LECTURE I

Harvard Econ 2416
Professor Gabriel Chodorow-Reich
Spring 2021

OUTLINE

Outline

Big picture

- Structural models have parameters. Data should inform the numerical values of those parameters.
- This lecture about ways to identify parameters.
- I start with solution methods.
- Then cover ML, SMM, IRF-matching, Bayesian.
- Each of these sub-topics could be a full lecture.
- The handbook chapter by Fernández-Villaverde,Rubio-Ramírez, and Schorfheide is a comprehensive treatment.
- Other useful resources:
  https://www.sas.upenn.edu/~jesusfv/teaching.html
  http://www.wouterdenhaan.com/notes.htm
- Throughout, I use neoclassical growth model to make concepts concrete.

Identification

- Given a sample Y and a parameter vector θ, identification means $p(Y|\theta) = p(Y|\theta') \Rightarrow \theta = \theta'$, i.e. the data identify a unique parameter vector.
- In econometrics, parameters are identified or not.
- Applied researchers sometimes slip into discussion of "identifying assumption" or "identified moments".
- An identifying assumption means that under the assumption, the identified parameter vector is equal to the true value.
- An identified moment can be used to identify the true θ from the data.
- Today's lecture about identification in formal sense.

Outline

Model

- Problem:

$$\max_{\{c_{t+h},k_{t+1+h}\}_{h=0}^{\infty}} E_t \sum_{h=0}^{\infty} \beta^h \frac{c_{t+h}^{1-\gamma}-1}{1-\gamma}$$

$$\text{s.t.} \qquad c_{t+h}+k_{t+1+h} = z_{t+h}k_{t+h}^{\alpha}+(1-\delta)\,k_{t+h},$$

$$\ln z_t = \rho \ln z_{t-1} + \sigma\varepsilon_t,\ \ \varepsilon_t \sim N(0,1),$$

$$k_t \quad \text{given.}$$

- From first order conditions:

$$\text{Euler:} \qquad c_t^{-\gamma} = E_t\left[\beta\, c_{t+1}^{-\gamma}\left(\alpha z_{t+1} k_{t+1}^{\alpha-1}+1-\delta\right)\right],$$

$$\text{Resource constraint:} \quad c_t + k_{t+1} = z_t k_t^{\alpha} + (1-\delta)\,k_t.$$

- Solution is policy function $c_t = c(k_t, z_t)$ and transition equation $k_{t+1} = k(k_t, z_t)$.

Special case

- For $\gamma = 1$, $\delta = 1$ (full depreciation):

$$c_t = (1 - \alpha\beta) z_t k_t^{\alpha},$$
$$k_{t+1} = \alpha\beta z_t k_t^{\alpha}.$$

- Verify:

Euler: $$\frac{1}{(1-\alpha\beta) z_t k_t^{\alpha}} = E_t \left[\beta \frac{\alpha z_{t+1} k_{t+1}^{\alpha-1}}{(1-\alpha\beta) z_{t+1} k_{t+1}^{\alpha}} \right],$$

Rewrite: $$\frac{1}{z_t k_t^{\alpha}} = \frac{\alpha\beta}{(\alpha\beta z_t k_t^{\alpha})},$$

Resource: $$(1-\alpha\beta) z_t k_t^{\alpha} + \alpha\beta z_t k_t^{\alpha} = z_t k_t^{\alpha}.$$

- Exact solution because no state variable with full depreciation.
- Otherwise, need to approximate.

Outline

Overview

- Much later, we will use a Kalman filter.
- Covering it now rather than break flow later.
- Kalman filter uses state-space representation.
- Recent advances use more flexible particle filters that accommodate non-linear state spaces and non-normal shocks (e.g. Herbst and Schorfheide, JoE 2019), but same basic idea.
- Given observed data, filter recovers unobserved shocks (e.g. productivity) and true, latent values of variables observed with measurement error.
- We will use it to compute likelihood given observed data.

