EC 607: Bayesian Econometrics

Bayesian Computation I

Prof. Jeremy M. Piger

Department of Economics

University of Oregon

Last Revised: March 22, 2019

1 Bayesian Econometrics in Practice

- The examples given previously were all cases where Bayesian objects of interest, such as the parameter posterior distribution, marginal likelihood, or posterior predictive density were known distributions that could be computed analytically.
- However, these are very special cases that correspond to particular combinations of likelihoods and priors. Deviations from these cases often result in Bayesian objects of interest that can't be computed analytically.
- This fact limited the applied usefulness of Bayesian methods for many years.
- The modern approach to Bayesian econometrics when we can't characterize a distribution analytically is to sample the unknown distribution. Then any object of interest from the distribution can be estimated from these samples. If we are able to sample the distribution a lot, these estimates will be very accurate.
- This option has only become available with modern computing power. The revolution in computing power has also revolutionized the implementation of Bayesian econometrics.

2 Sampling Distributions of Interest: Introduction

- The bulk of our discussion going forward in this class will be about obtaining and using random samples of some vector of random variables X from some probability distribution $p(X)$. Before beginning, we describe five useful results about random sampling.

    1. Suppose we are interested in obtaining a random sample of X_i, which is the i^{th} element of X, from the marginal distribution $p(X_i)$. We can do this by obtaining

a random sample of X from the joint distribution $p(X)$. Then the i^{th} element of this draw will be a random sample from $p(X_i)$.

2. Partition X into two vectors, denoted X_i and X_j, where $X = \left(X_i', X_j'\right)'$. The law of total probability tells us that $p(X_i, X_j) = p(X_i|X_j)\, p(X_j)$. Suppose one obtains a random sample of X_j from the marginal distribution $p(X_j)$, and denote this random sample as $\widetilde{X_j}$. Then, suppose we obtain a random sample of X_i from the conditional distribution $p\left(X_i|\widetilde{X_j}\right)$, and denote this random sample as $\widetilde{X_i}$. Then $\widetilde{X} = \left(\widetilde{X}_i', \widetilde{X}_j'\right)'$ is a random sample from the joint distribution $p(X_i, X_j)$ and, given result #1 above, X_i and X_j are valid random samples from $p(X_i)$ and $p(X_j)$.

3. Suppose we obtain a random sample of X from $p(X)$. Denote this random sample as $\widetilde{X}$. We can then obtain a random sample from any deterministic function of X, $h(X)$, by computing $h\left(\widetilde{X}\right)$.

4. The previous point motivates a procedure known as **Monte Carlo Integration**. Specifically, suppose we wish to compute $E(h(X))$, where $h(X)$ is a deterministic function of X, and X is a random variable arising from $p(X)$. Let $X^{[g]}$ be a random sample from $p(X)$, for $g = 1, 2, \cdots, G$. Then:

$$\lim_{G\to\infty} \frac{1}{G} \sum_{g=1}^{G} h\left(X^g\right) = E(h(X)) \tag{1}$$

5. Suppose we want to sample a truncated distribution:

$$p(\theta) \propto I(\theta)\, g(\theta)$$

where $g(\theta)$ is a probability density function and $I(\theta)$ is an indicator function that is 1 if θ is inside of an acceptable region and 0 if it is not. A simple example would be a truncated normal distribution. A more complicated example would be a vector of AR parameters that are constrained such that the AR process is

covariance stationary.

Suppose we are able to obtain a random draw from $g(\theta)$. In this case, a general approach to obtain a random draw from the truncated distribution is via **rejection sampling**:

(a) Draw θ^* from $g(\theta)$.

(b) If $I(\theta^*) = 1$ then accept the draw of θ^*.

(c) If $I(\theta^*) = 0$ then reject the draw of θ^* and go to step (a).

- Recall, the three main Bayesian probability distributions of interest are the parameter posterior density $p(\theta|Y)$, the marginal likelihood $p(Y)$, and the posterior predictive density $p(y^*|Y)$.

- Let us begin by focusing on the posterior density $p(\theta|Y)$. The idea behind simulation is to obtain random samples from $p(\theta|Y)$ that can then be used to estimate any feature of the distribution. For most cases we care about, a law of large numbers will apply that implies convergence of the sample estimate to the population counterpart. Thus, the accuracy of this estimation can be made arbitrarily high simply by increasing the number of random samples taken.

- We will typically refer to the number of random samples taken as the "number of replications" rather than the "sample size" to avoid confusion with the sample size of Y. However, the number of replications does play the role of a traditional sample size for estimation of the relevant feature of the posterior density.

