The Metropolis-Hastings Algorithm

June 8, 2012

The Plan

1. Understand what a simulated distribution is
2. Understand why the Metropolis-Hastings algorithm works
3. Learn how to apply the Metropolis-Hastings algorithm

Most of today's lecture can be found in Chib (2001) and An and Schorfheide (2007)

Bayesian statistics: From last time

- Variance of estimator (frequentist) vs variance of parameter (Bayesian)
- Subjective view of probability
  - Probabilities are statement about our knowledge
- Treating model parameters as random variables thus do not mean that we think that they necessarily vary over time.

Bayesian statistics: From last time

Bayesian procedures can be derived from 4 Bayesian Principles

1. The Likelihood Principle
2. The Sufficiency Principle
3. The Conditionality Principle
4. The Stopping Rule Principle

Main concepts and notation

The main components in Bayesian inference are:

- Data (observables) $Z^T \in \mathbb{R}^{T \times n}$
- A model:
  - Parameters $\theta \in \mathbb{R}^k$
  - A prior distribution $p(\theta) : \mathbb{R}^k \longrightarrow \mathbb{R}^+$
  - Likelihood function $p(Z \mid \theta) : \mathbb{R}^{T \times n} \times \mathbb{R}^k \longrightarrow \mathbb{R}^+$
  - Posterior density $p(\theta \mid Z) : \mathbb{R}^{T \times n} \times \mathbb{R}^k \longrightarrow \mathbb{R}^+$

We need a method to construct the posterior density

The end product of Bayesian statistics

Most of Bayesian econometrics consists of simulating distributions of parameters using numerical methods.

- A simulated posterior is a numerical approximation to the distribution $p(Z \mid \theta)p(\theta)$
- This is useful since the the distribution $p(Z \mid \theta)p(\theta)$ (by Bayes' rule) is proportional to $p(\theta \mid Z)$

$$p(\theta \mid Z) = \frac{p(Z \mid \theta)p(\theta)}{p(Z)}$$

- We rely on ergodicity, i.e. that the moments of the constructed sample correspond to the moments of the distribution $p(\theta \mid Z)$

The most popular (and general) procedure to simulate the posterior is called the Metropolis-Hastings Algorithm

Metropolis-Hastings

Metropolis-Hastings is a way to simulate a sample from a *target distribution*

- In practice, the target distribution will in most cases be the posterior density $p(\theta \mid Z)$ but it doesn't have to be

But what does it mean to sample from a given distribution?

Sampling from a distribution

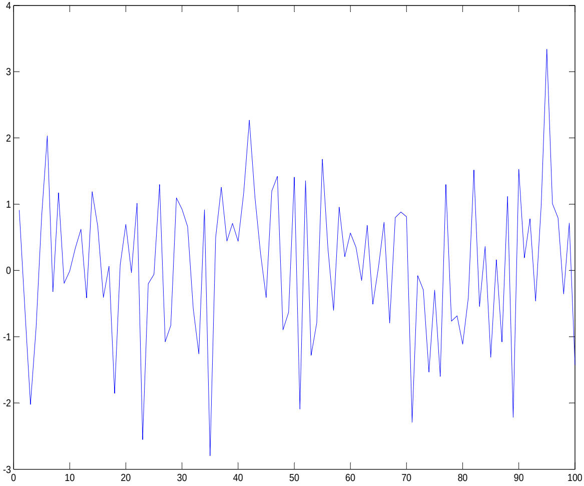
Example:

- Standard Normal distribution $N(0, 1)$

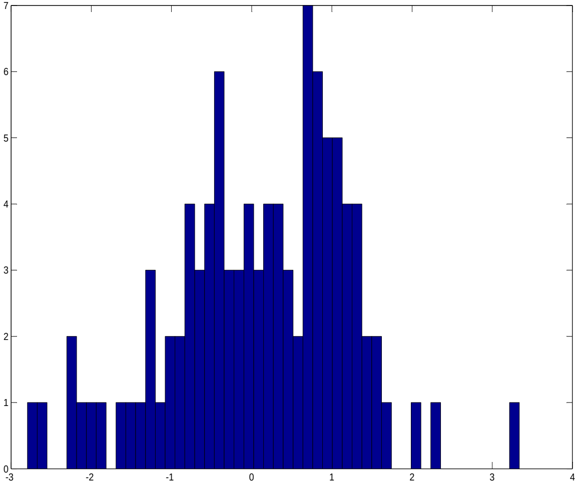
If distribution is ergodic, sample shares all the properties of the true distribution asymptotically

- Why don't we just compute the moments directly?
  - When we can, we should (as in the Normal distribution's case)
  - Not always possible, either because tractability reasons or for computational burden