State-space representation

- Let $Y_t = \{y_1, y_2, \ldots, y_t\}$ be a sequence of observed data.
- Observation and state equation:

$$\begin{aligned} \text{Observation:} \quad & y_t = Hs_t + w_t, \\ \text{State:} \quad & s_t = Fs_{t-1} + v_t, \\ \text{where:} \quad & \begin{bmatrix} w_t \\ v_t \end{bmatrix} \sim iidN\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} R & 0 \\ 0 & Q \end{bmatrix} \right). \end{aligned}$$

- y_t, s_t may be vector-valued. s_t may contain latent or hidden states.
- Assume H, F, R, Q known (or have candidate draw in estimation loop). These encode decision rules and parameter values.
- Notation:

$$\begin{aligned} x_{t|t-k} &= E\left[x_t | Y_{t-k}\right], \\ P_{t|t-k} &= Var\left[s_t | Y_{t-k}\right]. \end{aligned}$$

State space representation of growth model

- Assume model solved for linear policy rules
  $c_t = c(k_t, z_t) = a_{ck} k_t + a_{cz} z_t, k_{t+1} = k(k_t, z_t) = a_{kk} k_t + a_{kz} z_t$.

- State equation: $\underbrace{\begin{bmatrix} k_t \\ z_t \end{bmatrix}}_{s_t} = \underbrace{\begin{bmatrix} a_{kk} & a_{kz} \\ 0 & \rho \end{bmatrix}}_{F} \underbrace{\begin{bmatrix} k_{t-1} \\ z_{t-1} \end{bmatrix}}_{s_{t-1}} + \underbrace{\begin{bmatrix} 0 \\ \varepsilon_t \end{bmatrix}}_{v_t}$.

- Observation equation with all variables observed:
  $$\underbrace{\begin{bmatrix} c_t \\ k_t \\ z_t \end{bmatrix}}_{y_t} = \underbrace{\begin{bmatrix} a_{ck} & a_{cz} \\ 1 & 0 \\ 0 & 1 \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} k_t \\ z_t \end{bmatrix}}_{s_t} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}}_{w_t}$$

- Observation equation with only capital and measurement error $u_{k,t}$:
  $$\underbrace{\begin{bmatrix} k_t^* \end{bmatrix}}_{y_t} = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} k_t \\ z_t \end{bmatrix}}_{s_t} + \underbrace{\begin{bmatrix} u_{k,t} \end{bmatrix}}_{w_t}.$$

Updating

- Least squares projection formula: if $\begin{bmatrix} a \\ b \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_a \\ \mu_b \end{bmatrix}, \begin{bmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{bmatrix}\right)$, then $a|b \sim N\left(\mu_a + \Sigma_{ab}\Sigma_{bb}^{-1}(b - \mu_b), \Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba}\right)$.

Updating

- Least squares projection formula: if $\begin{bmatrix} a \\ b \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_a \\ \mu_b \end{bmatrix}, \begin{bmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{bmatrix}\right)$, then $a|b \sim N\left(\mu_a + \Sigma_{ab}\Sigma_{bb}^{-1}(b - \mu_b), \Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba}\right)$.
- Normality:

$$\begin{bmatrix} s_t \\ y_t \end{bmatrix} | Y_{t-1} \sim N\left(\begin{bmatrix} s_{t|t-1} \\ y_{t|t-1} \end{bmatrix}, \begin{bmatrix} P_{t|t-1} & P_{t|t-1}H' \\ HP_{t|t-1} & HP_{t|t-1}H' + R \end{bmatrix}\right).$$

Updating

- Least squares projection formula: if $\begin{bmatrix} a \\ b \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_a \\ \mu_b \end{bmatrix}, \begin{bmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{bmatrix}\right)$, then $a|b \sim N\left(\mu_a + \Sigma_{ab}\Sigma_{bb}^{-1}(b-\mu_b), \Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba}\right)$.
- Normality:

$$\begin{bmatrix} s_t \\ y_t \end{bmatrix} | Y_{t-1} \sim N\left(\begin{bmatrix} s_{t|t-1} \\ y_{t|t-1} \end{bmatrix}, \begin{bmatrix} P_{t|t-1} & P_{t|t-1}H' \\ HP_{t|t-1} & HP_{t|t-1}H' + R \end{bmatrix}\right).$$

- Apply projection formula: $s_t | Y_{t-1}, y_t = s_t | Y_t \sim N\left(s_{t|t}, P_{t|t}\right)$, where:

$$s_{t|t} = s_{t|t-1} + K_t\left(y_t - y_{t|t-1}\right),$$
$$P_{t|t} = P_{t|t-1} - K_t HP_{t|t-1},$$
$$K_t = P_{t|t-1}H'\left[HP_{t|t-1}H' + R\right]^{-1} \text{ is Kalman gain.}$$

