Preliminaries: Likelihood for the linear regression model

$$y = X\beta + e, \quad e \sim N\left(0, \sigma^2 I_n\right),$$

- The likelihood for the linear model is:

$$p(y_1, y_2, \ldots, y_n | X, \beta, \sigma^2) = p(y|X, \beta, \sigma^2) = (2\pi\sigma^2)^{-n/2} \exp\left(-\frac{Q}{2\sigma^2}\right),$$

with

$$Q = (y - X\beta)'(y - X\beta) = e'e \; (= \sum_{i=1}^{n} e_i^2).$$

Heteroskedasticity of known form example

- The model, $y = X\beta + e$, $e \sim N(0, V)$.

Heteroskedasticity of known form example

- The model, $y = X\beta + e$, $e \sim N(0, V)$.
- Suppose we specify the error variance as a function of some variable(s) z.

Heteroskedasticity of known form example

- The model, $y = X\beta + e$, $e \sim N(0, V)$.
- Suppose we specify the error variance as a function of some variable(s) z.
- A standard specification is an exponential function, $\sigma_i^2 = \exp(z_i\alpha)$ (another is $\sigma_i^2 = \sigma^2 z_i^2$), for example

$$E(e_i^2) = \sigma_i^2 = \exp(\alpha_1 + \alpha_2 z_i).$$

Maximum likelihood estimation

- The log likelihood function for the model $y = X\beta + e$, $e \sim N(0, V)$, with the ith diagonal element of V given by $\sigma_i^2 = \exp(\alpha_1 + \alpha_2 z_i) = \exp(-\mathbf{z}_i\alpha)$ is

$$\ln L(\beta, \sigma^2, V|y, X) = -\frac{n}{2}\ln 2\pi - \frac{1}{2}\ln|V| - \frac{(y - X\beta)'V^{-1}(y - X\beta)}{2}.$$

$$= -\frac{n}{2}\ln 2\pi - \frac{1}{2}\sum_{i=1}^{m}\mathbf{z}_i\alpha - \frac{1}{2}\sum_{i=1}^{m}\exp(-\mathbf{z}_i\alpha)(y_i - X_i\beta)^2.$$

where $\mathbf{z}_i = [1\ z_i]$ is the ith row of the matrix of covariates for the variance, $Z = [1\,\mathbf{z}]$.

Maximum likelihood estimation

- The log likelihood function for the model $y = X\beta + e$, $e \sim N(0, V)$, with the ith diagonal element of V given by $\sigma_i^2 = \exp(\alpha_1 + \alpha_2 z_i) = \exp(-\mathbf{z}_i\alpha)$ is

$$\ln L(\beta, \sigma^2, V|y, X) = -\frac{n}{2}\ln 2\pi - \frac{1}{2}\ln|V| - \frac{(y - X\beta)'V^{-1}(y - X\beta)}{2}.$$

$$= -\frac{n}{2}\ln 2\pi - \frac{1}{2}\sum_{i=1}^{m}\mathbf{z}_i\alpha - \frac{1}{2}\sum_{i=1}^{m}\exp(-\mathbf{z}_i\alpha)(y_i - X_i\beta)^2.$$

  where $\mathbf{z}_i = [1\ z_i]$ is the ith row of the matrix of covariates for the variance, $Z = [1\,\mathbf{z}]$.

- Note that σ_i^2 can be written as $\sigma_i^2 = \exp(\alpha_1 + \alpha_2 z_i) = \exp(\alpha_1)\exp(\alpha_2 z_i) = \sigma^2\exp(\alpha_2 z_i)$, where $\sigma^2 = \exp(\alpha_1)$

Maximum likelihood estimation

- The log likelihood function for the model $y = X\beta + e$, $e \sim N(0, V)$, with the ith diagonal element of V given by $\sigma_i^2 = \exp(\alpha_1 + \alpha_2 z_i) = \exp(-\mathbf{z}_i\alpha)$ is

$$\ln L(\beta, \sigma^2, V|y, X) = -\frac{n}{2}\ln 2\pi - \frac{1}{2}\ln|V| - \frac{(y - X\beta)'V^{-1}(y - X\beta)}{2}.$$

$$= -\frac{n}{2}\ln 2\pi - \frac{1}{2}\sum_{i=1}^{m}\mathbf{z}_i\alpha - \frac{1}{2}\sum_{i=1}^{m}\exp(-\mathbf{z}_i\alpha)(y_i - X_i\beta)^2.$$

  where $\mathbf{z}_i = [1\ z_i]$ is the ith row of the matrix of covariates for the variance, $Z = [1\,\mathbf{z}]$.