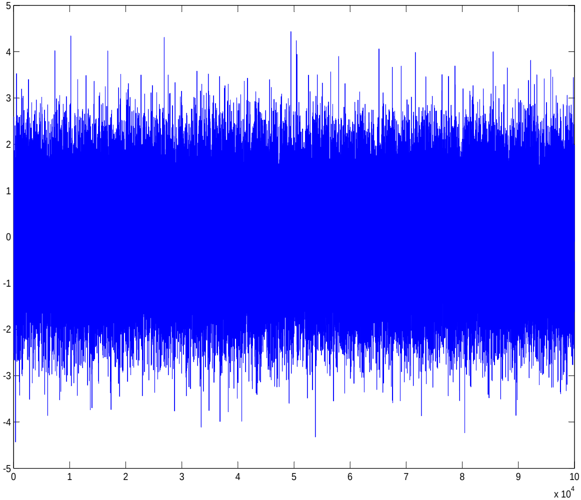
Sample from Standard Normal



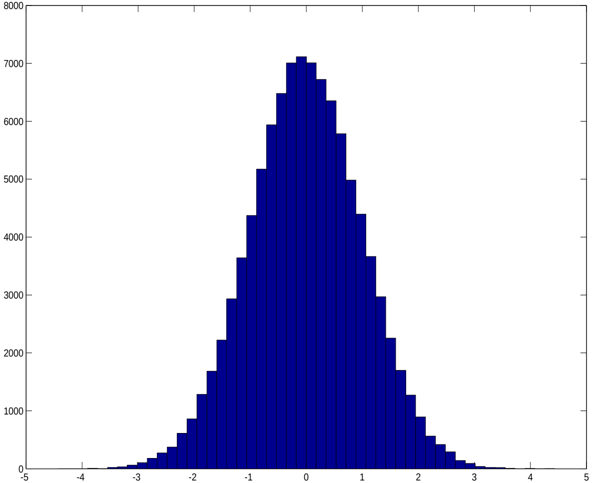
Sample from Standard Normal



Sample from Standard Normal



Sample from Standard Normal



Sampling from a distribution

Example:

- Standard Normal distribution $N(0,1)$

If distribution is ergodic, sample shares all the properties of the true distribution asymptotically

- Why don't we just compute the moments directly?
  - When we can, we should (as in the Normal distribution's case)
  - Not always possible, either because tractability reasons or for computational burden

What is a Markov Chain?

A process for which the distribution of next period variables are independent of the past , once we condition on the current state

- A Markov chain is a stochastic process with the Markov property
- "Markov chain" is sometimes taken to mean only processes with a countably finite number of states
  - Here, the term will be used in the broader sense

Markov Chain Monte Carlo methods provide a way of generating samples that share the properties of the *target density* (i.e. the object of interest)

What is a Markov Chain?

Markov property

$$p(\theta_{j+1} \mid \theta_j) = p(\theta_{j+1} \mid \theta_j, \theta_{j-1}, \theta_{j-2}, ..., \theta_1)$$

Transition density function $p(\theta_{j+1} \mid \theta_j)$ describes the distribution of θ_{j+1} conditional on θ_j.

- In most applications, we know the conditional transition density and can figure out unconditional properties like $E(\theta)$ and $E\left(\theta^2\right)$
- MCMC methods can be used to do the opposite: Determine a particular conditional transition density such that the unconditional distribution converges to that of the *target distribution.*

Let's have a first look at the Metropolis-Hastings Algorithm

The Random-Walk Metropolis Algorithm

1. Start with an arbitrary value θ_0
2. Update from θ_j to $\theta_{j+1} (j = 1, 2, \ldots J)$ by
   2.1 Generate $\theta^* \sim N(\theta_j, \Sigma)$
   2.2 Define
   $$\alpha = \min\left(\frac{p(\theta^*)}{p(\theta_j)}, 1\right) \tag{1}$$
   2.3 Take
   $$\theta_{j+1} = \left\{ \begin{array}{c} \theta^* \text{ with probability } \alpha \\ \theta_j \text{ otherwise} \end{array} \right\}$$
3. Repeat Step 2 J times.

But why does it work?

Simulating distributions using MCMC methods

We can look at a simple discrete state space example:

- θ can take two values $\theta \in \{\theta^L, \theta^H\}$
- Probabilities $p(\theta^L) = 0.4$ and $p(\theta^H) = 0.6$

How can we construct a sequence $\theta_{(1)}, \theta_{(2)}, \theta_{(3)}, ..., \theta_{(J)}$ such that the relative number of occurrences of θ^L and θ^H in the sample correspond to those of the density described by $p(\theta)$?

A two-state Markov Chain for ?

Transition probabilities are defined as

$$\pi_{i,k} = p\left(\theta_{j+1} = \theta^{i} \mid \theta_{j} = \theta^{k}\right) : i, k \in \{L, H\}$$

Unconditional probabilities solves the equation

$$\begin{bmatrix} \pi_L & \pi_H \end{bmatrix} = \begin{bmatrix} \pi_L & \pi_H \end{bmatrix} \begin{bmatrix} \pi_{LL} & \pi_{HL} \\ \pi_{LH} & \pi_{HH} \end{bmatrix}$$

Problem: Define transition probabilities such that unconditional probabilities equal are those of the target distribution

A two-state Markov Chain for ?

