A Primer on Bayesian Econometrics with Macroeconomic Applications

Fei Tan

Chaifetz School of Business
Saint Louis University

Chinese Academy of Finance and Development
Central University of Finance and Economics

May 28, 2018

Frequentist v.s. Bayesian

Probability axioms

1. $0 \leq \mathbb{P}(A) \leq 1$ for any event A
2. $\mathbb{P}(A) = 1$ if event A represents logical truth
3. $\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B)$ for disjoint events A and B
4. $\mathbb{P}(A|B) = \mathbb{P}(A \cap B)/\mathbb{P}(B)$ (conditional probability)

- Any assignment of probabilities must satisfy above axioms
  - Frequentists assign probabilities to events that describe outcome of *repeated* experiment
  - Bayesians assign 'subjective' probability to any uncertain event (de Finetti's (1990) coherency principle)
- How likely it rains tomorrow?

What Is the Lecture About?

- Introduce Bayesian computational devices applicable in macroeconomics and related fields
- Why Bayesian paradigm
  - handle sophisticated economic models
  - uncertainty in forecasting & policymaking
- Main references
  - Greenberg (2008), "*Introduction to Bayesian Econometrics*", Cambridge University Press
  - Herbst & Schorfheide (2015), "*Bayesian Estimation of DSGE Models*", Princeton University Press
  - An & Schorfheide (2007), "*Bayesian Analysis of DSGE Models*", Econometric Reviews

The Road Ahead...

- Part I: Fundamentals of Bayesian econometrics
  - prior, likelihood, and posterior
  - posterior inference
  - classical simulation methods
  - Markov chain Monte Carlo methods
- Part II: Macroeconomic applications
  - solving linearized DSGE model
  - prior distribution and likelihood function
  - tailored randomized block algorithm
  - selected further readings

Part I: Fundamentals of Bayesian econometrics

Prior, Likelihood, and Posterior

Bayes Theorem

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \propto p(y|\theta)p(\theta)$$

- Learning of random vector $Y = [Y_1, \dots, Y_n]'$
  - call their realizations $y = [y_1, \dots, y_n]'$ *data*
  - parametric distribution $\mathbb{P}_\theta$
  - learning of unknown parameter θ
- Bayesian approach treats θ as being random
  - start with *prior* density $p(\theta)$
  - update by *likelihood* function $p(y|\theta)$
  - *posterior* density $p(\theta|y)$ proportional to prior $\times$ likelihood

Coin-Tossing Example

- Likelihood function
  - single toss: $y_i = 1$ if head or 0 if tail

$$y_i \sim \text{Bernoulli}(\theta) \quad \Rightarrow \quad p(y_i|\theta) = \theta^{y_i}(1-\theta)^{1-y_i}$$

  - n independent tosses

$$p(y_1, \ldots, y_n|\theta) = \theta^{\sum y_i}(1-\theta)^{n-\sum y_i}$$

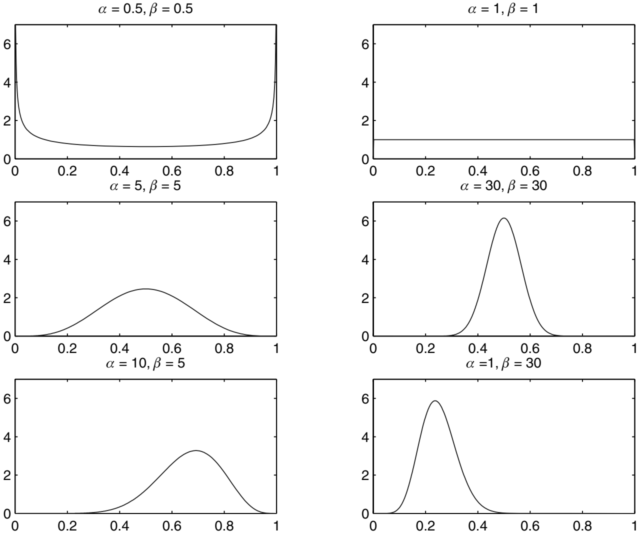
- Prior density: $\theta \sim \text{Beta}(\alpha, \beta)$

$$p(\theta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}, \quad 0 \le \theta \le 1, \quad \alpha, \beta > 0$$

