Part III: Likelihood Evaluation of Nonlinear DSGE Models

Fei Tan

Chaifetz School of Business
Saint Louis University

Mini-Course Prepared for School of Finance
Zhejiang Gongshang University

June 26, 2018

Introduction

- State space representation of DSGE models amenable to likelihood evaluation
  - nonlinearity + non-Gaussian shocks
  - numerical evaluation via particle filter

- Main references
  - Doucet & Johansen (2011), "*A Tutorial on Particle Filtering and Smoothing: Fifteen years later*", Handook of Nonlinear Filtering, Oxford University Press
  - DeJong & Dave (2011), "*Structural Macroeconometrics*", Princeton University Press
  - Herbst & Schorfheide (2015), "*Bayesian Estimation of DSGE Models*", Princeton University Press

The Road Ahead...

- Nonlinear state space models
  - small new Keynesian DSGE
  - state space representation
- Likelihood evaluation
  - generic filtering algorithm
  - bootstrap particle filter
  - generic particle filter & adaption

Small New Keynesian DSGE

- Dynamic IS relation

$$\hat{y}_t = \mathbb{E}_t \hat{y}_{t+1} + \hat{g}_t - \mathbb{E}_t \hat{g}_{t+1} - \frac{1}{\tau}(\hat{R}_t - \mathbb{E}_t \hat{\pi}_{t+1} - \mathbb{E}_t \hat{z}_{t+1})$$

- New Keynesian Phillips curve

$$\hat{\pi}_t = \beta \mathbb{E}_t \hat{\pi}_{t+1} + \kappa(\hat{y}_t - \hat{g}_t)$$

- Monetary policy rule

$$\hat{R}_t = \rho_R \hat{R}_{t-1} + (1-\rho_R)\psi_1 \hat{\pi}_t + (1-\rho_R)\psi_2(\hat{y}_t - \hat{g}_t) + \epsilon_{R,t}$$

- Exogenous driving processes

$$\hat{g}_t = \rho_G \hat{g}_{t-1} + \epsilon_{G,t}, \quad \hat{z}_t = \rho_Z \hat{z}_{t-1} + \epsilon_{Z,t}$$

State Space Representation

- Transition equations

$$s_t = \Phi(s_{t-1}, \epsilon_t; \theta)$$

- Measurement equations

$$\underbrace{\begin{pmatrix} \mathsf{YGR}_t \\ \mathsf{INF}_t \\ \mathsf{INT}_t \end{pmatrix}}_{y_t} = \underbrace{\begin{pmatrix} \gamma^{(Q)} \\ \pi^{(A)} \\ \pi^{(A)} + r^{(A)} + 4\gamma^{(Q)} \end{pmatrix}}_{D(\theta)} + \underbrace{\begin{pmatrix} \hat{y}_t - \hat{y}_{t-1} + \hat{z}_t \\ 4\hat{\pi}_t \\ 4\hat{R}_t \end{pmatrix}}_{Z(\theta)s_t} + u_t$$

- Distributional assumption: $\epsilon_t \sim p_\epsilon(\cdot;\theta)$, $u_t \sim p_u(\cdot;\theta)$

Generic Filtering Algorithm

- Initialization: set $p(s_0|y_0,\theta) = p(s_0|\theta)$
- recursion: for $t = 1, \ldots, T$

  step 1: forecasting s_t via model solution

  $$p(s_t|y_{1:t-1},\theta) = \int p(s_t|s_{t-1},\cancel{y_{1:t-1}},\theta)p(s_{t-1}|y_{1:t-1},\theta)ds_{t-1}$$

  step 2: forecasting y_t via measurement equations

  $$p(y_t|y_{1:t-1},\theta) = \int p(y_t|s_t,\cancel{y_{1:t-1}},\theta)p(s_t|y_{1:t-1},\theta)ds_t$$

  step 3: filtering s_t via Bayes' Theorem

  $$p(s_t|y_{1:t},\theta) = \frac{p(y_t|s_t,\cancel{y_{1:t-1}},\theta)p(s_t|y_{1:t-1},\theta)}{p(y_t|y_{1:t-1},\theta)}$$