- Note that σ_i^2 can be written as $\sigma_i^2 = \exp(\alpha_1 + \alpha_2 z_i) = \exp(\alpha_1)\exp(\alpha_2 z_i) = \sigma^2\exp(\alpha_2 z_i)$, where $\sigma^2 = \exp(\alpha_1)$

- We can use an iterative numerical optimization routine to compute the argmax for β and α using this log likelihood (typically some form of quadratic hill climbing).

Best algorithms of the 20th Century

- Dongarra and Sullivan (2000) *Computing in Science and Engineering*, *2*, 22-23.
- The top 10 algorithms with the greatest influence on the development and practice of science and engineering in the 20th century (in chronological order)

1. **Metropolis algorithm for Monte Carlo**
2. Simplex method for linear programming
3. Krylov subspace iteration methods
4. The decompositional approach to matrix computations
5. The Fortran optimizing compiler
6. QR algorithm for computing eigenvalues
7. Quicksort algorithm for sorting
8. Fast Fourier transform
9. Integer Relation Detection
10. Fast Multipole Method

A Brief history of MCMC Theory

- While convalescing from an illness in 1946, Stan Ulum was playing solitaire. It occurred to him to try and compute the chances that a particular laid out solitaire of 52 cards would come out successfully.

A Brief history of MCMC Theory

- While convalescing from an illness in 1946, Stan Ulum was playing solitaire. It occurred to him to try and compute the chances that a particular laid out solitaire of 52 cards would come out successfully.
- After attempting exhaustive combinatorial calculations, he decided to go for a more practical approach, laying out several solitaires at random and then observing and counting the number of successful plays.

A Brief history of MCMC Theory

- While convalescing from an illness in 1946, Stan Ulum was playing solitaire. It occurred to him to try and compute the chances that a particular laid out solitaire of 52 cards would come out successfully.
- After attempting exhaustive combinatorial calculations, he decided to go for a more practical approach, laying out several solitaires at random and then observing and counting the number of successful plays.
- This idea of selecting a statistical sample to approximate a hard combinatorial problem by a much simpler problem is the main idea behind modern Monte Carlo simulation methods.

A Brief history of MCMC Theory

- While convalescing from an illness in 1946, Stan Ulum was playing solitaire. It occurred to him to try and compute the chances that a particular laid out solitaire of 52 cards would come out successfully.
- After attempting exhaustive combinatorial calculations, he decided to go for a more practical approach, laying out several solitaires at random and then observing and counting the number of successful plays.
- This idea of selecting a statistical sample to approximate a hard combinatorial problem by a much simpler problem is the main idea behind modern Monte Carlo simulation methods.
- Ulum realized that computers could be used in this way to answer questions of neutron diffusion and mathematical physics.

A Brief history of MCMC Theory (cont.)

- He contacted John Von Neumann and they developed many Monte Carlo algorithms that have been rediscovered in 1980s and 90s (importance sampling, rejection sampling, etc.)

A Brief history of MCMC Theory (cont.)

- He contacted John Von Neumann and they developed many Monte Carlo algorithms that have been rediscovered in 1980s and 90s (importance sampling, rejection sampling, etc.)
- First paper: Metropolis and Ulum (**1949**) The Monte Carlo Method, *JASA*.

A Brief history of MCMC Theory (cont.)

- He contacted John Von Neumann and they developed many Monte Carlo algorithms that have been rediscovered in 1980s and 90s (importance sampling, rejection sampling, etc.)
- First paper: Metropolis and Ulum (**1949**) The Monte Carlo Method, *JASA*.
- Coworkers at Los Alamos, Metropolis, Teller, Von Neuman, *et al.*, coded these methods for use with the state-of-the-art computer (ENIAC).

A Brief history of MCMC Theory (cont.)