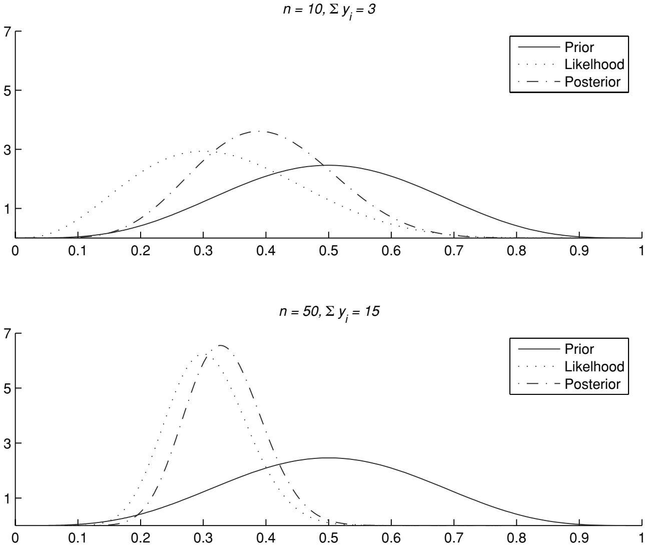
- Posterior density: $\theta \sim \text{Beta}(\alpha + \sum y_i, \beta + n - \sum y_i)$

$$p(\theta|y) \propto \theta^{\alpha+\sum y_i - 1}(1-\theta)^{\beta+n-\sum y_i - 1}$$

Beta Distribution



Sample Size



n = 10, Σ y_i = 3
Prior
Likelhood
Posterior
n = 50, Σ y_i = 15
Prior
Likelhood
Posterior

Posterior Inference

- Identification
  - observational equivalence: $p(y|\theta_1) = p(y|\theta_2)$ for all y
  - identification through data v.s. prior
- Posterior estimates
  - point estimate: $\hat{\theta} = \arg\max \int L(\hat{\theta}, \theta) p(\theta|y) d\theta$
  - interval estimate: $\mathbb{P}(\theta_L \leq \theta \leq \theta_U) = .95$
- Prediction: $p(y_{n+1}|y) = \int p(y_{n+1}|\theta, y) p(\theta|y) d\theta$
- Model comparison (& averaging)

$$\underbrace{\frac{p(\mathcal{M}_1|y)}{p(\mathcal{M}_2|y)}}_{\text{posterior odd}} = \underbrace{\frac{p(\mathcal{M}_1)}{p(\mathcal{M}_2)}}_{\text{prior odd}} \underbrace{\frac{p(y|\mathcal{M}_1)}{p(y|\mathcal{M}_2)}}_{\text{Bayes factor}}$$

where $$\underbrace{p(y|\mathcal{M}_i)}_{\text{marginal likelihood}} = \int p(y|\theta_i, \mathcal{M}_i) p(\theta_i|\mathcal{M}_i) d\theta_i$$

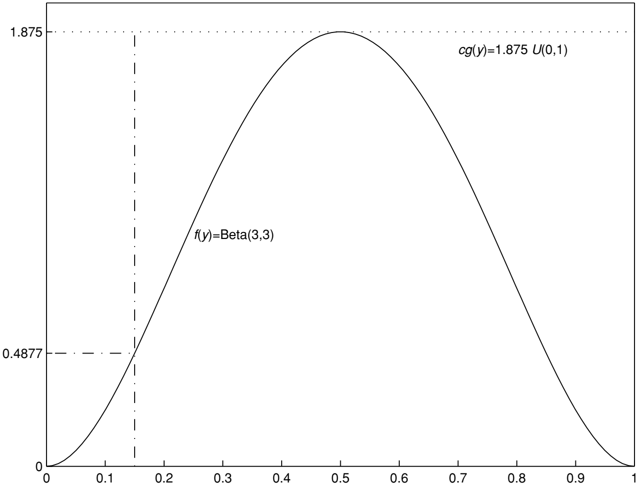
Method 1: Probability Integral Transform