- For example, suppose we wish to obtain the mean of the posterior distribution of an element of θ denoted θ_i. This posterior is denoted $p(\theta_i|Y)$. We could obtain G samples of θ from $p(\theta|Y)$, denoted $\theta^{[g]}$, $g = 1, \cdots, G$, and then form the sample mean:

$$\frac{1}{G}\sum_{g=1}^{G}\theta_i^{[g]}$$

This will produce an arbitrarily accurate estimate of $E(\theta_i|Y)$ as G is made larger and larger. Implicit in this procedure is result #1 above.

- As another example, say you are interested in computing $\Pr[\theta_i|Y] > a$. One can estimate this by sampling θ_i G times from $p(\theta|Y)$, and then computing the proportion of the G samples for which $\theta_i > a$. This is an example of Monte Carlo integration.

- We could also figure out the distribution itself with an arbitrary degree of accuracy. For a discrete distribution we can just compute the proportion of a large number of draws that fall on each of the discrete values. For a continuous distribution, we could use a kernel density estimator applied to lots of draws. In Matlab, the command ksdensity() will plot an estimate of the density that will be very accurate with a large number of draws.

- Note that we could, given samples from the posterior distribution of θ, also simulate samples from the posterior predictive distribution. Suppose we have G draws of θ from $p(\theta|Y)$. We can then simulate G draws from the posterior predictive distribution, $p(y^*|Y)$, by simulating y^* from $p\left(y^*|\theta^{[g]}, Y\right)$, $g = 1, 2, \cdots, G$. Implicit in this procedure is result #2 above.

- There are also approaches we can use to estimate the marginal likelihood via simulation. We will discuss these in detail later.

3 Markov-Chain Monte Carlo Methods

3.1 Introduction

- We have seen that if we can obtain a large number of samples from the posterior distribution $p(\theta|Y)$ then we can effectively implement the Bayesian approach to econometrics, even if we can't analytically characterize objects of interest.

- The remaining question is how we can obtain such samples? There are a large number of techniques in the Bayesian literature, with many that are effective in low dimensional problems.

- By far the most popular approaches in use today, particularly in moderate to high dimensional problems, are examples of what is known as **Markov-Chain Monte Carlo (MCMC or MC^2)** methods.

- These methods are powerful enough that there are few, if any, distributions arising in econometric practice that can't be sampled using MCMC.

- The basic idea behind MCMC is as follows. Suppose we wish to draw random samples from a distribution, known as the target distribution. The MCMC approach constructs a stochastic process such that if we were to draw realizations of random variables from the stochastic process for a long time, then eventually these draws would come from a stationary distribution that is the target distribution. We can then let the stochastic process run, and once it converges to the target distribution, we can start collecting realizations of the stochastic process as draws from the target distribution.

- In the MCMC approach, the stochastic process we construct is a **markov chain**. Before we get to MCMC, we will begin by reviewing some basic theory about markov chains.

3.2 Markov Chain Basics

- A **Markov chain** is a random sequence of (continuous or discreet) random variables $(X_0, X_1, X_2, \cdots)$ that have the **Markov property**, meaning the probability density of X_t, given all preceding realizations, depends at most on the immediately preceding realization:

$$\Pr\left(X_{t+1} \in A | X_t, X_{t-1}, \cdots, X_0\right) = \Pr\left(X_{t+1} \in A | X_t\right)$$

where A is a subset of χ, the sample space for X_{t+1}.

- The right hand side of this equation is called a transition probability. The value taken by X_t is called the **state** of the Markov chain at t. The chain is **homogenous** if the transition probabilities do not depend on the date, t.

- In Bayesian applications of MCMC the state of the chain X_t is a vector random variable, usually representing a parameter vector. Its value is a value for each component of X_t.

- A homogenous Markov chain can be fully described by its initial state and the rule describing how the chain moves from its state at t to its state at $t+1$. This rule is described by the **transition kernel**, which is a function $K\left(X_t, X_{t+1}\right)$ that for each X_t provides a probability density for X_{t+1}. For example, for a Markov chain defined over a discrete sample space, the transition kernel is given by the pmf:

$$K\left(X_t, X_{t+1}\right) = \Pr\left(X_{t+1} | X_t\right)$$

For a continuous sample space the transition kernel is written as a pdf:

$$K\left(X_t, X_{t+1}\right) = p\left(X_{t+1} | X_t\right)$$

- As we are usually dealing with continuous state spaces, we will write most formulas going forward using the pdf notation.

- The probability density function of X_{t+1} is given by:

$$p_{t+1}\left(X_{t+1}\right) = \int_{X_t \in \chi} K\left(X_t, X_{t+1}\right) p_t\left(X_t\right) dX_t,$$

- A **stationary distribution** for the Markov chain is a solution to the vector of functional equations:

$$p\left(X_{t+1}\right)=\int_{X_t\in\chi}K\left(X_t,X_{t+1}\right)p\left(X_t\right)dX_t$$

- A key element of interest in Markov chain theory is existence of a unique stationary distribution for a Markov chain. Here we will give the conditions required for a Markov-chain with a countably infinite state space to have a unique stationary distribution. Versions of these conditions are available for a continuous state space:

  1. **Irreducibility:** It must be possible to get from any state to any other state. Thus, every state will be visited if the chain runs long enough, regardless of where the chain starts.
  2. **Positive Recurrence:** The mean time to return to a state is finite. Thus, every state will be visited infinitely often if the chain runs long enough.