Updating

- Least squares projection formula: if $\begin{bmatrix} a \\ b \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_a \\ \mu_b \end{bmatrix}, \begin{bmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{bmatrix}\right)$, then $a|b \sim N\left(\mu_a + \Sigma_{ab}\Sigma_{bb}^{-1}(b-\mu_b), \Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba}\right)$.
- Normality:

$$\begin{bmatrix} s_t \\ y_t \end{bmatrix} | Y_{t-1} \sim N\left(\begin{bmatrix} s_{t|t-1} \\ y_{t|t-1} \end{bmatrix}, \begin{bmatrix} P_{t|t-1} & P_{t|t-1}H' \\ HP_{t|t-1} & HP_{t|t-1}H' + R \end{bmatrix}\right).$$

- Apply projection formula: $s_t|Y_{t-1}, y_t = s_t|Y_t \sim N\left(s_{t|t}, P_{t|t}\right)$, where:

$$s_{t|t} = s_{t|t-1} + K_t\left(y_t - y_{t|t-1}\right),$$
$$P_{t|t} = P_{t|t-1} - K_t HP_{t|t-1},$$
$$K_t = P_{t|t-1}H'\left[HP_{t|t-1}H' + R\right]^{-1} \text{ is Kalman gain.}$$

- From observation and state equations:

$$s_{t|t-1} = Fs_{t-1|t-1},$$
$$P_{t|t-1} = FP_{t-1|t-1}F' + Q,$$
$$y_{t|t-1} = Hs_{t|t-1}.$$

Kalman filter

1. Start with initial values for $s_{0,0}, P_{0,0}$. For example, model steady state.
2. Obtain $s_{1|0}, P_{1,0}, y_{1,0}$ using last three equations from previous slide.
3. Use $s_{1|0}, P_{1,0}, y_{1,0}$ and previous equations for $s_{t|t}, P_{t|t}$ to obtain $s_{1|1}, P_{1|1}$.
4. Repeat from step 2.

Outline

Solution methods

- Model estimation requires you to first solve the model.
- Have to solve model fast, so estimation can loop over solutions.
- Active area of research; I am going to give you high-level overview.
- Two main approaches: local (perturbation) and global (projection).

Outline

Overview

- You know how to log-linearize a model to solve it.
- Perturbation provides a formalization of the linearization approach and easily extends to higher orders.
- Typically extremely fast.
- Provides local solution.
- Is a local solution sufficient? Perturbation can be quite accurate even for large shocks, especially with higher order solutions. But less accurate if model features important non-linearity (zero lower bound, occasionally binding constraints, etc.).

Implementation in neoclassical growth model

- Introduce perturbation parameter λ:
$$\ln z_t = \rho \ln z_{t-1} + \lambda \sigma \varepsilon_t, \ \varepsilon_t \sim N(0,1).$$
  - $\lambda = 0$: deterministic steady-state with $z = 1$.
  - $\lambda = 1$: original model.
- Index decision rules by λ:
$$c_t = c(k_t, z_t; \lambda),$$
$$k_{t+1} = k(k_t, z_t; \lambda).$$
- Steady state for $k_t = k, z_t = 1, \lambda = 0$:
$$c(k,1;0) = k^\alpha - \delta k,$$
$$k(k,1;0) = k.$$
- Next build local approximation around $(k,1;0)$.

Perturbation and Taylor approximation

- Additional notation:

$$\mathscr{H}\left(c_t, c_{t+1}, k_t, k_{t+1}, z_t; \lambda\right) = E_t \begin{pmatrix} c_t^{-\gamma} - \left[\beta c_{t+1}^{-\gamma}\left(\alpha z_{t+1} k_{t+1}^{\alpha-1} + 1 - \delta\right)\right] \\ c_t + k_{t+1} - z_t k_t^{\alpha} - (1-\delta) k_t \end{pmatrix}.$$

- Note that:

$$\begin{aligned} & \mathscr{H}\left(c_t, c_{t+1}, k_t, k_{t+1}, z_t; \lambda\right) \\ = & \mathscr{H}\left(c\left(k_t, z_t; \lambda\right), c\left(k\left(k_t, z_t; \lambda\right), \rho z_t + \lambda \sigma \varepsilon_{t+1}; \lambda\right), k_t, k\left(k_t, z_t; \lambda\right), z_t; \lambda\right) \\ \equiv & F\left(k_t, z_t; \lambda\right) = 0. \end{aligned}$$

- Since $F\left(k_t, z_t; \lambda\right) = 0$ for any values of its arguments, derivatives of $F(k, 1; 0)$ also must be zero. Using the chain rule:

$$0 = F_k(k, 1; 0) = \mathscr{H}_1 c_k + \mathscr{H}_2 c_k k_k + \mathscr{H}_3 + \mathscr{H}_4 k_k,$$

$$0 = F_z(k, 1; 0) = \mathscr{H}_1 c_z + \mathscr{H}_2 c_k k_z + \mathscr{H}_4 k_z + \mathscr{H}_5.$$

- Four equations ($\mathscr{H}$ is 2×1) in four unknowns: c_k, c_z, k_k, k_z.
- Solve with standard methods (Blanchard–Kahn; Uhlig; Sims; Klein).