- Goal: represent $f(y)$ with $\mathbb{P}(Y \leq y) = F(y)$ by simulating *independent* samples
  - step 1: draw $u \sim \text{Uniform}(0, 1)$
  - step 2: return $y = F^{-1}(u)$ as a draw from $f(y)$
- Example: $f(y) = \frac{3}{8}y^2$ for $0 \leq y \leq 2$ and 0 otherwise
  - compute distribution function $F(y) = \frac{1}{8}y^3$
  - draw $u \sim \text{Uniform}(0, 1) \Rightarrow y = 2u^{\frac{1}{3}} \sim f(y)$
- Remark: inefficient in multivariate case

Method 2: Accept-Reject

- Goal: represent target $f(y)$ by simulating *independent* samples from proposal $g(y)$ with $f(y) \leq cg(y)$ for some $c \geq 1$
  - step 1: draw $y \sim g(y)$
  - step 2: draw $u \sim \text{Uniform}(0, 1)$
  - step 3: accept y as a draw from $f(y)$ if $u \leq \frac{f(y)}{cg(y)}$; otherwise reject and return to step 1
- Example: sample from Beta$(3, 3)$
  - choose proposal Uniform$(0, 1)$
  - set $c = 1.875$
- Remark: difficult to find proposal in multivariate case

Method 2: Accept-Reject (Cont'd)



Method 3: Importance Sampling

- Goal: estimate $\mathbb{E}[g(X)] = \int g(x)f(x)dx$ by simulating *independent* samples from proposal $h(x)$
  - step 1: draw a sample $\{x_i\}_{i=1}^M$ from $h(x)$
  - step 2: compute

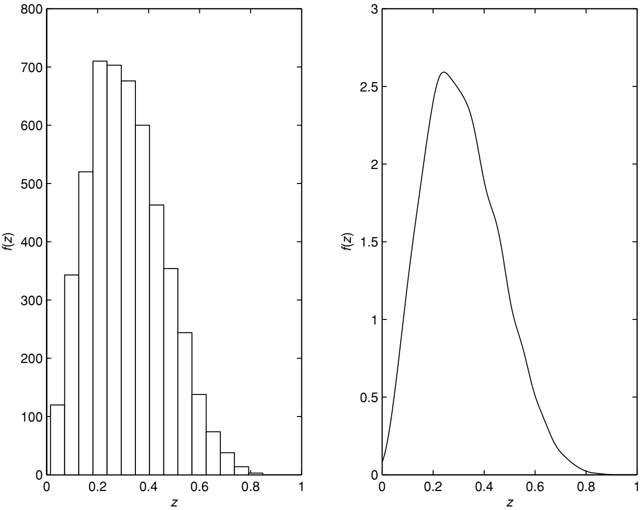
$$\mathbb{E}[g(X)] \approx \frac{1}{M}\sum_{i=1}^{M} g(x_i)\ \underbrace{f(x_i)/h(x_i)}_{\text{importance weight}}$$

- Example: $\mathbb{E}[1/(1+x^2)]$ where $x \sim \text{Exp}(1)$ truncated to $[0,1]$
  - step 1: draw a sample $\{x_i\}_{i=1}^M$ from $\text{Beta}(2,3)$
  - step 2: compute

$$\frac{1}{M}\sum_{i=1}^{M} \frac{1}{1+x_i^2}\,\frac{e^{-x_i}}{1-e^{-1}}\,\frac{\text{Beta}(2,3)}{x_i(1-x_i^2)}$$