- We are also interested in whether a Markov chain converges to its unique stationary distribution. Convergence implies that regardless of the initial distribution, $p_0\left(X_0\right)$, $p_t\left(x_t\right)\rightarrow p\left(x_t\right)$ as $t\rightarrow\infty$.

- A countably infinite Markov chain with a unique stationary distribution will converge to this stationary distribution irrespective of the initial distribution p_0 provided that the chain is **aperiodic**. An aperiodic chain rules out a transition kernel that implies periodic behavior. More formally, it requires that the the probability of a state n periods in the future, conditional on being in that state today, is not zero once n crosses some suitably large threshold. There is a version of this convergence requirement for Markov chains with continuous state spaces.

- Finally, it can be shown that a Markov chain that satisfies properties such that it converges to a unique stationary distribution will also satisfy an **ergodic theorem**,

which justifies the use of post-convergence realizations of X_t to estimate objects of interest about the target distribution. In particular, the ergodic theorem justifies Monte Carlo Integration using post-convergence realizations of X_t.

- Putting it all together, suppose we have a Bayesian pdf of interest, such as a posterior density function. With MCMC techniques, we will simulate realizations of X_t from a Markov chain where the chain converges to a unique stationary distribution that is the posterior density function. Once this convergence has occurred, the simulations from X_t will be from the posterior density function. The ergodic theorem then implies that we can use these draws to estimate objects of interest about the posterior density, such as the posterior mean. These estimates can be made arbitrarily good by increasing the number of draws taken from the chain.

- The remaining difficulty is then to design a Markov chain that converges to the target density you have in mind. We turn to this next.

4 Metropolis-Hastings Sampler

4.1 Method

- The Metropolis-Hastings (MH) Sampler is an MCMC technique that is very general. In principle, it can be applied for **any** model for which we can evaluate the likelihood function.

- Suppose we have a target distribution of interest that we want to take samples from. To fix ideas, suppose it is a posterior pdf $p(\theta|Y)$, where θ may be either a scalar or a vector. We will be interested in designing an MCMC technique that produces a sequence of draws of θ, where the $g^{[th]}$ draw is denoted $\theta^{[g]}$.

- We will start by describing the implementation of the MH sampler. Then we will map

this implementation into a transition kernel and show that this transition kernel has stationary density given by $p(\theta|Y)$.

- The MH sampler is implemented through the following algorithm:

  1. Generate a proposed value of $\theta^{[g+1]}$, called θ^*, from a **proposal distribution** $q\left(\theta|\theta^{[g]}\right)$. Note that the proposal density is potentially a conditional density, where the conditioning is on the previous drawn value, $\theta^{[g]}$.
  2. Compute the **acceptance probability**:

$$\alpha\left(\theta^{[g]}, \theta^*\right) = \min\left(\frac{p\left(\theta^*|Y\right) q\left(\theta^{[g]}|\theta^*\right)}{p\left(\theta^{[g]}|Y\right) q\left(\theta^*|\theta^{[g]}\right)}, 1\right)$$

  3. Set $\theta^{[g+1]} = \theta^*$ with probability α and $\theta^{[g+1]} = \theta^{[g]}$ with probability $1 - \alpha$.
  4. Increment g and go to 1.

  The sampler is initiated with an arbitrary initial value, $\theta^{[0]}$.

- Note that the acceptance probability depends on the posterior distribution, which is what is unknown and we are trying to sample. Thus, at first glance it may appear that we can't implement this algorithm. However, note that the posterior distribution enters twice, once in the denominator and once in the numerator. Thus, we must only be able to evaluate the ratio:

$$\frac{p\left(\theta^*|Y\right)}{p\left(\theta^{[g]}|Y\right)} = \frac{\frac{p(Y|\theta^*)p(\theta^*)}{p(Y)}}{\frac{p\left(Y|\theta^{[g]}\right)p\left(\theta^{[g]}\right)}{p(Y)}} = \frac{p\left(Y|\theta^*\right)p\left(\theta^*\right)}{p\left(Y|\theta^{[g]}\right)p\left(\theta^{[g]}\right)}$$

  and the acceptance probability becomes:

$$\alpha\left(\theta^{[g]}, \theta^*\right) = \min\left(\frac{p\left(Y|\theta^*\right)p\left(\theta^*\right)q\left(\theta^{[g]}|\theta^*\right)}{p\left(Y|\theta^{[g]}\right)p\left(\theta^{[g]}\right)q\left(\theta^*|\theta^{[g]}\right)}, 1\right)$$

Assuming we can evaluate the parts of the likelihood function $p(Y|\theta^*)$ and the prior $p(\theta)$ that depend on θ we are able to evaluate this ratio. In other words, all we need to be able to evaluate is the kernel of the posterior.