First-order solution

- Coefficients c_k, c_z, k_k, k_z characterize linearized policy rules around $(k, 1)$ steady-state.

- First-order perturbation is formally equivalent to linearization.

- Sometimes implement with change-of-variables, for example by replacing c_t with $\ln c_t$.

- Certainty equivalence: $0 = F_\lambda(k, 1; 0) = \mathscr{H}_1 c_\lambda + \mathscr{H}_2 (c_k k_\lambda + c_\lambda) + \mathscr{H}_4 k_\lambda + \mathscr{H}_6$ is linear and homogenous system in c_λ, k_λ since $\mathscr{H}_6 = 0$. Therefore: $c_\lambda = k_\lambda = 0$.

Higher-order solutions

- Take second-order derivatives of $F(k_t, z_t; \lambda)$ around $k, 1, 0$ and set equal to zero.
- Substitute coefficients we already know from first-order solution.
- Result: 12 equations in 12 unknowns.
- Can continue iteration to arbitrary degree.
- Recursive $\Rightarrow$ speed advantage.

Pruning

- At degree d, drop recursive terms of degree higher than d.

- Second-order example ($\rho = 0$):

$$\begin{aligned} \text{Decision rule:} \quad k(k_t) &= a_1 k_t + a_2 k_t^2 + b_1 \varepsilon_t \\ \text{Subst. backward:} \quad &= a_1 k_t + a_2 \left(a_1 k_{t-1} + a_2 k_{t-1}^2 + b_1 \varepsilon_{t-1}\right)^2 + b_1 \varepsilon_t . \end{aligned}$$

- Second-order perturbation contains terms in k_{t-1}^3, k_{t-1}^4.

- Suppose at some date t a large realization of ε_t generates a value of k_{t+1} far from the steady-state value. Going forward, this value will be raised to cubic and higher-order powers and trigger an explosive path.

- Intuition: higher order more accurate locally, but not necessarily globally.

Outline

Overview

- Global methods look for policy rules that satisfy equilibrium conditions at many points in the state space (nodes).
- Can accommodate highly non-linear models.
- A lot of art (and some science) to choosing nodes efficiently.
- Underlying procedure is Stone-Weierstrass theorem: any continuous function can be approximated by a polynomial, i.e. if $f(x)$ is continous on $[a,b]$, there exists $h(x) = \sum_{m=1}^{n} \xi_m x^m$ such that $\forall \varepsilon > 0 \; |f(x) - h(x)| < \varepsilon$.
- Find approximating polynomial that minimizes errors at nodes.
- I will illustrate using a particular method known as *parameterized expectations* and then return to more general case.

Parameterized expectations (PE)

- Let $g(k_t, z_t) = E_t\left[\beta c_{t+1}^{-\gamma}\left(\alpha z_{t+1} k_{t+1}^{\alpha-1} + 1 - \delta\right)\right]$.

- Note: as before, can substitute policy rules to show conditional expectation must be function only of state variables.

- Goal: approximate $g(k_t, z_t)$ with a polynomial.

- Why $g(k_t, z_t)$? Expectations typically smoother than realizations.

- Notation: $P_n(k_t, z_t; \xi_n)$ is the nth order polynomial in k_t, z_t with coefficients indexed by ξ:
$$g(k_t, z_t) \approx P_n(k_t, z_t; \xi_n).$$

- Note: as before, may want to define in terms of logs of variables.

Stochastic PE algorithm

1. Fix the polynomial order n, an initial capital stock k_0, and a randomly drawn productivity process.
2. Guess (randomly?) values for the polynomial coefficients ξ^1, where the superscript refers to the number of the iteration.
3. Simulate the economy for T periods using the policy rule $c_t = P_n(k_t, z_t; \xi_n)^{-1/\gamma}$ and $k_{t+1} = (1-\delta)k_t + z_t k_t^\alpha - c_t$ to sequentially update simulated values.
4. Define $y_{t+1} = \beta c_{t+1}^{-\gamma}\left(\alpha z_{t+1} k_{t+1}^{\alpha-1} + 1 - \delta\right)$ as the realized value.
5. Regress y_{t+1} on an n-order polynomial in k_t, z_t and define ξ^2 as the coefficients from this regression.
6. Return to step 3 and iterate until convergence.