- Remark: difficult to find proposal in multivariate case

Using Simulated Output



- $\{x_i\}_{i=1}^{M} \sim \text{Beta}(3,3)$, $\{y_i\}_{i=1}^{M} \sim \text{Beta}(5,3)$, $z_i = x_i y_i$
- $\{z_i\}_{i=1}^{M}$ represent distribution of $Z = XY$

MCMC Algorithm: Big Picture

A central equation

$$\int_A \int_{\mathbb{R}^d} p(x,y)\pi^*(x)dxdy = \int_A \pi^*(y)dy, \quad \forall A \in \mathcal{B}(\mathbb{R}^d)$$

- What is Markov chain theory doing? Know transition kernel $p(\cdot,\cdot)$, find invariant distribution $\pi^*(\cdot)$

$$\int_A \int_{\mathbb{R}^d} p(x,y)\pi^{(n-1)}(x)dxdy = \int_A \pi^{(n)}(y)dy \to \int_A \pi^*(y)dy$$

- Markov chain Monte Carlo (MCMC) is doing opposite: know $\pi^*(\cdot)$, find corresponding $p(\cdot,\cdot)$ such that

$$\pi^*(x)p(x,y) = \pi^*(y)p(y,x) \quad \text{(reversibility)}$$

- Remark: greatly broaden scope of Bayesian methods though at cost of simulating *dependent* samples

Metropolis-Hastings Algorithm

- Generic MH algorithm
  - initialization: set $\theta^{(0)} = \arg\max p(y|\theta)p(\theta)$
  - recursion: for $k = 1, \ldots, N$
    step 1: draw $\vartheta \sim q(\theta^{(k-1)}, \cdot)$ (proposal density)
    step 2: set $\theta^{(k)} = \vartheta$ with probability of move

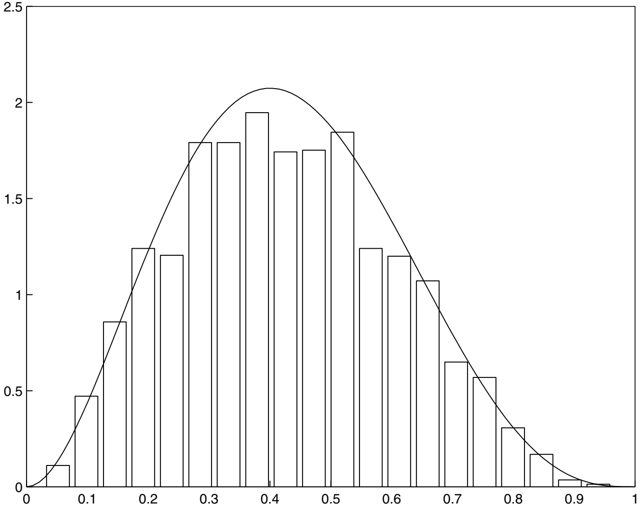
$$\alpha(\theta^{(k-1)}, \vartheta) = \min\left\{ \frac{p(y|\vartheta)p(\vartheta)}{p(y|\theta^{(k-1)})p(\theta^{(k-1)})} \frac{q(\vartheta, \theta^{(k-1)})}{q(\theta^{(k-1)}, \vartheta)}, 1 \right\}$$

    otherwise set $\theta^{(k)} = \theta^{(k-1)}$
  - discard burn-in phase
- MH's choice of $p(\cdot, \cdot)$

$$p_{\mathsf{MH}}(\theta^{(k-1)}, \vartheta) \equiv q(\theta^{(k-1)}, \vartheta)\alpha(\theta^{(k-1)}, \vartheta)$$

satisfies reversibility with invariant distribution $p(\theta|y)$

MH Output



- Target Beta$(3, 4)$, proposal Uniform$(0, 1)$
- $M = 5,000$ after initial 500 burn-in