- What is the transition kernel that corresponds to this algorithm? The transition kernel is given by:

$$K\left(\theta^{[g]}, \theta^{[g+1]}\right) = \alpha\left(\theta^{[g]}, \theta^{[g+1]}\right) q\left(\theta^{[g+1]}|\theta^{[g]}\right) + \left(1 - r\left(\theta^{[g]}\right)\right) I_{\left(\theta^{[g]}\right)}\left(\theta^{[g+1]}\right)$$

where $I_{\theta^{[g]}}\left(\theta^{[g+1]}\right)$ as an indicator function that is 1 if $\theta^{[g]} = \theta^{[g+1]}$ and is 0 otherwise and:

$$r\left(\theta^{[g]}\right) = \int \alpha\left(\theta^{[g]}, \theta^{[g+1]}\right) q\left(\theta^{[g+1]}|\theta^{[g]}\right) d\theta^{[g+1]}$$

This transition kernel can be loosely interpreted as follows. The first product on the right hand side is the probability that a $\theta^{[g+1]}$ is proposed and accepted, conditional on the chain being at $\theta^{[g]}$. The remainder of this term assigns extra probability to the possibility that $\theta^{[g+1]} = \theta^{[g]}$. Here, $r\left(\theta^{[g]}\right)$ gives the probability that the proposal, given $\theta^{[g]}$, is accepted, so that $1 - r\left(\theta^{[g]}\right)$ is the probability that the proposal isn't accepted. In this case, $\theta^{[g+1]} = \theta^{[g]}$, so this extra probability is given to this case when we evaluate the transition kernel at $\theta^{[g+1]} = \theta^{[g]}$.

It is easy to see that this transition kernel is a valid pdf for $\theta^{[g+1]}$ conditional on $\theta^{[g]}$ as it integrates to one over $\theta^{[g+1]}$ conditional on $\theta^{[g]}$.

- Does the MH transition kernel have a stationary distribution equal to $p(\theta|Y)$? Consider the following result:

A transition kernel is said to be **reversible** or satisfy **detailed balance** if the following is true:

$$g\left(\theta^{[g]}\right) K\left(\theta^{[g]}, \theta^{[g+1]}\right) = g\left(\theta^{[g+1]}\right) K\left(\theta^{[g+1]}, \theta^{[g]}\right)$$

for some pdf $g(\cdot)$ and all $\theta^{[g]}$, $\theta^{[g+1]}$. If a transition kernel is reversible, then $g(\cdot)$ is a stationary distribution of the Markov chain.

Proof: To be a stationary distribution of the chain we require:

$$g\left(\theta^{[g+1]}\right) = \int_{\theta^{[g]}} K\left(\theta^{[g]}, \theta^{[g+1]}\right) g\left(\theta^{[g]}\right) d\theta^{[g]}$$

Is this true?

$$\begin{aligned} g\left(\theta^{[g+1]}\right) &= \int_{\theta^{[g]}} K\left(\theta^{[g+1]}, \theta^{[g]}\right) g\left(\theta^{[g+1]}\right) d\theta^{[g]} \\ g\left(\theta^{[g+1]}\right) &= g\left(\theta^{[g+1]}\right) \int_{\theta^{[g]}} K\left(\theta^{[g+1]}, \theta^{[g]}\right) d\theta^{[g]} \\ g\left(\theta^{[g+1]}\right) &= g\left(\theta^{[g+1]}\right) \end{aligned}$$

The first line is justified by detailed balance. The third line is justified since:

$$\int_{\theta^{[g]}} K\left(\theta^{[g+1]}, \theta^{[g]}\right) d\theta^{[g]} = 1$$

- Simple algebra demonstrates that the MH transition kernel is reversible with $g(\cdot) = p(\theta|Y)$. Thus, $p(\theta|Y)$ is a stationary distribution of the chain.
- Robert and Casella (1999, *Monte Carlo Statistical Methods*) give conditions necessary for convergence. These conditions will hold for the vast majority of econometric models and versions of the MH algorithm that we will encounter. Indeed, they are almost never checked in practice.
- In principle, the proposal density, $q\left(\theta|\theta^{[g]}\right)$ can be any probability density function. In practice, nearly all applications of the MH sampler use one of two approaches:

1. **Independence Proposal**: $q\left(\theta^*|\theta^{[g]}\right) = q\left(\theta^*\right)$. Such a proposal doesn't depend on where the chain is now, $\theta^{[g]}$. A common choice for $q(\cdot)$ is the t-distribution.

