Projection

Wouter J. Den Haan
London School of Economics

June 3, 2011

Model

$$
\begin{aligned}
c_t^{-\nu} &= \mathsf{E}_t\left[\beta c_{t+1}^{-\nu}\alpha z_{t+1}k_{t+1}^{\alpha-1}\right] \\
c_t + k_{t+1} &= z_t k_t^{\alpha} \\
\ln(z_{t+1}) &= \rho \ln(z_t) + \varepsilon_{t+1} \\
& \quad \varepsilon_{t+1} \sim N(0,\sigma^2) \\
& \quad k_1, z_1 \text{ given}
\end{aligned}
$$

k_t is beginning-of-period t capital stock

Projection Methods

True rational expectations solution:

$$c_t = c(k_t, z_t)$$

- Why a difficult problem to find $c(k_t, z_t)$?

Approximation:

- structural parameters $(\alpha, \beta, \rho, \sigma)$ have fixed numerical values (thus not included as arguments in function)

$$c_t = c(k_t, z_t) \approx P_n(k_t, z_t; \eta_n)$$

- $P_n(\cdot)$: from class of approximating functions
  - such as polynomials or splines
- n is fixed $\Longrightarrow$ solve for η_n, a *finite-dimensional* object

Which equations to use?

- goal: solve for $P_n(k_t, z_t; \eta_n) \approx c(k_t, z_t)$,
  - i.e., N_n elements of η_n
  - $k(k_t, z_t)$ implicitly defined by budget constraint
- One first-order equation left, namely Euler equation
  - this is a different equation at each point in the state space
  - $\Longrightarrow$ plenty of equations

Which equations to use?

- Equations at M grid points $\{k_i, z_i\}$ with $M \geq N_n$

$$P_n(k_i, z_i; \eta_n)^{-\nu} =$$

$$\mathsf{E}\begin{bmatrix} \alpha\beta\times \\ P_n(\{\boldsymbol{k}'\}, \{\mathbf{z}'\}; \eta_n)^{-\nu}\times \\ \{\boldsymbol{z}'\}\times \\ (\{\boldsymbol{k}'\})^{\alpha-1} \end{bmatrix}$$

Which equations to use?

- **Goal:** $\forall$ grid point get 1 equation with η_n as only unknown

$$P_n(k_i, z_i; \eta_n)^{-\nu} = E\begin{bmatrix} \alpha\beta\times \\ P_n(\boldsymbol{k}', \mathbf{z}'; \eta_n)^{-\nu}\times \\ \boldsymbol{z}'\times \\ (\boldsymbol{k}')^{\alpha-1} \end{bmatrix}$$

- Note that k_i and z_i are known

Which equations to use?

$$P_n(k_i, z_i; \eta_n)^{-\nu} =$$

$$\mathsf{E}\left[\begin{array}{c} \alpha\beta\times \\ P_n(z_i k_i^\alpha - P_n(k_i, z_i; \eta_n), \exp\{\rho\ln(z_i) + \varepsilon'\}; \eta_n)^{-\nu}\times \\ \exp\{\rho\ln(z_i) + \varepsilon'\}\times \\ \left(z_i k_i^\alpha - P_n(k_i, z_i; \eta_n)\right)^{\alpha-1} \end{array}\right]$$

How to deal with expectations operator?

Let $\{\omega_j, \zeta_j\}_{j=1}^{J}$ be the Hermite Gaussian quadrature nodes

$$P_n(k_i, z_i; \eta_n)^{-\nu} =$$

$$\sum_{j=1}^{J} \left[\begin{array}{c} \alpha\beta \times \\ P_n(z_i k_i^{\alpha} - P_n(k_i, z_i; \eta_n), \exp\{\rho \ln(z_i) + \sqrt{2}\sigma\zeta_j\}; \eta_n)^{-\nu} \times \\ \exp\{\rho \ln(z_i) + \sqrt{2}\sigma\zeta_j\} \times \\ \left(z_i k_i^{\alpha} - P_n(k_i, z_i; \eta_n)\right)^{\alpha - 1} \\ \omega_j / \sqrt{\pi} \end{array} \right]$$

Define error terms

$$e(k_i, z_i; \eta_n) = P_n(k_i, z_i; \eta_n)^{-\nu} -$$

$$\sum_{j=1}^{J} \left[\begin{array}{c} \alpha\beta\times \\ \\ P_n(z_i k_i^{\alpha} - P_n(k_i, z_i; \eta_n), \exp\{\rho \ln(z_i) + \sqrt{2}\sigma\zeta_j\}; \eta_n)^{-\nu} \times \\ \\ \exp\{\rho \ln(z_i) + \sqrt{2}\sigma\zeta_j\} \times \\ \\ \left(z_i k_i^{\alpha} - P_n(k_i, z_i; \eta_n)\right)^{\alpha-1} \\ \\ \omega_j / \sqrt{\pi} \end{array} \right]$$

How to find coefficients of approximation?