Random Walk MH

MATLAB pseudo-code

```
function [chain,rej] = RandomWalk_MH(c,Sigma)
[...]
for k = 1:N
    theta = mvt_rnd(chain(k-1,:),c^2*Sigma,inf,1);
    pk_next = PostKer(theta);
    alpha = min([exp(pk_next-pk_last) 1]);
    if rand > alpha                          % reject
        chain(k,:) = chain(k-1,:);
        rej = rej+1;
    else                                     % accept
        chain(k,:) = theta;
        pk_last = pk_next;
    end
end
```

Block-at-a-Time Algorithm

Conditional invariant distributions

$$\int_{A_1}\int_{\mathbb{R}^{d_1}} p_1(x_1, y_1|x_2)\pi^*_{1|2}(x_1|x_2)dx_1 dy_1 = \int_{A_1} \pi^*_{1|2}(y_1|x_2)dy_1$$

$$\int_{A_2}\int_{\mathbb{R}^{d_2}} P_2(x_2, y_2|x_1)\pi^*_{2|1}(x_2|x_1)dx_2 dy_2 = \int_{A_2} \pi^*_{2|1}(y_2|x_1)dy_2$$

- Product of kernels principle
  - $p_1(x_1, y_1|x_2)p_2(x_2, y_2|y_1)$ has invariant density $\pi^*(x_1, x_2)$
  - underlying Gibbs, MH within Gibbs, & TaRB
- Reference: Chib & Greenberg (1995), "*Understanding Metropolis-Hastings Algorithm*", American Statistician

Part II: Macroeconomic Applications

Simple New Keynesian Model

- 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

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

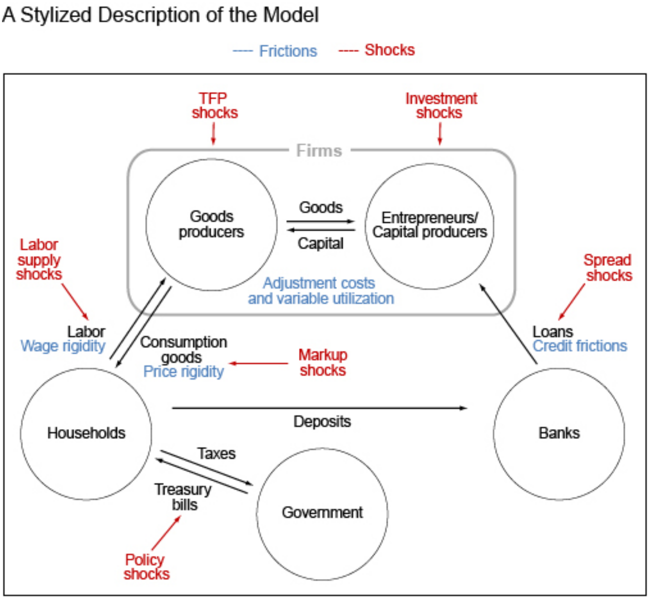
- Monetary policy rule

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

- Exogenous driving processes

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

FRB-NY Model



A Stylized Description of the Model
---- Frictions
---- Shocks
TFP shocks
Investment shocks
Firms
Goods producers
Goods
Capital
Entrepreneurs/ Capital producers
Adjustment costs and variable utilization
Labor supply shocks
Labor
Wage rigidity
Consumption goods
Price rigidity
Markup shocks
Spread shocks
Loans
Credit frictions
Households
Deposits
Banks
Taxes
Treasury bills
Government
Policy shocks

State Space Form

- Solving linear rational expectations models
  - time domain (our focus), e.g. Sims (2001)

$$s_t = C(\theta) + G(\theta)s_{t-1} + M(\theta)\epsilon_t$$

  - frequency domain, e.g. Walker & Tan (2015)

$$s_t = \sum_{k=0}^{\infty} C_{\theta,k}\epsilon_{t-k} \equiv C_\theta(L)\epsilon_t$$