2. **Random Walk Proposal**: $q\left(\theta^*|\theta^{[g]}\right) = q\left(|\theta^* - \theta^{[g]}|\right)$. Such a proposal depends only on the distance between where the chain is now, $\theta^{[g]}$, and the proposed value, θ^*. Such a proposal can be alternatively written as:

$$\theta^* = \theta^{[g+1]} + v$$

where v is a vector random variable with mean zero that is distributed i.i.d. with symmetric distribution $q(v)$. This is where this type of proposal gets the name random walk. A popular choice for $q(\cdot)$ is a multivariate normal distribution, $v \sim N(0, R)$.

Note that for the random walk proposal, $q\left(\theta^*|\theta^{[g]}\right) = q\left(\theta^{[g]}|\theta^*\right)$ and thus the acceptance probability simplifies to:

$$\alpha\left(\theta^{[g]}, \theta^*\right) = \min\left(\frac{p\left(Y|\theta^*\right)p\left(\theta^*\right)}{p\left(Y|\theta^{[g]}\right)p\left(\theta^{[g]}\right)}, 1\right)$$

- What makes a good proposal? Technically, any proposal will do if we could have our chain run infinitely. However, some proposal distributions will be more efficient than others, in that they will adequately sample the target distribution more quickly. Define the **acceptance rate** as the proportion of the proposals that are accepted. From the acceptance probability, the acceptance rate will be determined by the relative posterior likelihood of proposals vs. where you are now. For an efficient sampler, the rule of thumb is to have an acceptance rate that is not too high and not too low. If the acceptance rate is very low then this means that very few proposals are being accepted, so your sampler is not exploring the distribution very quickly. It usually indicates that you are generating proposals that are too dispersed, and lie out in the tails of the posterior. If the acceptance rate is very high, this could mean that your proposed value

is very close to your current value, which makes the acceptance probability near one. Again, this will be a case where you will cover the distribution very slowly. Although there is no general optimal acceptance rate, a general rule of is to have an acceptance rate that falls somewhere between 20% and 50%, although moderate deviations from this are unlikely to make a huge difference.

- While a very high acceptance probability is usually considered to be a bad thing, it is worth noting that this doesn't have to be true. As an extreme example, consider a case where the proposal density function is the posterior density function, $p(\theta|Y)$. In this case, we would want to accept every draw from the proposal density function, since these draws are draws from the target density function. If we look at the MH acceptance probability, this is exactly what would happen:

$$\alpha\left(\theta^{[g]}, \theta^{*}\right) = \min\left(\frac{p\left(\theta^{*}|Y\right) p\left(\theta^{[g]}|Y\right)}{p\left(\theta^{[g]}|Y\right) p\left(\theta^{*}|Y\right)}, 1\right) = 1$$

  Thus, a high acceptance probability could simply mean that your proposal density is very close to your target density. However, in order to be safe this is not usually how a high acceptance probability is interpreted.

- How can we calibrate the proposal density? Here we will discuss one popular approach based on a **Large Sample Approximation**:

- It can be shown that as $n \to \infty$, the posterior density converges to a Gaussian limiting distribution with mean equal to $\widehat{\theta}$, the posterior mode, and variance-covariance matrix equal to the inverse of the negative Hessian matrix for the posterior distribution evaluated at the mode. This motivates the approximation:

$$p\left(\theta|Y\right) \approx (2\pi)^{-\frac{k}{2}} |H|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}\left(\theta - \widehat{\theta}\right)' H^{-1}\left(\theta - \widehat{\theta}\right)\right]$$

where $H = \left[- \left[\frac{d^2}{d\theta^2} \log p\left(\theta | Y\right) \right]_{\theta = \widehat{\theta}} \right]^{-1}$.

Note that both $\widehat{\theta}$ and H can be computed given only the kernel of the posterior.

- We might then use $\widehat{\theta}$ and H as the location and scale parameters of a t-distribution for the independence proposal. The additional degrees of freedom parameter can be calibrated to yield an acceptable acceptance rate.

- We might also use H to calibrate the variance-covariance matrix of the variance v in the random walk proposal. This would usually be implemented by setting $R = cH$, where c is a scalar parameter that can be scaled to yield an acceptable acceptance rate.

- It is important to note that this approach does not rely on this large sample approximation to describe the posterior distribution. It simply uses it to scale the proposal efficiently.

- The presentation of the MH algorithm above sampled all of the elements of θ at the same time. It is sometimes convenient to separate your parameters into multiple "blocks", and then implement the MH algorithm one block at a time.