Practical aside: burn periods and add stochastic step in (5).

Non-stochastic PE

- Perturbation approximates decision rules with polynomial optimized around one point in k, z space.
- Stochastic PE approximates decision rules with polynomial optimized for simulated values of k_t, z_t.
- Non-stochastic PE approximates decision rules with polynomial optimized for user-defined grid points of k, z.
- Calculate for node j and iteration i:

$$\begin{aligned}
\text{Approximation:} \quad & c_j^i = P_n\left(k_j, z_j; \xi_n^i\right)^{-1/\gamma}, \\
\text{Resource constraint:} \quad & k_j^{i\prime} = (1-\delta)k_j + z_j k_j^{\alpha} - c_j^i, \\
\text{Expectation:} \quad & y_j^i = E\left[P_n\left(k_j^{i\prime}, z_j'; \xi_n^i\right)\left(\alpha z_j'\left(k_j^{i\prime}\right)^{\alpha-1} + 1 - \delta\right)\right],
\end{aligned}$$

  where x_j' denotes next period's value.
- Stochastic element in y_j^i is z_j'. Compute $E[]$ using Monte Carlo (brute force), numeric integration (e.g. Gaussian Hermite quadrature using $z_j' | z_j \sim N()$), or solve PDE.

Non-stochastic PE algorithm

1. Fix the polynomial order n and a set of grid points $\{k_j, z_j\}$.
2. Guess (randomly?) values for the polynomial coefficient ξ^1.
3. For each grid point, calculate y_j^1.
4. Regress y_j^1 on an n-order polynomial in k_j, z_j and define ξ^2 as the coefficients from this regression.
5. Return to step 3 and iterate until convergence.

CURSE OF DIMENSIONALITY

- Neoclassical growth model has two state variables: k, z.
- Suppose discretize each variable into M values.
- Tensor grid has M^2 nodes.
- For N state variables, grid has M^N nodes.
- Computationally burdensome because evaluate criterion at each node.

General projections

- Non-stochastic PE is example of projection method.
- Generalization: $P_n(k_j, z_j; \xi_n)$ in terms of general basis functions such as Chebyshev polynomials.
- Generalization: minimize criteria other than Euler equation error.
- So 3 choices: minimization criterion, basis functions, grid points.
- Can spline basis functions, for example at ZLB or default threshold.
- Optimizing these choices is research frontier.
  - Sparse grids (e.g. Smolyak): more efficient set of grid points than full tensor product.
  - Adaptive grids: concentrate grid points in places in state space visited frequently and where model is especially non-linear.
- Also recent progress in how to solve for basis function coefficients (e.g. collocation).

Aside: Chebyshev polynomials

- Definition: $T_0(x) = 1$, $T_1(x) = x$, $T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$.
- A few: $T_0(x) = 1$, $T_1(x) = x$, $T_2(x) = 2x^2 - 1$, $T_3(x) = 4x^3 - 3x, \ldots$
- Chebyshev polynomials are basis functions: can write any polynomial of order N as $\sum_{n=1}^{N} a_n T_n(x)$.
- Chebyshev polynomials are orthogonal basis functions.

General projection algorithm

1. Choose basis functions and polynomial order $\Psi_0, \ldots, \Psi_n: \; P_n(k,z;\xi_n) = \sum_{m=1}^{n} \xi_m \Psi_m(k,z)$.

2. Choose model policy function(s) to approximate and equation error(s) to minimize, e.g. $c(k,z) \approx P_n(k,z;\xi_n)^{-1/\gamma}$ and $d(k,z) = P_n(k,z;\xi_n) - E\left[P_n(k',z';\xi_n)\left(\alpha z_j'\left(k_j'\right)^{\alpha-1} + 1 - \delta\right)\right]$.

3. Choose set of nodes $\mathscr{J} = \{k_1, z_1\}, \{k_2, z_2\}, \ldots$.

4. Miminize chosen criterion at nodes, i.e. $\hat{\xi}_n = \arg\min_{\xi_n} \sum_{j \in \mathscr{J}} d(k_j, z_j)^2$.

Parallelization

- For estimation, solution method needs to be fast.
- Common bottleneck is evaluating $\sum_{j \in \mathscr{J}} d(k_j, z_j)^2$.
- Why? At each node, need to compute expectation.
- Very parallelizable: calculation of $d(k_1, z_1)$ independent of $d(k_j, z_j)$ for $j \neq 1$.
- As easy as PARFOR instead of FOR in Matlab.