- Measurement equations

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

- Distributional assumption: $\epsilon_t \sim \mathbb{N}(0, \Sigma_\epsilon)$, $u_t \sim \mathbb{N}(0, \Sigma_u)$

Sims' Method

MATLAB pseudo-code

```
function [C,G,M,eu] = SolveModel(para,ssp,P,V)
[...]
% Input equations
j = j+1;        % new Keynesian Phillips curve
G0(j,V.model.pi) = 1;
G0(j,V.model.E_pi) = -ssp(P.beta);
G0(j,V.model.y) = -para(P.kappa);
G0(j,V.model.g) = para(P.kappa);
j = j+1;        % trend shock
G0(j,V.model.z) = 1;
G1(j,V.model.z) = para(P.rho_Z);
Psi(j,V.shock.eps_Z) = 1;
j = j+1;        % E_pi error
G0(j,V.model.pi) = 1;
G1(j,V.model.E_pi) = 1;
Pi(j,V.fore.pi) = 1;
% Solve model (see Chris Sims' webpage)
[G,C,M,~,~,~,~,eu] = gensys(G0,G1,CC,Psi,Pi);
```

Prior Distribution

Name	Domain	Density	Mean	S.D.
τ	$\mathbb{R}^+$	$\mathbb{G}$	2.00	0.50
κ	$\mathbb{R}^+$	$\mathbb{G}$	0.20	0.10
ψ_1	$(1, \infty)$	$\mathbb{G}$	1.50	0.25
ψ_2	$\mathbb{R}^+$	$\mathbb{G}$	0.50	0.25
$r^{(A)}$	$\mathbb{R}^+$	$\mathbb{G}$	0.50	0.50
$\pi^{(A)}$	$\mathbb{R}^+$	$\mathbb{G}$	7.00	2.00
$\gamma^{(Q)}$	$\mathbb{R}$	$\mathbb{N}$	0.40	0.20
ρ_R	$[0, 1)$	$\mathbb{B}$	0.50	0.20
ρ_G	$[0, 1)$	$\mathbb{B}$	0.80	0.10
ρ_Z	$[0, 1)$	$\mathbb{B}$	0.66	0.15
σ_R	$\mathbb{R}^+$	$\mathbb{IG}$	0.50	0.26
σ_G	$\mathbb{R}^+$	$\mathbb{IG}$	1.25	0.65
σ_Z	$\mathbb{R}^+$	$\mathbb{IG}$	0.63	0.33

Prior Evaluation

MATLAB pseudo-code

```
function logprior = prior_pdf(x,mean,sd,type)

switch type
    case 'G'          % Gamma distribution
        a = mean^2/sd^2;
        b = sd^2/mean;
        logprior = log(gampdf(x,a,b));
    case 'N'          % Normal distribution
        logprior = log(normpdf(x,mean,sd));
    case 'B'          % Beta distribution
        a = -mean*(sd^2+mean^2-mean)/sd^2;
        b = (mean-1)*(sd^2+mean^2-mean)/sd^2;
        logprior = log(betapdf(x,a,b));
    case 'I1'         % Inv-Gamma type-1 distribution
        [...]
end
```

Likelihood Function

- Generic filter
  - initialization: set $p(s_0|y_0, \theta) = p(s_0|\theta)$
  - recursion: for $t = 1, \dots, T$

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

Kalman Filter

MATLAB pseudo-code

```
function [m_fs,V_fs,loglik] = KalmanFilter(Y,SSR)
[...]
for t = 1:T
    % Period-(t-1) predictive density
    m_ps = C+G*m_fs(:,t-1);
    V_ps = G*V_fs(:,:,t-1)*G'+M*V_e*M';

    % Period-t log likelihood
    m_py = D+Z*m_ps;
    V_py = Z*V_ps*Z'+V_u;
    loglik(t) = mvt_pdf(Y(t,:),m_py',V_py,inf);

    % Period-t filtering density
    gain = (V_ps*Z')/V_py;
    m_fs(:,t) = m_ps+gain*(Y(t,:)'-m_py);
    V_fs(:,:,t) = V_ps-gain*Z*V_ps;
end
```