- True rational expect. solution gives zero error term $\forall(k_i, z_i)$
- Thus, choose η_n such that error terms are as small as possible.

- **Collacation** $(M = N_n)$: Use equation solver to get errors exactly equal to zero on grid
- **Galerkin** $(M > N_n)$: Use minimization routine (and possibly smart weighting of error terms)

Different types of approximating functions

- $P_n(k_i, z_i; \eta_n)$ could be polynomial or spline
- dimension η_n usually higher for splines
  - may make eq. solver/minimization less appropriate
  - use iteration scheme instead

How to find coefficients of approximation?

1. Equation solver or minimization routine
2. Iteration procedures
   1. fixed-point iteration
   2. time iteration

Iterating versus eq. solver/minimization

- Advantage:
  - less of a black box
  - can deal with many coefficients
    - e.g. when spline is used
  - some iteration schemes are guaranteed to converge
    - under some regularity conditions
- Disadvantage:
  - does not use information on how best to update

Iteration procedure: Construct Grid

- Construct a grid with nodes for k and z
- At the nodes construct the basis functions of $P_n(k,z;\eta_n)$.
- For example, if

$$P_n(k,z;\eta_n) = \eta_{0,n} + \eta_{k,n}k + \eta_{z,n}z + \eta_{kk}k^2 + \eta_{kz}kz + \eta_{zz}z^2$$

then construct the matrix (where subscripts denote grid numbers)

$$X = \begin{bmatrix} 1 & k_1 & z_1 & k_1^2 & k_1 z_1 & z_1^2 \\ 1 & k_2 & z_2 & k_2^2 & k_2 z_2 & z_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & k_M & z_M & k_M^2 & k_M z_M & z_M^2 \end{bmatrix}$$

and calculate $(X'X)^{-1}X'$

Iteration procedure: Construct Grid

- **Chebyshev nodes:** Using Chebyshev nodes is important. This ensures uniform convergence. With equidistant nodes it is possible that the oscillations between grid point explode as the order of the polynomial increases.
- **Chebyshev polynomials**: If you have (i) no problems finding initial conditions and (ii) only low-order appoximations so that calculating the inverse of $X'X$ can be done accurately, then you can use regular polynomials. Orthogonal Chebyshev polynomials can overcome these problems. They ensure that $X'X$ is diagonal (and trivial to invert). This does require scaling of the state variables so they are between -1 and 1.

Fixed-point Iteration

The value of η_n used in the q^{th} iteration is referred to as η_n^q. Follow the following iteration scheme until convergence

- At each grid point:
  - Calculate the RHS of the Euler equation using the latest value for η_n, i.e., η_n^{q-1}
  - Use RHS to calculate c_i, value for c at i^{th} grid point
- Use values for c_i to obtain an estimate for η_n, $\hat{\eta}_n^q$
  - Polynomial: run a regression to get $\hat{\eta}_n^q$
  - Spline: the values of c at the nodes are the new values of η_n
- Let $\eta_n^q = \lambda \hat{\eta}_n^q + (1 - \lambda)\eta_n^{q-1}$

Fixed-point Iteration

- **Step 1: Calculate current consumption values implied by η_n^{j-1} at each grid point**
  - Use η_n^{q-1} to calculate $k' = z_i k_i^\alpha - P_n(k_i, z_i; \eta_n^{q-1})$
  - Use η_n^{q-1} to calculate $c' = P_n(k', z'; \eta_n^{q-1})$
  - Then, get c_i from

$$(c_i)^{-\nu} =$$

$$\sum_{j=1}^{J} \left[\begin{array}{c} \alpha\beta\times \\ P_n(z_i k_i^{\alpha} - P_n(k_i, z_i; \eta_n^{q-1}), \exp\{\rho \ln(z_i) + \sqrt{2}\sigma\zeta_j\}; \eta_n^{q-1})^{-\nu} \times \\ \exp\{\rho \ln(z_i) + \sqrt{2}\sigma\zeta_j\} \times \\ \left(z_i k_i^{\alpha} - P_n(k_i, z_i; \eta_n^{q-1}) \right)^{\alpha - 1} \\ \omega_j / \sqrt{\pi} \end{array} \right]$$