Outline

Overview

- Assume you have solved the model and obtained policy functions $c_t = c(k_t, z_t), k_t = k(k_t, z_t)$.

- These are approximations, but we now ignore approximation error.

- The policy functions depend on model parameters $\theta = (\alpha \quad \beta \quad \delta \quad \gamma \quad \rho \quad \sigma)$.

- Note: may not depend on all parameters (e.g. σ drops out if first-order perturbation).

- Given data vector Y_t, how can you estimate parameters θ?

Outline

Overview

- Simulated method of moments, minimum distance, and indirect inference all refer to procedures that minimize the distance between a set of moments generated from simulated data and actual data.
- Examples:
  - Auto-covariances of data series.
  - Coefficient from regressing consumption growth on output growth.
- Moments not necessarily causally identified, e.g. consumption growth and output growth are jointly endogenous variables. Comparison of model–to–data moments still valid.
- Distance metric typically identity matrix or weighted by precision of data moments.
- Analysis extends to GMM, in which case data moments are zeros. GMM does not necessarily require a full model solution (e.g. single equation estimation).

ALGORITHM

1. Collect some moments of the data in the vector $\hat{m}$.

2. Simulate data given parameters θ by randomly drawing shocks and using decision rules to calculate endogenous variables. Burn early observations to reduce dependence on initial conditions.

3. Collect in $m(\theta)$ the moments of the simulated data at parameter vector θ.

4. Obtain $\hat{\theta} = \arg\min_{\theta} \left(\hat{m} - m(\theta)\right)' W \left(\hat{m} - m(\theta)\right)$ for weight matrix W.

Important to treat data and model exactly the same in this exercise. For example, if moments computed on de-trended data, should also de-trend simulated data before computing moments.

Outline

Overview

- Impulse responses are key object of interest in data.
- Natural to think about using them for parameter identification.
- Especially attractive when researcher is interested in particular aspect of model. For example, monetary policy.
- IRF-matching was subject of fierce debate in early 2000s.
- Question: what if model does not admit VAR representation used in data to estimate IRFs?
- Answer (Chari,Kehoe,McGrattan, JME, 2008): compare data IRF to model IRF *computed by estimating VAR on model data*.
- This is just a special case of indirect inference.

Outline

Overview

- Same procedure as you learned in econometrics.

- Rarely used this way in practice because of lack of identification (likelihood function flat). Can reduce problem by externally calibrating some parameters or imposing priors (Quasi-Bayesian ML, covered next).

- Likelihood approach also introduces problem of stochastic singularity (next slide).

Stochastic singularity

- Recall $\gamma = 1, \delta = 1 \Rightarrow c_t = (1 - \alpha\beta) z_t k_t^{\alpha}$.
- In words, consumption-output ratio is flat. (Why?)
- In data, consumption-output ratio fluctuates over time. Therefore, likelihood function is zero for any set of parameters.
- Solution:
  1. Add measurement error to data (blame inability of model to fit data on the data).
  2. Add unobserved shocks (say to time preference β_t).
- General rule: for every observable, require at least one unobservable shock or measurement error.
- Measurement error is misnomer since also (mostly) reflects model mis-specification.

Kalman filter redux

- How do you introduce measurement error or unobserved shocks?
- Answer: state space representation and Kalman filter. Recall:

$$\text{Observation:} \qquad y_t = Hs_t + w_t,$$
$$\text{State:} \qquad s_t = Fs_{t-1} + v_t.$$

- Introduce measurement error via w_t, e.g. $\underbrace{\begin{bmatrix} k_t^* \end{bmatrix}}_{y_t} = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} k_t \\ z_t \end{bmatrix}}_{s_t} + \underbrace{\begin{bmatrix} u_{k,t} \end{bmatrix}}_{w_t}$.

  Important: agents in model know and use k_t, not k_t^*.
- Introduce unobserved shocks as variables in state but not observation equation. Important: need to re-solve model (obtain new decision rules) to reflect new shocks. E.g. consumption and capital would depend on fluctuations in β_t.
- Can also obtain unobserved shocks by inverting policy rules. E.g.: $k_{t+1} = a_{kk}k_t + a_{kz}z_t \Rightarrow z_t = (k_{t+1} - a_{kk}k_t)/a_{kz}, \varepsilon_t = z_t - \rho z_{t-1}$.