TaRB-MH Algorithm

- A powerful and highly efficient MCMC approach
  - randomize number & components of blocks
  - tailor proposal to posterior location & curvature
- Tailored randomized block (TaRB) algorithm
  - initialization: set $\theta^{(0)} = \arg\max p(y|\theta)p(\theta)$
  - recursion: for $k = 1, \ldots, N$

    step 1: randomize blocks $(\theta_{k,1}, \theta_{k,2}, \ldots, \theta_{k,B_k})$

    step 2: tailor proposal density by optimization routine

    $$q_l(\theta_{k,l}|\theta_{k,-l}, y) = t(\theta_{k,l}|\hat{\theta}_{k,l}, V_{k,l}, \nu)$$

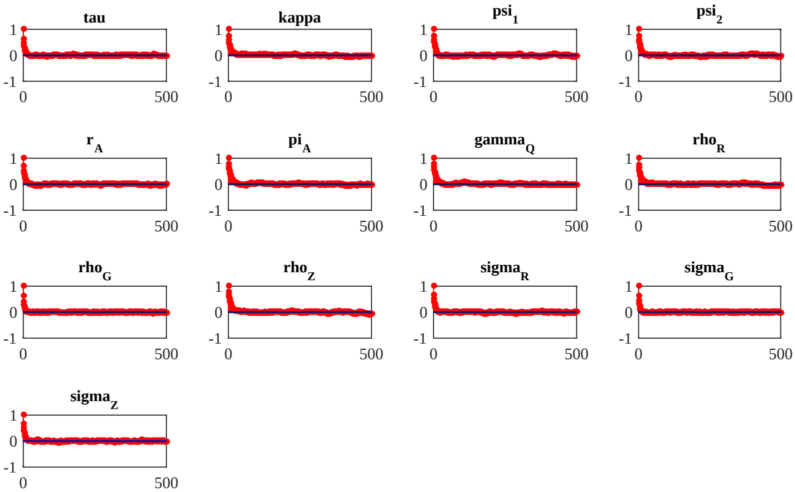
    step 3: update each block with $\alpha_l(\theta_{k,l}, \vartheta_{k,l}|\theta_{k,-l}, y)$
  - discard burn-in phase
- Chib & Ramamurphy (2010), "*TaRB MCMC Methods with Application to DSGE Models*", Journal of Econometrics

Posterior Distribution

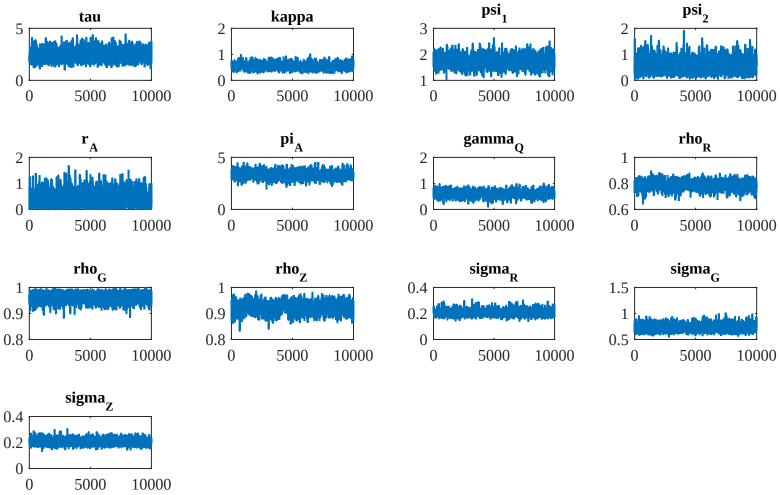
Name	Mean	90% interval	Ineff.
τ	2.435	$[1.712, 3.314]$	8.1
κ	0.543	$[0.362, 0.758]$	16.7
ψ_1	1.738	$[1.400, 2.093]$	10.1
ψ_2	0.570	$[0.198, 1.088]$	13.7
$r^{(A)}$	0.388	$[0.038, 0.869]$	9.4
$\pi^{(A)}$	3.379	$[2.789, 3.968]$	16.6
$\gamma^{(Q)}$	0.605	$[0.399, 0.806]$	16.4
ρ_R	0.791	$[0.735, 0.841]$	17.1
ρ_G	0.963	$[0.933, 0.987]$	7.4
ρ_Z	0.924	$[0.890, 0.956]$	18.0
σ_R	0.208	$[0.173, 0.247]$	10.2
σ_G	0.736	$[0.637, 0.856]$	7.0
σ_Z	0.209	$[0.172, 0.249]$	9.3