- Likelihood evaluation: $p(y_{1:T}|\theta) = \prod_{t=1}^{T} p(y_t|y_{1:t-1},\theta)$

Bootstrap Particle Filter

- Initialization: draw $\{s_0\}_{i=1}^N \sim p(s_0)$ and set $W_0^i = \frac{1}{N}$
- Recursion: for $t = 1, \ldots, T$

  step 1: propagation

$$s_t^i = \Phi(s_{t-1}^i, \epsilon_t^i; \theta), \quad \epsilon_t^i \sim p_\epsilon(\cdot; \theta)$$

  step 2: likelihood evaluation

$$\hat{p}(y_t | y_{1:t-1}, \theta) = \sum_{i=1}^N W_{t-1}^i p(y_t | s_t^i, \theta)$$

  step 3: filtering & resampling

$$W_t^i = \frac{W_{t-1}^i p(y_t | s_t^i, \theta)}{\hat{p}(y_t | y_{1:t-1}, \theta)} \quad \Rightarrow \quad \{(s_t^i, \epsilon_t^i, W_t^i)\}_{i=1}^N$$

- Likelihood aggregation: $\hat{p}(y_{1:T} | \theta) = \prod_{t=1}^T \hat{p}(y_t | y_{1:t-1}, \theta)$

Generic Particle Filter

- Initialization: draw $\{s_0\}_{i=1}^N \sim p(s_0)$ and set $W_0^i = \frac{1}{N}$
- Recursion: for $t = 1, \ldots, T$

  step 1: propagation

$$s_t^i = \Phi(s_{t-1}^i, \epsilon_t^i; \theta), \quad \epsilon_t^i \sim g_t(\cdot; \theta)$$

  and define importance weight

$$w_t^i = p_\epsilon(\epsilon_t^i; \theta) / g_t(\epsilon_t^i; \theta)$$

  step 2: likelihood evaluation

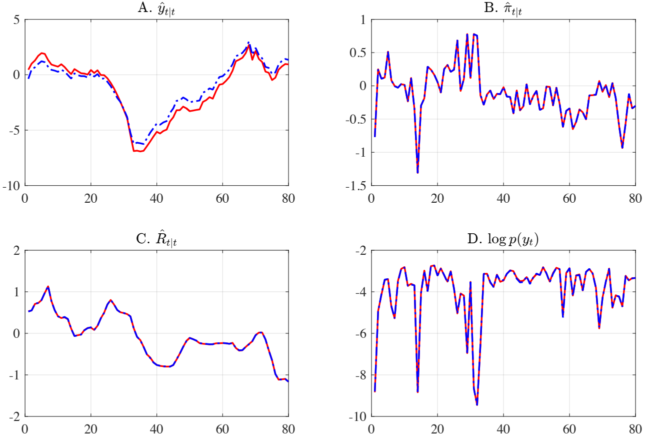
$$\hat{p}(y_t | y_{1:t-1}, \theta) = \sum_{i=1}^N W_{t-1}^i p(y_t | s_t^i, \theta) w_t^i$$

  step 3: filtering & resampling (same)
- Likelihood aggregation: $\hat{p}(y_{1:T} | \theta) = \prod_{t=1}^T \hat{p}(y_t | y_{1:t-1}, \theta)$

MATLAB Pseudo-Code

```
function [fs,loglik] = ParticleFilter(Y,SSR)
[...]
for t = 1:T
    % Period-t propagation
    [swarm,w] = Transit(Y(t,:),SSR,swarm);

    % Period-t log likelihood
    w_inc = log(W)+w+Measure(Y(t,:),SSR,swarm);
    loglik(t) = log(sum(exp(w_inc)));

    % Period-t filtering
    W = exp(w_inc-loglik(t));

    % Period-t resampling
    [...]

    % Period-t filtered states
    fs(:,t) = swarm*W;
end
```

Assessing Accuracy



- Particle filter (red solid) vs Kalman filter (blue dashed) with $N = 500$ particles at posterior mode