Unobserved shocks versus measurement error

- Unobserved shocks popularized in Smets and Wouters (2003; 2007):
  - Medium-scale DSGE model: RBC core + sticky prices, sticky wages, habit formation in consumption, investment adjustment costs, variable capital utilization, etc.
  - Seven shocks: TFP, risk premium, investment-specific technology, wage mark-up, price mark-up, government spending, monetary policy.
- Paul Romer critique: leading models “attribute fluctuations in aggregate variables to imaginary causal forces.”
- Some economists dislike measurement error because it is “black box.”
- My view: stochastic singularity arises because the model is not a perfect representation of the true data generating process. Measurement error is transparent approach to assessing how mis-specified model is.

Romer description of Smets and Wouters shocks

- A general type of phlogiston that increases the quantity of consumption goods produced by given inputs
- An "investment-specific" type of phlogiston that increases the quantity of capital goods produced by given inputs
- A troll who makes random changes to the wages paid to all workers
- A gremlin who makes random changes to the price of output
- Aether, which increases the risk preference of investors
- Caloric, which makes people want less leisure

Romer annotation of Smets and Wouters

While “demand” shocks such as the [aether AKA] risk premium, exogenous spending, and investment-specific [phlogiston AKA] technology shocks explain a significant fraction of the short-run forecast variance in output, both the [troll’s] wage mark-up (or [caloric AKA] labor supply) and, to a lesser extent, output-specific [phlogiston AKA] technology shocks explain most of its variation in the medium to long run. ... Third, inflation developments are mostly driven by the [gremlin’s] price mark-up shocks in the short run and the [troll’s] wage mark-up shocks in the long run (p. 587).

Unobserved shocks versus measurement error

- Unobserved shocks popularized in Smets and Wouters (2003; 2007):
  - Medium-scale DSGE model: RBC core + sticky prices, sticky wages, habit formation in consumption, investment adjustment costs, variable capital utilization, etc.
  - Seven shocks: TFP, risk premium, investment-specific technology, wage mark-up, price mark-up, government spending, monetary policy.
- Paul Romer critique: leading models "attribute fluctuations in aggregate variables to imaginary causal forces."
- Some economists dislike measurement error because it is "black box."
- My view: stochastic singularity arises because the model is not a perfect representation of the true data generating process. Measurement error is transparent approach to assessing how mis-specified model is.

Serially-correlated measurement error

- True measurement error possibly serially uncorrelated. Measurement error due to model mis-specification likely serially correlated.

- Suppose $k_t^* = k_t + u_{k,t},\ u_{k,t} = \gamma u_{k,t-1} + e_{k,t},\ e_{k,t} \sim iidN(0, \sigma_{e,k}^2)$.

- State space representation:

Observation: $$\underbrace{\begin{bmatrix} k_t^* \end{bmatrix}}_{y_t} = \underbrace{\begin{bmatrix} 1 & 0 & 1 \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} k_t \\ z_t \\ u_{k,t} \end{bmatrix}}_{s_t} + \underbrace{\begin{bmatrix} 0 \end{bmatrix}}_{w_t},$$

State: $$\underbrace{\begin{bmatrix} k_t \\ z_t \\ u_{k,t} \end{bmatrix}}_{s_t} = \underbrace{\begin{bmatrix} a_{kk} & a_{kz} & 0 \\ 0 & \rho & 0 \\ 0 & 0 & \gamma \end{bmatrix}}_{F} \underbrace{\begin{bmatrix} k_{t-1} \\ z_{t-1} \\ u_{k,t-1} \end{bmatrix}}_{s_{t-1}} + \underbrace{\begin{bmatrix} 0 \\ \varepsilon_t \\ e_{k,t} \end{bmatrix}}_{v_t}.$$

Outline

Overview

- With many parameters, likelihood function typically not well-behaved enough to do standard ML.

- Bayesian ML imposes priors on parameters.

- Recall Bayes rule for parameters θ and data Y:

$$\begin{aligned}
\text{Posterior:} \quad & p(\theta|Y) = \frac{p(\theta, Y)}{p(Y)} = \frac{p(Y|\theta)\,p(\theta)}{p(Y)} = \frac{L(\theta|Y)\,p(\theta)}{p(Y)}, \\
\text{where:} \quad & \\
\text{Prior:} \quad & p(\theta), \\
\text{Likelihood:} \quad & L(\theta|Y) = p(Y|\theta).
\end{aligned}$$

Interpreting prior: artificial data

- Often prior chosen based on information from some other studies and data.