- He contacted John Von Neumann and they developed many Monte Carlo algorithms that have been rediscovered in 1980s and 90s (importance sampling, rejection sampling, etc.)
- First paper: Metropolis and Ulum (**1949**) The Monte Carlo Method, *JASA.*
- Coworkers at Los Alamos, Metropolis, Teller, Von Neuman, *et al.,* coded these methods for use with the state-of-the-art computer (ENIAC).
- Metropolis *et al.* (**1953**) Equations of state calculations by fast computing machines, *Journal of Chemical Physics,* was the pioneering paper on MCMC, but it was overlooked by statisticians, partly because it was published in a chemistry journal, and partly because of the primitive level of computer technology available at the time.

How long until we recognize a good idea? 1970-1990s

- Hastings (**1970**) *Biometrika* and his student Peskun (1973) *Biometrika* generalized the Metropolis algorithm.

How long until we recognize a good idea? 1970-1990s

- Hastings (**1970**) *Biometrika* and his student Peskun (1973) *Biometrika* generalized the Metropolis algorithm.
- Geman and Geman (1984) *IEEE Transactions* developed the Gibbs sampler for use in image processing, and Tanner and Wong (1987) *JASA* developed the data augmentation approach and were arguably the first to recognize the potential for Bayesian MCMC inference.

How long until we recognize a good idea? 1970-1990s

- Hastings (**1970**) *Biometrika* and his student Peskun (1973) *Biometrika* generalized the Metropolis algorithm.
- Geman and Geman (1984) *IEEE Transactions* developed the Gibbs sampler for use in image processing, and Tanner and Wong (1987) *JASA* developed the data augmentation approach and were arguably the first to recognize the potential for Bayesian MCMC inference.
- However, it was the classic expository paper by Gelfand and Smith (**1990**) *JASA*, that brought the Gibbs sampler to the attention of a wider audience.

MCMC methods developed mostly in 1990s

- Widespread recognition of the practical importance of these algorithms occurred among statisticians in the 1990s (following Gelfand and Smith (1990)).

MCMC methods developed mostly in 1990s

- Widespread recognition of the practical importance of these algorithms occurred among statisticians in the 1990s (following Gelfand and Smith (1990)).
- This led to the rapid development of a generic set of MCMC tools for Bayesian inference and subsequently revolutionized the field of statistics.

MCMC methods developed mostly in 1990s

- Widespread recognition of the practical importance of these algorithms occurred among statisticians in the 1990s (following Gelfand and Smith (1990)).
- This led to the rapid development of a generic set of MCMC tools for Bayesian inference and subsequently revolutionized the field of statistics.
- Most of the theoretical developments in MCMC were achieved in the 1990s.

MCMC methods developed mostly in 1990s

- Widespread recognition of the practical importance of these algorithms occurred among statisticians in the 1990s (following Gelfand and Smith (1990)).
- This led to the rapid development of a generic set of MCMC tools for Bayesian inference and subsequently revolutionized the field of statistics.
- Most of the theoretical developments in MCMC were achieved in the 1990s.
- 2010+: An Econometric Oddity - There are still many "frequentists" in econometrics who refuse to use these methods just "because I'm a frequentist"!

Econometrics for the 21st century

- The posterior distribution for many models is often not available in an analytical form.

Econometrics for the 21st century

- The posterior distribution for many models is often not available in an analytical form.
- The discovery that MCMC computational methods could be used to simulate from general Bayesian Hierarchical Models (Gelfand and Smith, 1990) revolutionized Statistical Science and made ever more complicated modeling scenarios possible." [Cressie & Wilke, p.31]

Econometrics for the 21st century

- The posterior distribution for many models is often not available in an analytical form.
- The discovery that MCMC computational methods could be used to simulate from general Bayesian Hierarchical Models (Gelfand and Smith, 1990) revolutionized Statistical Science and made ever more complicated modeling scenarios possible." [Cressie & Wilke, p.31]
- "The emphasis in Statistical Science has moved from being able to derive *analytical* and/or asymptotic expressions for $p(X|D)$ and its properties, to being able to *simulate* from $p(X|D)$.