Fixed-point iteration

Step 2: Get new estimate for η_n by running a projection step

- Let $Y = [c_1, c_2, \cdots, c_M]'$
- If

$$P_n(k, z; \eta_n) = \eta_{0,n} + \eta_{k,n}k + \eta_{z,n}z + \eta_{kk}k^2 + \eta_{kz}kz + \eta_{zz}z^2$$

then

$$\hat{\eta}_n^q = \left(X'X\right)^{-1} X'Y$$

Fixed-point iteration

Step 2: Get new estimate for η_n by running a projection step

- If

  $$P_n(k, z; \eta_n) = \exp\left(\eta_{0,n} + \eta_{k,n}k + \eta_{z,n}z + \eta_{kk}k^2 + \eta_{kz}kz + \eta_{zz}z^2\right)$$

  then

  $$\widehat{\eta}_n^q = \left(X'X\right)^{-1} X' \ln(Y)$$

- no stochastic error term $\Longrightarrow$ ok to take ln of LHS & RHS

Fixed-point iteration

Step 3: Update η_n

$$\eta_n^q = \lambda \widehat{\eta}_n^q + (1-\lambda)\eta_n^{q-1} \text{ for } 0 < \lambda \leq 1$$

- Fixed-point iteration does not always converge
    - Choosing a lower value of λ:
        - convergence more likely
        - slows down algorithm if lower value not needed for convergence
- Alternative is **time iteration**

Time Iteration

- At each grid point use η_n^{q-1} only for *next period's* choices
- Again solve for c_i at each grid point
    - this is now a bit trickier (non-linear problem)
- Get n_n^q as with fixed-point iteration
    - guaranteed to converge without dampening (under regularity conditions)

Time Iteration - solving for c

Solve c_i from following non-linear equation

$$(\boldsymbol{c_i})^{-\nu} = \sum_{j=1}^{J} \left[\begin{array}{c} \alpha\beta\times \\ P_n(z_i k_i^\alpha - \boldsymbol{c_i}, \exp\{\rho\ln(z_i) + \sqrt{2}\sigma\zeta_j\}; \eta_n^{q-1})^{-\nu}\times \\ \exp\{\rho\ln(z_i) + \sqrt{2}\sigma\zeta_j\}\times \\ \left(z_i k_i^\alpha - \boldsymbol{c_i})\right)^{\alpha-1} \\ \omega_j/\sqrt{\pi} \end{array} \right]$$

Time Iteration

- Natural interpretation for η_n^{q-1} and η_n^q, namely
  - η_n^{q-1} is tomorrow's policy function and
  - η_n^q is today's policy function

- Time iteration is reliable and convergent
  - (the proof is related to the convergence of value function iteration, which uses the same idea)

Fixed-point versus time iteration

- Fixed-point iteration uses η_n^{q-1} for *all* terms on the RHS, i.e., both next period's consumption choice and today's capital choice
- Time iteration uses η_n^{q-1} only to evaluate next period's consumption
- The structure of time iteration mimics the choice of value function iteration:
    - next period's behavior described by previous solution for value function
    - Bellman equation used to solve for choice of c and k *simultaneously*

Endogenous grid points

- Simple idea: construct grid for k' instead of a grid for k
- Instead of solving for the choice k' given k, we now solve for the value of k that would have led to the choice k'
- In both cases you end up at each grid point with a set of values for k and a set of corresponding values for k'.
- Terminology is a bit confusing: the grid itself is exogenous and fixed but it is for an endogenous variable
- You can use endogenous grid points both with fixed-point and with time iteration
- The added value with time iteration lies in getting rid of the non-linear problem of solving for today's choices

Endogenous grid points and time iteration

- Time iteration $\Longrightarrow$
  - use η_n^{q-1} for tomorrow's choices and
  - use η_n^q only for today's choices (which show up on both sides of the policy function
- Then, get c_i from

Endogenous grid points and time iteration

$$(c_i)^{-\nu} = \sum_{j=1}^{J} \begin{bmatrix} \alpha\beta\times \\ P_n(k_i', \exp\{\rho\ln(z_i)+\sqrt{2}\sigma\zeta_j\}; \eta_n^{q-1})^{-\nu}\times \\ \exp\{\rho\ln(z_i)+\sqrt{2}\sigma\zeta_j\}\times \\ (k_i')^{\alpha-1} \\ \omega_j/\sqrt{\pi} \end{bmatrix}$$

and k_i from

$$k_i' + c_i = z_i k^{\alpha}$$

Perturbation versus projection

- Nondifferentiabilities
  - impossible for perturbation
- Large number of state variables
  - difficult for projection
- Constructing the grid can be difficult
  - apriori hard to know what sensible points are
  - some calculations may not be well defined everywhere

Perturbation versus projection

- Global versus local
  - Projection designed to be global method
  - Perturbation designed to be local method
    - but could give accurate global approximation
    - question is whether (lower-order) derivatives at perturbation point capture global behavior

References

- Judd, K. L., 1998, Numerical Methods in Economics, The MIT Press.
- Rendahl, P., 2006, Inequality constraints in recursive economies.
  - shows that time-iteration converges even in the presence of inequality constraints