- In case of conjugate prior, can make analogy exact.

- Definition: a conjugate prior is a distribution for θ such that the posterior has the same distribution as the prior.

- Often analytically convenient.

Example: Beta distribution and Bernoulli

- Beta distribution: $f(x;\alpha,\beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha,\beta)}$, $x \subseteq [0,1]$, where the Beta function is: $\mathrm{B}(\alpha,\beta) = \int_0^1 x^{\alpha-1}(1-x)^{\beta-1}\,dx = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$.
- Moments: Mean: $\mu(\alpha,\beta) = \frac{\alpha}{\alpha+\beta}$, Variance: $\sigma^2(\alpha,\beta) = \frac{\alpha\beta}{(\alpha+\beta+1)(\alpha+\beta)^2}$.
- Suppose data are n Bernoulli trials with s successes and f failures:

$$\text{Prior:} \qquad p(\theta;\alpha,\beta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{\mathrm{B}(\alpha,\beta)},$$

$$\text{Likelihood:} \qquad p(Y|\theta) = \binom{s+f}{s}\theta^s(1-\theta)^f,$$

$$\text{Posterior:} \qquad p(\theta|Y) = \frac{p(Y|\theta)\,p(\theta)}{\int p(Y|\theta')\,p(\theta')\,d\theta'} = \frac{\theta^{\alpha-1+s}(1-\theta)^{\beta-1+f}}{\mathrm{B}(\alpha+s,\beta+f)},$$

$$\text{Post. mean:} \qquad \mu(\alpha+s,\beta+f) = \frac{\alpha+s}{\alpha+\beta+s+f}.$$

- Interpretation: prior based on previous experiment with $n_0 = \alpha+\beta$ observations.

Why MCMC?

- We want to compute $E[g(\theta)|Y]$, e.g. the posterior
  - mean: $g(\theta) = \theta$,
  - coverage ratio: $g(\theta) = 1\{\theta \in \theta^c\}$.
- Monte Carlo sampling: randomly draw from posterior and compute statistics of interest.
- But numerical techniques required to evaluate likelihood.
- Therefore numerical techniques also required to evaluate posterior.
- Sampling directly from posterior would require random number generator for a numerical and typically unusual distribution.
- Instead use Markov Chain Monte Carlo (MCMC).
- Popular choice is random walk Metropolis–Hasting algorithm.

Properties of MCMC

- MCMC produces sequence $\{\theta^i\}_{i=1}^N$ s.t. $\{\theta^i\}$ converges in distribution to the posterior distribution as $N \to \infty$.

- Compute moments and quantiles of posterior from $\{\theta^i\}$.

- Basic idea is to construct Markov transition kernel $K(\theta^i|\theta^{i-1})$ such that if θ^{i-1} is draw from the posterior, so is θ^i.

Random-walk MH algorithm

1. Start with initial value θ^0.
2. In ith iteration, draw $\theta^* \sim q(\theta^*|\theta^{i-1})$.
   - $q(\theta^*|\theta^{i-1})$ is “stand-in” density. In R-WMH, $q(\theta^*|\theta^{i-1}) = N(\theta^{i-1}, \Sigma)$.
3. Set $\alpha = \min\left\{\frac{p(\theta^*|Y)}{p(\theta^{i-1}|Y)}, 1\right\}$. α is the transition probability.
4. Set $\theta^i = \theta^*$ with probability α and $\theta^i = \theta^{i-1}$ otherwise.

Idea: if posterior density of θ^i larger than posterior density of θ^{i-1}, always update to θ^i. Otherwise, update with probability declining in ratio to make sure chain doesn't get stuck at a local optimum.

Practical considerations

- Discard burn-in phase to reduce dependence on θ^0.
- Need to parameterize Σ. Can do "on-the-fly" during burn-in period. Want acceptance rates in the neighborhood of 0.3 (Gelman et al., 2004).
- Can draw whole vector θ or block-by-block. Easier to calibrate Σ if block-by-block, but need more draws in total.
- Generate several independent chains from different starting values. These should converge to same distribution.
- RWMH not necessarily efficient since many draws away from "typical set." Fernández-Villaverde and Guerrón-Quintana ("Estimating DSGE Models") advocate Hamiltonian Monte Carlo.
- Make Dynare your ally, not your master.