Econometrics for the 21st century

- The posterior distribution for many models is often not available in an analytical form.
- The discovery that MCMC computational methods could be used to simulate from general Bayesian Hierarchical Models (Gelfand and Smith, 1990) revolutionized Statistical Science and made ever more complicated modeling scenarios possible." [Cressie & Wilke, p.31]
- "The emphasis in Statistical Science has moved from being able to derive *analytical* and/or asymptotic expressions for $p(X|D)$ and its properties, to being able to *simulate* from $p(X|D)$.
- Analytical calculations are still important because from them come deep understanding and sharp focus; however, the benefits of simulation are enormous." [CW (2011), p.44]

MCMC - Motivation

- *Simulation* from this distribution allows statistical inference to proceed, often in a relatively straightforward manner.

MCMC - Motivation

- *Simulation* from this distribution allows statistical inference to proceed, often in a relatively straightforward manner.
- Bayesian methods have recently produced some remarkably efficient solutions to complex inference problems.

MCMC - Motivation

- *Simulation* from this distribution allows statistical inference to proceed, often in a relatively straightforward manner.
- Bayesian methods have recently produced some remarkably efficient solutions to complex inference problems.
- The approach is based on a combination of hierarchical prior modeling and MCMC simulation methods.

MCMC - Motivation

- *Simulation* from this distribution allows statistical inference to proceed, often in a relatively straightforward manner.
- Bayesian methods have recently produced some remarkably efficient solutions to complex inference problems.
- The approach is based on a combination of hierarchical prior modeling and MCMC simulation methods.
- This approach is able to tackle estimation and model interpretation in situations that are quite challenging by other means.

MCMC - Motivation

- See Greenberg, Part II - also Lancaster, chapter 4, Chib Handbook of Econometrics article, Gamerman and Lopes book for more details.

MCMC - Motivation

- See Greenberg, Part II - also Lancaster, chapter 4, Chib Handbook of Econometrics article, Gamerman and Lopes book for more details.
- Suppose you have a (posterior) distribution for some unknown quantities $\theta = (\theta_1, \theta_2)$ (two unknowns).

MCMC - Motivation

- See Greenberg, Part II - also Lancaster, chapter 4, Chib Handbook of Econometrics article, Gamerman and Lopes book for more details.
- Suppose you have a (posterior) distribution for some unknown quantities $\theta = (\theta_1, \theta_2)$ (two unknowns).
- If you draw (pseudo-) randomly from this distribution, $p(\theta_1, \theta_2|y)$, R times (i.e. R is the number of replications), you will have

$$\theta_R = \begin{bmatrix} \theta_1^{(1)} & \theta_2^{(1)} \\ \theta_1^{(2)} & \theta_2^{(2)} \\ \vdots & \vdots \\ \theta_1^{(R)} & \theta_2^{(R)} \end{bmatrix}.$$

MCMC Motivation (cont.)

- Each row of θ_R represents a randomly selected 'observation'/sampling from the **joint** distribution of θ_1 and θ_2, $p(\theta_1, \theta_2|y)$.

MCMC Motivation (cont.)

- Each row of θ_R represents a randomly selected 'observation'/sampling from the **joint** distribution of θ_1 and θ_2, $p(\theta_1, \theta_2|y)$.
- Each column contains R realizations/'observations' from the **marginal** distribution of θ_j, $p(\theta_j|y)$.

MCMC Motivation (cont.)

- Each row of θ_R represents a randomly selected 'observation'/sampling from the **joint** distribution of θ_1 and θ_2, $p(\theta_1, \theta_2|y)$.
- Each column contains R realizations/'observations' from the **marginal** distribution of θ_j, $p(\theta_j|y)$.
- So we can study the distribution of θ_j simply by ignoring the other column - "it's as simple as that." [Lancaster, p.52]

MCMC Motivation (cont.)

- Each row of θ_R represents a randomly selected 'observation'/sampling from the **joint** distribution of θ_1 and θ_2, $p(\theta_1, \theta_2|y)$.
- Each column contains R realizations/'observations' from the **marginal** distribution of θ_j, $p(\theta_j|y)$.
- So we can study the distribution of θ_j simply by ignoring the other column - "it's as simple as that." [Lancaster, p.52]
- "Computer assisted sampling to avoid integration is the key feature of this approach. Increasingly, difficult mathematics is being abandoned in favor of computer power." [Lancaster, p.52]

What to do with a sample of observations

- If we have a sample of n observations of some 'random variable' X_i $(i = 1, 2, \ldots, n)$, we know how to summarize information about this sample (from intro. statistics!).