- In particular, suppose we partition $\theta = \left(\theta_1', \theta_2'\right)'$ and that we wish to obtain samples from the joint posterior distribution, $p\left(\theta_1, \theta_2 | Y\right)$. Design two MH transition kernels, one with stationary distribution $p\left(\theta_1 | \theta_2, Y\right)$ and one with stationary distribution $p\left(\theta_2 | \theta_1, Y\right)$. We can then implement an MH algorithm one block at a time as follows:

  1. Generate a proposed value of $\theta_1^{[g+1]}$, called θ_1^*, from a proposal distribution, $q_1\left(\cdot\right)$. To ease notation, we will assume that the proposal doesn't depend on $\theta_2^{[g]}$, and so can be written as $q_1\left(\theta_1 | \theta_1^{[g]}\right)$. This is usually the case in practice.

2. Compute the acceptance probability:

$$\alpha\left(\theta_1^{[g]}, \theta_1^*\right) = \min\left(\frac{p\left(\theta_1^*|\theta_2^{[g]}, Y\right) q_1\left(\theta_1^{[g]}|\theta_1^*\right)}{p\left(\theta_1^{[g]}|\theta_2^{[g]}, Y\right) q_1\left(\theta_1^*|\theta_1^{[g]}\right)}, 1\right)$$
$$= \min\left(\frac{p\left(Y|\theta_1^*, \theta_2^{[g]}\right) p\left(\theta_1^*|\theta_2^{[g]}\right) q_1\left(\theta_1^{[g]}|\theta_1^*\right)}{p\left(Y|\theta_1^{[g]}, \theta_2^{[g]}\right) p\left(\theta_1^{[g]}|\theta_2^{[g]}\right) q_1\left(\theta_1^*|\theta_1^{[g]}\right)}, 1\right)$$

3. Set $\theta_1^{[g+1]} = \theta_1^*$ with probability α and $\theta_1^{[g+1]} = \theta_1^{[g]}$ with probability $1 - \alpha$.

4. Generate a proposed value of $\theta_2^{[g+1]}$, called θ_2^*, from a **proposal distribution**, $q_2\left(\theta_2|\theta_2^{[g]}\right)$.

5. Compute the acceptance probability:

$$\alpha\left(\theta_2^{[g]}, \theta_2^*\right) = \min\left(\frac{p\left(\theta_2^*|\theta_1^{[g+1]}, Y\right) q_2\left(\theta_2^{[g]}|\theta_2^*\right)}{p\left(\theta_2^{[g]}|\theta_1^{[g+1]}, Y\right) q_2\left(\theta_2^*|\theta_2^{[g]}\right)}, 1\right)$$
$$= \min\left(\frac{p\left(Y|\theta_1^{[g+1]}, \theta_2^*\right) p\left(\theta_2^*|\theta_1^{[g+1]}\right) q_2\left(\theta_2^{[g]}|\theta_2^*\right)}{p\left(Y|\theta_1^{[g+1]}, \theta_2^{[g]}\right) p\left(\theta_2^{[g]}|\theta_1^{[g+1]}\right) q_2\left(\theta_2^*|\theta_2^{[g]}\right)}, 1\right)$$

6. Set $\theta_2^{[g+1]} = \theta_2^*$ with probability α and $\theta_2^{[g+1]} = \theta_2^{[g]}$ with probability $1 - \alpha$.

7. Increment g and go to 1.

The chain can be initialized with arbitrary initial value $\theta_2^{[0]}$.

- It is somewhat surprising that this algorithm works. The draws in each step are only one draw from a chain that has a convergent distribution equal to a conditional posterior distribution, where the conditioning is with respect to the previous draw from the other block. However, it can be shown that the draws from this algorithm will be reversible with stationary distribution $p(\theta|Y)$. The fact that this is true is known as the **product of kernels principle**.

- The above algorithm can be generalized in the obvious way to a MH sampler with J blocks.

- Why we would want to run a MH sampler with more than one block? One reason is that it may make calibrating proposal densities easier. If there are lots of parameters, or if parameters are of very different types, breaking these into groups can make thinking about proposal densities simpler.

- Another reason is that you may be running a "Metropolis within Gibbs" sampler. We will talk about these after we cover the Gibbs Sampler.

4.2 Metropolis-Hastings Example: Nonlinear Regression

- Consider the nonlinear regression model:

$$Y = f(X, \gamma) + \epsilon$$

  where $Y = (y_1, y_2, \cdots, y_N)$, X $= (X_1, X_2, \cdots, X_k)$ is the standard matrix of k right-hand side variables, and γ is a vector of parameters. The function $f(\cdot)$ produces a vector output.

- The likelihood function for this model will be determined by the assumption about ϵ.

- The prior will be denoted $p(\gamma, h)$.

- Finally, we would need a proposal density, $q\left(\gamma, h|\gamma^{[g]}, h^{[g]}\right)$.

- Given these three items, we then implement the MH algorithm using the steps described above.