We want the chain to spend more time in the more likely state θ^H but not *all* the time.

- If $\theta_j = \theta^L$ then $\theta_{j+1} = \theta^H$
- If $\theta_j = \theta^H$ then $\theta_{j+1} = \theta^L$ with probability α and $\theta_{j+1} = \theta^H$ with probability $(1-\alpha)$

Finding the right α :

$$\left[\begin{array}{cc} \pi_L & \pi_H \end{array}\right] = \left[\begin{array}{cc} \pi_L & \pi_H \end{array}\right] \left[\begin{array}{cc} 0 & 1 \\ \alpha & (1-\alpha) \end{array}\right]$$

We get two equations, $\pi_L = \alpha\pi_H, \quad \pi_H = \pi_L + (1-\alpha)\pi_H,$ solving for α gives $\alpha = \frac{\pi_L}{\pi_H}$

Simulating distributions using MCMC methods

Let's do an example by hand.

Simulating distributions using MCMC methods

Using the ratio of relative probability/density to decide whether to accept a candidate turns out to be extremely general: exactly the same condition ensures that Markov Chain converges to target distribution also for continuous parameter spaces

- Loosely speaking, it is the condition that makes sure that the Markov chain spend "just the right amount of time" at each point in the parameter space
- How the candidate is generated almost doesn't matter at all as long as chain is:
  - Irreducible
  - Aperiodic
- Partly depends on choice that determine how proposal density is generated

Proposal density

Any proposal density with infinite support would do (e.g. normal)

The choice a matter of efficiency. Some options:

- RW-MH
- Adaptive Random Walk
- Independent M-H

The Random-Walk Metropolis Algorithm

1. Start with an arbitrary value θ_0
2. Update from θ_j to $\theta_{j+1}(j = 1, 2, ...J)$ by
    2.1 Generate $\theta^* \sim N(\theta_j, \Sigma)$
    2.2 Define
    $$\alpha = \min\left(\frac{L(Z \mid \theta^*)}{L(Z \mid \theta_j)}, 1\right) \tag{2}$$
    2.3 Take
    $$\theta_{j+1} = \left\{ \begin{array}{c} \theta^* \text{ with probability } \alpha \\ \theta_j \text{ otherwise} \end{array} \right\}$$
3. Repeat Step 2 J times.

We can use this to estimate the likelihood function of a simple DSGE model

A simple DSGE model

$$
\begin{aligned}
x_t &= \rho x_{t-1} + u_t^x \\
y_t &= E_t(y_{t+1}) - \frac{1}{\gamma}\left[r_t - E_t\left(\pi_{t+1}\right)\right] + u_t^y \\
\pi_t &= E_t\left(\pi_{t+1}\right) + \kappa\left[y_t - x_t\right] + u_t^\pi \\
r_t &= \phi_\pi \pi_t
\end{aligned}
$$

A simple DSGE model

Substitute in the interest rate in the Euler equation

$$
\begin{aligned}
x_t &= \rho x_{t-1} + u_t^x \\
y_t &= E_t(y_{t+1}) - \frac{1}{\gamma}\left[\phi_\pi \pi_t - E_t\left(\pi_{t+1}\right)\right] + u_t^y \\
\pi_t &= E_t\left(\pi_{t+1}\right) + \kappa\left[y_t - x_t\right] + u_t^\pi
\end{aligned}
$$

Solving model using method of undetermined coefficients

Conjecture that model can be put in the form

$$\begin{aligned} x_t &= \rho x_{t-1} + u_t^x \\ y_t &= a x_t + u_t^y \\ \pi_t &= b x_t + u_t^\pi \end{aligned}$$

Why is this a good guess?

Solving model using method of undetermined coefficients

Substitute in conjectured form of solution (ignoring the shocks u_t^y and u_t^π for now) into structural equation

$$
\begin{aligned}
ax_t &= a\rho x_t - \frac{1}{\gamma}\left[\phi_\pi b x_t - b\rho x_t\right] \\
bx_t &= b\rho x_t + \kappa\left[a x_t - x_t\right]
\end{aligned}
$$

where we used that $x_t = \rho x_{t-1} + u_t^x$ implies that $\mathrm{E}[x_{t+1} \mid x_t] = \rho x_t$

Solving model using method of undetermined coefficients

Equate coefficients on right and left hand side

$$\begin{aligned} a &= a\rho - \frac{1}{\gamma}\phi_\pi b + \frac{1}{\gamma} b\rho \\ b &= b\rho + \kappa \left[a - 1\right] \end{aligned}$$

or

$$\begin{bmatrix} (1-\rho) & \frac{1}{\gamma}(\phi_\pi - \rho) \\ -\kappa & (1-\rho) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ -\kappa \end{bmatrix}$$