What to do with a sample of observations

- If we have a sample of n observations of some 'random variable' X_i $(i = 1, 2, \ldots, n)$, we know how to summarize information about this sample (from intro. statistics!).
- We can calculate the mean, variance, standard deviation, etc.

What to do with a sample of observations

- If we have a sample of n observations of some 'random variable' X_i $(i = 1, 2, \ldots, n)$, we know how to summarize information about this sample (from intro. statistics!).
- We can calculate the mean, variance, standard deviation, etc.
- We can (nonparameterically) estimate the sampling distribution of X_i with a frequency count and plot histograms, etc.

What to do with a sample of observations

- If we have a sample of n observations of some 'random variable' X_i $(i = 1, 2, \ldots, n)$, we know how to summarize information about this sample (from intro. statistics!).
- We can calculate the mean, variance, standard deviation, etc.
- We can (nonparameterically) estimate the sampling distribution of X_i with a frequency count and plot histograms, etc.
- We can construct confidence intervals, perform hypothesis tests, etc., using the sample.

What to do with a sample of observations

- If we have a sample of n observations of some 'random variable' X_i $(i = 1, 2, \ldots, n)$, we know how to summarize information about this sample (from intro. statistics!).
- We can calculate the mean, variance, standard deviation, etc.
- We can (nonparameterically) estimate the sampling distribution of X_i with a frequency count and plot histograms, etc.
- We can construct confidence intervals, perform hypothesis tests, etc., using the sample.
- If we recognize that with an MCMC sample for θ_j we are in exactly this situation.

Markov Chain Monte Carlo (MCMC): the idea

- The goal of Bayesian computation is to obtain a sample of draws $\theta^{(t)}$, $t = 1, \ldots, M$, from the posterior distribution of the unknown quantity θ, with a large enough sample that quantities of interest can be estimated with reasonable accuracy.

Markov Chain Monte Carlo (MCMC): the idea

- The goal of Bayesian computation is to obtain a sample of draws $\theta^{(t)}$, $t = 1, \ldots, M$, from the posterior distribution of the unknown quantity θ, with a large enough sample that quantities of interest can be estimated with reasonable accuracy.
- MCMC simulation is a general method based on drawing values of θ from distributions that result in a sample from the target posterior distribution, $p(\theta|y)$.

Markov Chain Monte Carlo (MCMC): the idea

- The goal of Bayesian computation is to obtain a sample of draws $\theta^{(t)}$, $t = 1, \ldots, M$, from the posterior distribution of the unknown quantity θ, with a large enough sample that quantities of interest can be estimated with reasonable accuracy.
- MCMC simulation is a general method based on drawing values of θ from distributions that result in a sample from the target posterior distribution, $p(\theta|y)$.
- The sample is drawn sequentially, with the tth draw, $\theta^{(t)}$, depending only on the previous draw, $\theta^{(t-1)}$.

Some Markov chain theory

- This dependence on only the previous draw is the defining property of a Markov chain, so that MCMC is a practical application of Markov chain theory.

Some Markov chain theory

- This dependence on only the previous draw is the defining property of a Markov chain, so that MCMC is a practical application of Markov chain theory.
- Some understanding of the theory of Markov chains is thus helpful in practice, particularly in evaluating the performance and convergence of MCMC chains.

Some Markov chain theory

- This dependence on only the previous draw is the defining property of a Markov chain, so that MCMC is a practical application of Markov chain theory.
- Some understanding of the theory of Markov chains is thus helpful in practice, particularly in evaluating the performance and convergence of MCMC chains.
- See Greenberg, part II. Many other texts available, two particularly good ones are:
- Gelman et al. (2004) and
- Gamerman and Lopes (2006).

Markov Chain basics

- Consider a stochastic process (r.v. if you prefer) indexed by time or iteration, X_t, that takes only a finite set of values, $S = \{1, 2, \ldots s\}$. E.g. rolls of a six-sided die, number of sales of a particular product in a given time period (a day or week say).