- To gain additional insights we consider a specific example. Suppose y_i measures the output of the i^{th} firm, and $x_{1,i}$ and $x_{2,i}$ measure labor and capital input for this firm.

Suppose output is produced via a CES production function plus a constant and a random disturbance term unique to the i^{th} firm.

$$y_i = \gamma_1 + \left(\gamma_2 x_{1,i}^{\gamma_4} + \gamma_3 x_{2,i}^{\gamma_4}\right)^{\frac{1}{\gamma_4}} + \epsilon_i$$

Suppose we have N observations $Y = (y_1, y_2, \cdots, y_N)$, X $= (X_1, X_2)$. We write the model as:

$$Y = f(X, \gamma) + \epsilon$$

where $\gamma = (\gamma_1, \gamma_2, \gamma_3, \gamma_4)$, $\epsilon = (\epsilon_1, \epsilon_2, \cdots, \epsilon_N)$, and the i^{th} element of $f(\cdot)$ is:

$$\gamma_1 + \left(\gamma_2 x_{1,i}^{\gamma_4} + \gamma_3 x_{2,i}^{\gamma_4}\right)^{\frac{1}{\gamma_4}}$$

We assume that:

$$\epsilon \sim N\left(0, h^{-1} I_N\right).$$

- **Likelihood Function:**

$$p(Y|\gamma, h, \mathrm{X}) = (2\pi)^{-\frac{N}{2}} h^{\frac{N}{2}} \exp\left[-\frac{h}{2}(Y - f(X, \gamma))'(Y - f(X, \gamma))\right]$$

- **Prior Distribution:**

We assume independent Normal and gamma priors for γ and h, so that $p(\gamma, h) = p(\gamma)\, p(h)$

$$\gamma \sim N(\mu, V)$$

$$h \sim Gamma(m, v)$$

So:

$$p(\gamma) = (2\pi)^{-\frac{k}{2}} |V|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\gamma - \mu)' V^{-1} (\gamma - \mu)\right]$$

$$p(h) = \frac{1}{\left(\frac{2m}{v}\right)^{v/2}} \frac{1}{\Gamma\left(\frac{v}{2}\right)} h^{\frac{v-2}{2}} \exp\left[-\frac{hv}{2m}\right]$$

- **Posterior Distribution:**

  Again, to implement the MH Sampler we only need evaluate the kernel of the posterior distribution. Using Bayes Rule, and eliminating all the terms that don't depend on γ or h this is given by:

$$\begin{aligned}
&p(\gamma, h|Y, \mathrm{X}) \\
&\quad \propto \quad p(Y|\gamma, h, \mathrm{X})\, p(\gamma, h|\mathrm{X}) \\
&\quad \propto \quad p(Y|\gamma, h, \mathrm{X})\, p(\gamma)\, p(h) \\
&\quad \propto \quad (2\pi)^{-\frac{N}{2}} h^{\frac{N}{2}} \exp\left[-\frac{h}{2}(Y - f(X,\gamma))'(Y - f(X,\gamma))\right] \\
&\quad \times \quad (2\pi)^{-\frac{k}{2}} |V|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\gamma - \mu)' V^{-1}(\gamma - \mu)\right] \\
&\quad \times \quad \frac{1}{\left(\frac{2m}{v}\right)^{v/2}} \frac{1}{\Gamma\left(\frac{v}{2}\right)} h^{\frac{v-2}{2}} \exp\left[-\frac{hv}{2m}\right] \\
&\quad \propto \quad h^{\frac{N}{2}} \exp\left[-\frac{h}{2}(Y - f(X,\gamma))'(Y - f(X,\gamma))\right] \exp\left[-\frac{1}{2}(\gamma - \mu)' V^{-1}(\gamma - \mu)\right] \\
&\quad \times \quad h^{\frac{v-2}{2}} \exp\left[-\frac{hv}{2m}\right]
\end{aligned}$$

  Note that we can evaluate these terms, and this is all that is required to compute the acceptance rate.

- **Proposal Density:**

  Collect the parameters into the vector $\theta = (\gamma, h)$. For the proposal density we consider a random walk proposal:

$$\theta^* = \theta^{[g]} + v$$

  where $v \sim N(0_5, R)$. In this case, R is a 5×5 variance-covariance matrix. To calibrate

R we might use the variance covariance matrix of the large sample approximation to the posterior:

$$R = \left[- \left[\frac{d^2}{d\theta^2} \ln \left(p\left(\theta | Y\right) \right) \right]_{\theta = \widehat{\theta}} \right]^{-1} = \left[- \left[\frac{d^2}{d\theta^2} \ln \left(p\left(Y | \theta\right) p\left(\theta\right) \right) \right]_{\theta = \widehat{\theta}} \right]^{-1}.$$

where $\widehat{\theta}$ is the mode of the posterior distribution.