Solving model using method of undetermined coefficients

Solve for a and b

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} (1-\rho) & \frac{1}{\gamma}(\phi_\pi - \rho) \\ -\kappa & (1-\rho) \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ -\kappa \end{bmatrix}$$

or

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} -\kappa \frac{\phi-\rho}{-c} \\ \kappa\gamma \frac{1-\rho}{-c} \end{bmatrix}$$

where $c = \gamma - \kappa\rho - 2\gamma\rho + \kappa\phi + \gamma\rho^2 < 0$

A simple DSGE model

The solved model

$$
\begin{aligned}
x_t &= \rho x_{t-1} + u_t^x \\
y_t &= -\kappa \frac{\rho - \phi_\pi}{c} x_t + u_t^y \\
\pi_t &= \kappa\gamma \frac{\rho - 1}{c} x_t + u_t^\pi
\end{aligned}
$$

where $c = \gamma - \kappa\rho - 2\gamma\rho + \kappa\phi + \gamma\rho^2 < 0$

We want to estimate the distributions of $\theta = \{\rho, \gamma, \kappa, \phi, \sigma_x, \sigma_y, \sigma_\pi, \}$

A simple DSGE model

Put the solved model in state space form

$$\begin{aligned} X_t &= AX_{t-1} + Cu_t \\ Z_t &= DX_t + v_t \end{aligned}$$

where

$$\begin{aligned} X_t &= x_t, A = \rho, Cu_t = u_t^x \\ Z_t &= \begin{bmatrix} y_t \\ \pi_t \end{bmatrix}, D = \begin{bmatrix} -\kappa\frac{\phi_\pi - \rho}{-c} \\ \kappa\gamma\frac{1-\rho}{-c} \end{bmatrix}, v_t = \begin{bmatrix} u_t^y \\ u_t^\pi \end{bmatrix} \end{aligned}$$

The log likelihood function of a state space system

For a given state space system

$$\begin{aligned} X_t &= AX_{t-1} + C\mathbf{u}_t \\ \underset{(p\times 1)}{Z_t} &= DX_t + \mathbf{v}_t \end{aligned}$$

we can evaluate the log likelihood by computing

$$\mathcal{L}(Z \mid \Theta) = -.5 \sum_{t=0}^{T} \left[p \ln(2\pi) + \ln|\Omega_t| + \widetilde{Z}_t' \Omega_t^{-1} \widetilde{Z}_t \right]$$

where $\widetilde{Z}_t$ are the innovation from the Kalman filter

The parameter vector θ^0 and the variance of random walk innovations in MCMC Σ

Parameterize the model according to

$$\begin{aligned} \theta &= \{\rho, \gamma, \kappa, \phi, \sigma_x, \sigma_y, \sigma_\pi, \} \\ &= \{0.9, 2, 0.1, 1.5, 1, 1, 1\} \end{aligned}$$

- Generate data from true model and $T = 100$
- Set starting value $\theta^{(0)} = \theta$ (OK, this option is not available in practice.)
- Set covariance matrix of random walk increments in Metropolis Algorithm proportional to absolute values of true parameters

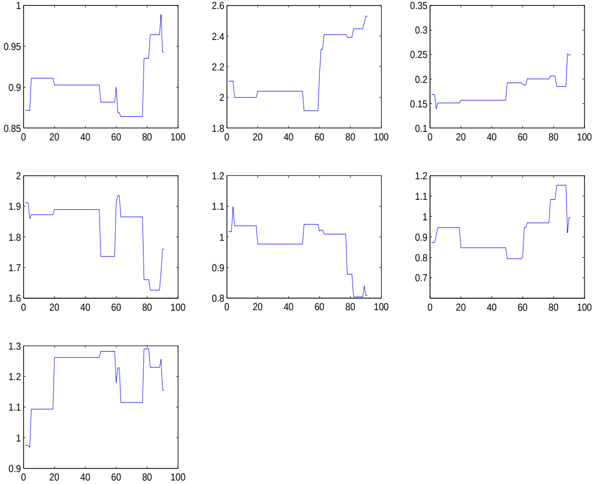
$$\Sigma = \varepsilon \times diag(abs(\theta))$$

The Random-Walk Metropolis Algorithm

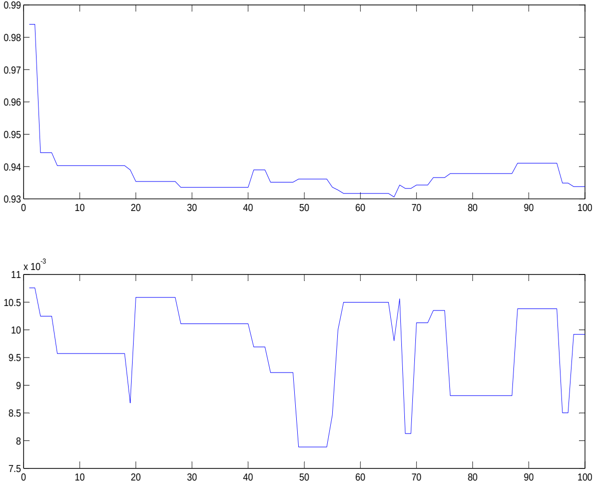
We now have all we need:

1. Start with an arbitrary value θ_0
2. Update from θ_j to $\theta_{j+1} (j = 1, 2, \ldots J)$ by
   2.1 Generate $\theta^* \sim N(\theta_j, \Sigma)$
   2.2 Define
   $$\alpha = \min\left(\frac{L(Z \mid \theta^*)}{L(Z \mid \theta_j)}, 1\right) \tag{3}$$
   2.3 Take
   $$\theta_{j+1} = \left\{ \begin{array}{c} \theta^* \text{ with probability } \alpha \\ \theta_j \text{ otherwise} \end{array} \right\}$$
3. Repeat Step 2 J times

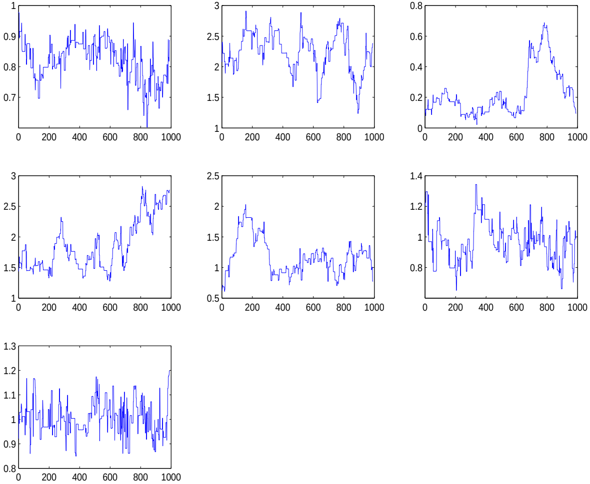
The MCMC with J=100



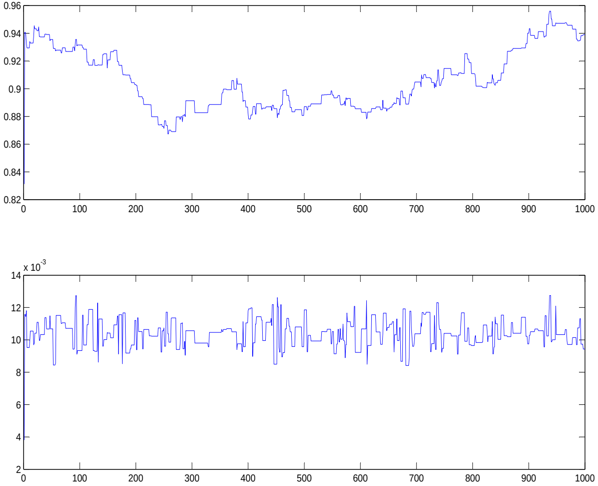
The MCMC with J=100



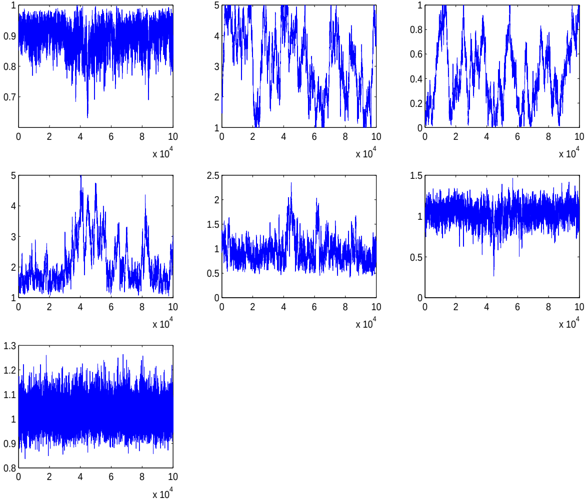
The MCMC with J=1000



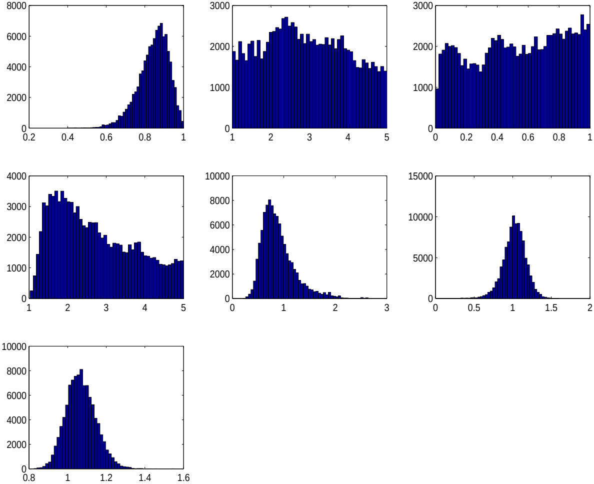
The MCMC with J=1000



The MCMC with J=100000



The MCMC with J=100000

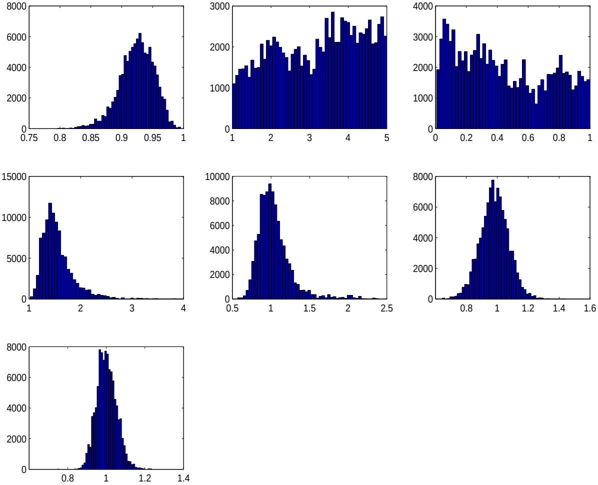


Identification

Some posterior distribution look ‘flat‘

- Can be a short sample issue
  - Nothing to do if we are estimating a model on real data, but here we can check
- Can be a problem with mapping between parameters and likelihood

The MCMC with T=200 and J=100000



Identification

Posterior distribution for γ and κ kappa still look 'flat' probably a true identification issue

- Little can be said about identification a priori

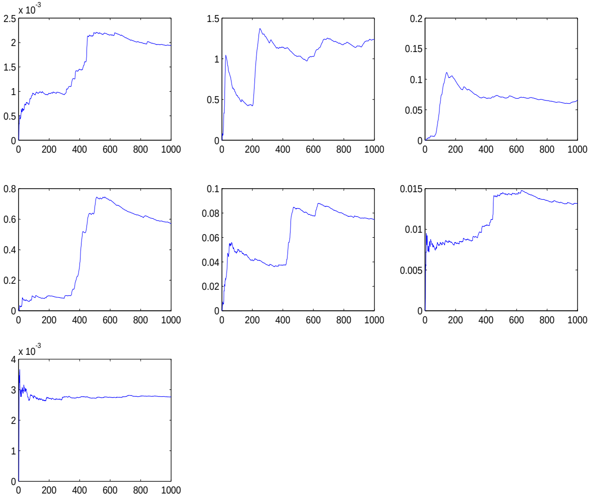
Convergence