NOTES: number of draws = 10,000 after first 1,000 burn-in; computational time = 17m:32s; rejection rate = 45.9%; average number of blocks = 3.4

Autocorrelation Function



Trace Plot

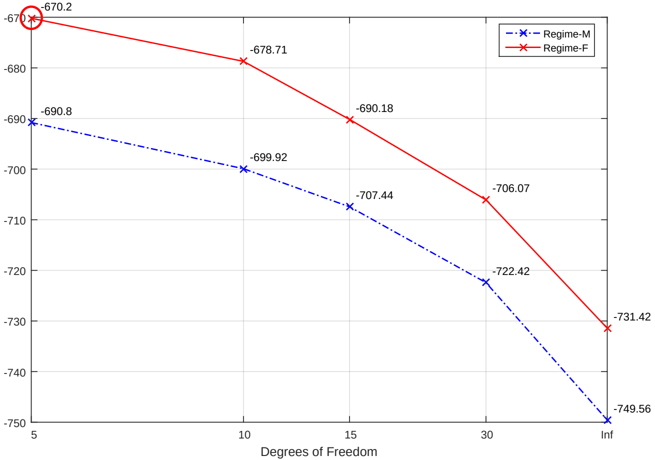


tau
kappa
psi1
psi2
rA
piA
gammaQ
rhoR
rhoG
rhoZ
sigmaR
sigmaG
sigmaZ
0
5000
10000

Monetary-Fiscal Policy Interaction

- Macro policies have two essential tasks to perform
  - determining inflation/price level
  - stabilizing government debt
- Conventional theory (regime-M)
  - MP 'actively' targets inflation by Taylor principle
  - FP 'passively' targets debt by fiscal adjustments
- Fiscal theory (regime-F)
  - FP 'actively' determines inflation
  - MP 'passively' maintains real value of debt
- Fundamental understandings of how macro economy works hinge on determining policy regime, e.g.
  - inflation monetary or fiscal phenomenon?
  - appropriate MP response to deflation?

Leeper-Traum-Walker Model



- Truth: regime-F with 5 degrees of freedom and 100 observations; 33 parameters, 48 equations, 8 observables
- True model yields highest marginal likelihood

Selected Further Readings

- Student-t shocks: Chib & Ramamurthy (2014), "*DSGE Models with Student-t Errors*", Econometric Reviews
- Stochastic volatility: Justiniano & Primiceri (2008), "*Time-Varying Volatility of Macro Fluctuations*", AER
- Regime-switching: Schorfheide (2005), "*Learning and Monetary Policy Shifts*", RED
- DSGE-VAR: Del Negro & Schorfheide (2004), "*Priors from General Equilibrium Models for VARs*", IER
- Prediction pool: Del Negro, Hasegawa & Schorfheide (2016), "*Dynamic Prediction Pools*", JoE
- Asset pricing: Rapach & Tan (2018), "*Asset Pricing with Recursive Preferences and Stochastic Volatility*", Manuscript