_{k,i}\left(\mathrm{a}_z\right) = \arg\max_{a'} u\left[\left(1+r\right)\mathrm{a}_z + w\mathrm{s}_i - a'\right] + \beta \sum_{j=1}^{n} \Pi_{ij} V_{k,j}\left(a'\right),$$

$$\text{s.t. } a' \geq 0.$$

- The key difference with the previous approach is the following: we do **NOT** impose $a' \in \mathcal{A}$; hence, we should make sure that $a'_i\left(\mathrm{a}_m\right) \leq \mathrm{a}_m$ for all i, for both theoretical and numerical reasons.

- This approach is slow, potentially accurate, and should converge (of course, IF the problem has a solution, which is not obvious, as you know by now).

Explicit optimization

- Computations can be accelerated by anticipating some equilibrium properties of the solution.

- In particular, being a' zero or strictly increasing in "cash in hand":

$$V_{k+1,i}\left(\mathrm{a}_z\right) = \max_{a'} u\left[\left(1+r\right)\mathrm{a}_z + w\mathrm{s}_i - a'\right] + \beta \sum_{j=1}^{n} \Pi_{ij} V_{k,j}\left(a'\right),$$

$$\text{s.t. } a' \in \left[a'_i\left(\mathrm{a}_{z-1}\right), \mathrm{a}_m\right].$$

- This helps the optimization routine to find a local solution.

References I