How many draws do we need?

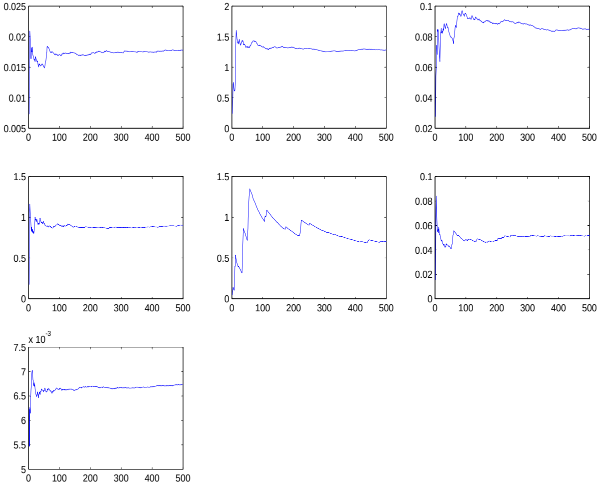
- Optimal J increases with number of parameters
- One informal check is to plot the diagonal of the recursive covariance matrix of the MCMC

$$\frac{1}{j}\sum_{i=0}^{j}\theta^{(i)}\theta'^{(i)} \text{ for } j = 1, 2, ...J$$

Checking for convergence



Checking for convergence J=500000



Combining prior and sample information

Sometimes we know more about the parameters than what the data tells us, i.e. we have some prior information.

What do we mean by prior information?

- For a DSGE model, we may have information about "deep" parameters
  - Range of some parameters restricted by theory, e.g. risk aversion should be positive
  - Discount rate is inverse of average real interest rates
  - Price stickiness can be measured by surveys
- We may know something about the mean of a process

How do we combine prior and sample information?

Bayes' theorem:

$$
\begin{aligned}
P(\theta \mid Z)\, P(Z) &= P(Z \mid \theta)\, P(\theta) \\
&\Leftrightarrow \\
P(\theta \mid Z) &= \frac{P(Z \mid \theta)\, P(\theta)}{P(Z)}
\end{aligned}
$$

- Since $P(Z)$ is a constant, we can use $P(Z \mid \theta)\, P(\theta)$ as the posterior likelihood (a likelihood function is any function that is proportional to the probability).