- **Acceptance Probability:**

  Given the above, the acceptance probability is then:

$$\begin{aligned} \alpha\left(\theta^{[g]}, \theta^*\right) &= \min\left(\frac{p\left(Y | \theta^*\right) p\left(\theta^*\right)}{p\left(Y | \theta^{[g]}\right) p\left(\theta^{[g]}\right)}, 1 \right) \\ &= \min\left(\frac{p\left(Y | \gamma^*, h^*\right) p\left(\gamma^*\right) p\left(h^*\right)}{p\left(Y | \gamma^{[g]}, h^{[g]}\right) p\left(\gamma^{[g]}\right) p\left(h^{[g]}\right)}, 1 \right) \end{aligned}$$

- **MH Algorithm**

  The MH sampler for our nonlinear regression model is then implemented as follows:

  1. Generate a proposal $\theta^* = (\gamma^*, h^*)$, as:

$$\theta^* = \theta^{[g]} + v$$

     where $v \sim N\left(0_5, R\right)$.

  2. Set $\theta^{[g+1]} = \theta^*$ with probability $\alpha\left(\theta^{[g]}, \theta^*\right)$ and $\theta^{[g+1]} = \theta^{[g]}$ with probability $1 - \alpha\left(\theta^{[g]}, \theta^*\right)$.

  3. Increment g and go to 1.

     The sampler is initiated with an arbitrary initial values, $\theta^{[0]} = \left(\gamma^{[0]}, h^{[0]}\right)$.

- Note that in the above sampler, it is possible for the proposal distribution to produce a proposed value for h that is negative. However, this proposal will be accepted with

probability zero, as $p(h^*)$ will be zero in the numerator of the acceptance probability.

5 Exercise: Nonlinear Regression

Consider the nonlinear regression model:

$$y_i = \left(\gamma_1 x_{1,i}^{\gamma_3} + \gamma_2 x_{2,i}^{\gamma_3}\right)^{\frac{1}{\gamma_3}} + \epsilon_i$$

Suppose we have N observations $Y = (y_1, y_2, \cdots, y_N)'$, $X_1 = (x_{1,1}, x_{1,2} \cdots, x_{1,N})'$, and $X_2 = (x_{2,1}, x_{2,2} \cdots, x_{2,N})'$. We write the model as:

$$Y = f(X_1, X_2, \gamma) + \epsilon$$

where $\epsilon = (\epsilon_1, \epsilon_2, \cdots, \epsilon_N)'$, $\gamma = (\gamma_1, \gamma_2, \gamma_3)'$, and the i^{th} element of $f(\cdot)$ is:

$$\left(\gamma_1 x_{1,i}^{\gamma_3} + \gamma_2 x_{2,i}^{\gamma_3}\right)^{\frac{1}{\gamma_3}}$$

We assume that:

$$\epsilon \sim N\left(0, h^{-1} I_N\right).$$

In the zip file "nonlinear regression.zip" you will find an Excel dataset called "nonlin_regression_data.xls," which holds data on $N = 100$ observations of Y, X_1 and X_2. You will also find a collection of Matlab programs. The main program in this folder is "NL_Regression_MH.m" and the rest are functions called by this program. The code is designed to sample from the posterior distribution $p(h, \gamma|Y)$. The code assumes the following priors:

$$p(\gamma, h) = p(\gamma)\, p(h)$$

$$\gamma \sim N(\mu, V)$$

$$h \sim Gamma\,(m, v)$$

The programs use a random walk proposal distribution:

$$\begin{bmatrix} \gamma^* \\ h^* \end{bmatrix} = \begin{bmatrix} \gamma^{[g]} \\ h^{[g]} \end{bmatrix} + \eta$$

where $\eta \sim N\,(0_4, R)$. To set R, we first calculate the asymptotic covariance matrix for the maximum likelihood estimates, call this $\widehat{\Sigma}$ and then set $R = \tau\widehat{\Sigma}$. In this program, setting $\tau = 1$ yields acceptance rates of around 40%.

Begin by looking through this code to make sure you understand how the code implements the MH algorithm. Take particular note of how the acceptance probability is calculated, first by computing the log of the acceptance probability and then exponentiating. This is done because the likelihood function values in the numerator and denominator of the acceptance probability can be so small that they will be numerically rounded to zero by MATLAB. By taking logs this problem is alleviated. In other words, to a computer, $p\,(Y|\gamma, h)$ reaches 0 faster than $\log\,(p\,(Y|\gamma, h))$ reaches $-\infty$.

Once you understand the code, experiment with different runs of the sampler, where you change the value of τ. Notice what happens to the acceptance rate as you do this.

See if you can calibrate the proposal density in a different way and still get reasonable acceptance rates. For example, try setting $R = \tau I_k$, where I_k is the $k \times k$ identity matrix.