We now need to choose $P(\theta)$

Choosing prior distributions

The beta distribution is a good choice when parameter is in [0,1]

$$P(x) = \frac{(1-x)^{b-1}\, x^{a-1}}{B(a, b)}$$

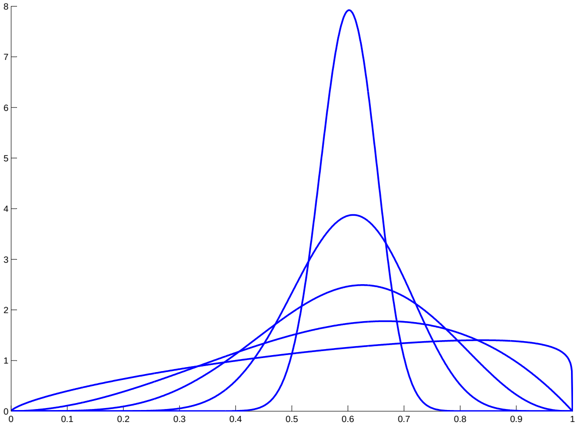
where

$$B(a, b) = \frac{(a-1)!(b-1)!}{(a+b-1)!}$$

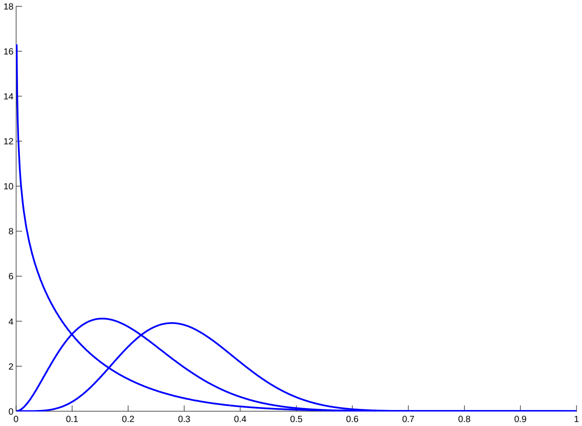
Easier to parameterize using expression for mean, mode and variance:

$$\begin{aligned} \mu &= \frac{a}{a+b}, \quad \widehat{x} = \frac{a-1}{a+b-2} \\ \sigma^2 &= \frac{ab}{(a+b)^2\,(a+b+1)} \end{aligned}$$

Examples of beta distributions holding mean fixed



Examples of beta distributions holding s.d. fixed



Choosing prior distributions

The inverse gamma distribution is a good choice when parameter is positive

$$P(x) = \frac{b^a}{\Gamma(a)}(1/x)^{a+1}\exp(-b/x)$$

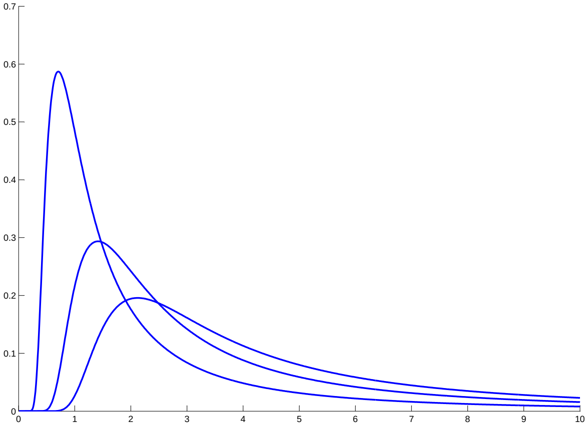
where

$$\Gamma(a) = (a-1)!$$

Again, easier to parameterize using expression for mean, mode and variance:

$$\begin{aligned} \mu &= \frac{b}{a-1}; a > 1, \quad \widehat{x} = \frac{b}{a+1} \\ \sigma^2 &= \frac{b^2}{(a-1)^2\,(a-2)}; a > 2 \end{aligned}$$

Examples of inverse gamma distributions



The Random-Walk Metropolis Algorithm with priors

1. Start with an arbitrary value θ_0
2. Update from θ_j to $\theta_{j+1} (j = 1, 2, \ldots J)$ by
   2.1 Generate $\theta^* \sim N(\theta_j, \Sigma)$
   2.2 Define
   $$\alpha = \min\left(\frac{L(Z \mid \theta^*) P(\theta^*)}{L(Z \mid \theta_j) P(\theta_j)}, 1\right) \tag{4}$$
   2.3 Take
   $$\theta_{j+1} = \left\{ \begin{array}{c} \theta^* \text{ with probability } \alpha \\ \theta_j \text{ otherwise} \end{array} \right\}$$

The only difference compared to before is that the priors appear in the ratio in (3).

Practical implementation

Since we compute log-likelihood we can use the log of the likelihood ratio in the Random Walk Metropolis Algorithm

$$\begin{aligned} \ln \alpha &= \min [(\ln L(Z \mid \theta^*) + \ln P(\theta^*) \\ &\quad - \ln L(Z \mid \theta_j) - \ln P(\theta_j)), 0] \end{aligned}$$

If priors across parameters are independent we have that $\ln P(\theta_j) = \ln P(\theta_{1,j}) + \ln P(\theta_{2,j}) + ... + \ln P(\theta_{q,j})$ where

$$\theta_j = \begin{bmatrix} \theta_{1,j} & \theta_{2,j} & \cdots & \theta_{q,j} \end{bmatrix}'$$

Let's get deep with an old friend:

$$
\begin{aligned}
x_t &= \rho x_{t-1} + u_t^x \\
y_t &= E_t(y_{t+1}) - \frac{1}{\gamma}\left[r_t - E_t\left(\pi_{t+1}\right)\right] + u_t^y \\
\pi_t &= E_t\left(\pi_{t+1}\right) + \kappa\left[y_t - x_t\right] + u_t^\pi \\
r_t &= \phi_\pi \pi_t
\end{aligned}
$$

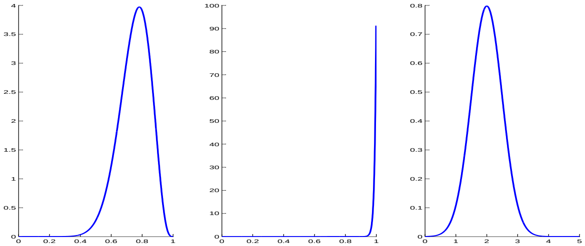
The parameter κ is in the benchmark 3-equation NK model given by

$$\kappa = \frac{(1-\delta)(1-\delta\beta)}{\delta}$$

where δ is the Calvo parameter of price stickiness and β is the discount factor. We now have a new parameter vector $\theta = \{\rho, \gamma, \delta, \beta, \phi, \sigma_x, \sigma_y, \sigma_\pi, \}$

The priors

- The prior on relative risk aversion γ is truncated Normal with mean 2 and s.d. 0.5.
- The prior on the discount factor β is Beta with mean 0.99 and s.d. 0.01
- The prior on the Calvo parameter δ is Beta with mean 0.75 and s.d. 0.1



The Random-Walk Metropolis Algorithm with priors

1. Start with an arbitrary value θ_0
2. Update from θ_j to $\theta_{j+1}(j = 1, 2, ...J)$ by
   2.1 Generate $\theta^* \sim N(\theta_j, \Sigma)$
   2.2 Define
   $$\alpha = \min\left(\frac{L(Z \mid \theta^*)\, P(\theta^*)}{L(Z \mid \theta_j)\, P(\theta_j)}, 1\right) \tag{5}$$
   2.3 Take
   $$\theta_{j+1} = \left\{ \begin{array}{c} \theta^* \text{ with probability } \alpha \\ \theta_j \text{ otherwise} \end{array} \right\}$$
3. Repeat Step 2 J times

The log prior

The log prior is given by

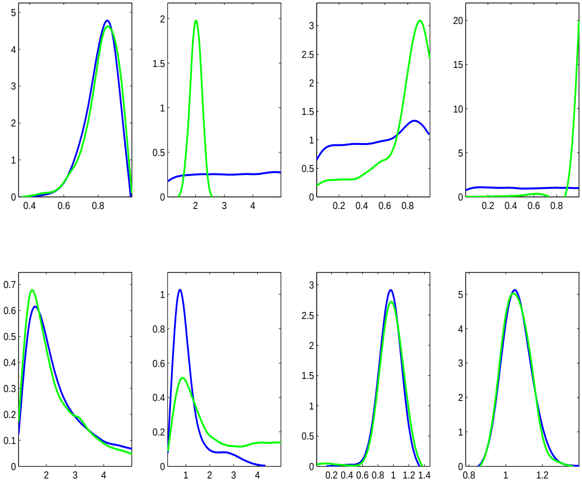
$$\begin{aligned} \ln P(\theta_j) &= \ln P(\theta_{1,j}) + \ln P(\theta_{2,j}) + \ldots + \ln P(\theta_{q,j}) \\ &= \ln P(\gamma_j) + \ln P(\delta_j) + \ln P(\beta_j) \end{aligned}$$

since we can ignore the (constant) probabilities on the uniform priors.
We then have that

$$\begin{aligned} \ln\left[L\left(Z \mid \theta^*\right) P(\theta^*)\right] &= \ln L(Z \mid \theta) + \ln P(\theta_j) \\ &\Longrightarrow \\ \alpha &= \min\left(\frac{\exp\left[\ln L(Z \mid \theta^*) + \ln P(\theta^*)\right]}{\exp\left[\ln L(Z \mid \theta_j) + \ln P(\theta_j)\right]}, 1\right) \end{aligned}$$

i.e. all we need for the RWMA

Posterior with uniform and informative priors



Inference about parameters

Compute sample equivalent:

- Central tendencies: Mean, mode
- Variance

Compute frequency of occurrence:

- How likely is it that θ^1 and θ^2 are both smaller than 0?

That's it for today.