Markov Chain basics

- Consider a stochastic process (r.v. if you prefer) indexed by time or iteration, X_t, that takes only a finite set of values, $S = \{1, 2, \ldots s\}$. E.g. rolls of a six-sided die, number of sales of a particular product in a given time period (a day or week say).
- For integers $i < s$ and $j < s$, p_{ij} is defined as the probability that $X_{t+1} = j$ given that $X_t = i$, i.e.

$$p_{ij} = Pr(X_{t+1} = j | X_t = i), \qquad i, j \in S.$$

Markov Chain basics

- Consider a stochastic process (r.v. if you prefer) indexed by time or iteration, X_t, that takes only a finite set of values, $S = \{1, 2, \ldots s\}$. E.g. rolls of a six-sided die, number of sales of a particular product in a given time period (a day or week say).
- For integers $i < s$ and $j < s$, p_{ij} is defined as the probability that $X_{t+1} = j$ given that $X_t = i$, i.e.

$$p_{ij} = Pr(X_{t+1} = j | X_t = i), \qquad i, j \in S.$$

- The p_{ij} are the **transition probabilities**.

Markov Chain basics (cont.)

- The assumption that the probability distribution at $t+1$ depends **only** on the state of the system at t is called the Markov property and it is the defining property of a **Markov process** (i.e. $Pr(X_{t+1} = j | X_t = i, \ldots, X_{t-k} = m) = Pr(X_{t+1} = j | X_t = i)$).

Markov Chain basics (cont.)

- The assumption that the probability distribution at $t+1$ depends **only** on the state of the system at t is called the Markov property and it is the defining property of a **Markov process** (i.e. $Pr(X_{t+1} = j | X_t = i, \ldots, X_{t-k} = m) = Pr(X_{t+1} = j | X_t = i)$).
- Since the p_{ij} are probabilities, we have that $p_{ij} \geq 0$ and $\sum_{j=1}^{s} p_{ij} = 1$.

Markov Chain basics (cont.)

- The assumption that the probability distribution at $t+1$ depends **only** on the state of the system at t is called the Markov property and it is the defining property of a **Markov process** (i.e. $Pr(X_{t+1} = j | X_t = i, \ldots, X_{t-k} = m) = Pr(X_{t+1} = j | X_t = i)$).
- Since the p_{ij} are probabilities, we have that $p_{ij} \geq 0$ and $\sum_{j=1}^{s} p_{ij} = 1$.
- We can write this in matrix form, defining a **transition matrix**, $P = [p_{ij}]$.

Markov Chain basics (cont.)

- The ith row of P specifies the distribution of the process at $t+1$, given that it is in state i at t. For example, suppose

$$P = \begin{bmatrix} 0.75 & 0.25 \\ 0.125 & 0.875 \end{bmatrix}$$

for a process with two possible states (e.g. a dependent coin toss - flip unfair coin 1 if last toss was heads, unfair coin 2 if tails).

Markov Chain basics (cont.)

- The ith row of P specifies the distribution of the process at $t+1$, given that it is in state i at t. For example, suppose

$$P = \begin{bmatrix} 0.75 & 0.25 \\ 0.125 & 0.875 \end{bmatrix}$$

  for a process with two possible states (e.g. a dependent coin toss - flip unfair coin 1 if last toss was heads, unfair coin 2 if tails).
- The process will remain in state 1 with probability 0.75 and moves to state 2 with probability 0.25 if it starts in state 1.

Markov Chain basics (cont.)

- The ith row of P specifies the distribution of the process at $t+1$, given that it is in state i at t. For example, suppose

$$P = \begin{bmatrix} 0.75 & 0.25 \\ 0.125 & 0.875 \end{bmatrix}$$

  for a process with two possible states (e.g. a dependent coin toss - flip unfair coin 1 if last toss was heads, unfair coin 2 if tails).
- The process will remain in state 1 with probability 0.75 and moves to state 2 with probability 0.25 if it starts in state 1.
- If the process starts in state 2, it moves to state 1 with prob. 0.125 and stays in state 2 with prob. 0.875.

Markov Chain basics (cont.)

- Now consider the distribution of the state at $t+2$ given that it is in state i at t, denoted $p_{ij}^{(2)}$

Markov Chain basics (cont.)

- Now consider the distribution of the state at $t+2$ given that it is in state i at t, denoted $p_{ij}^{(2)}$
- We can easily show that

$$p_{ij}^{(2)} = \sum_k p_{ik} p_{kj}.$$

Markov Chain basics (cont.)

- Now consider the distribution of the state at $t+2$ given that it is in state i at t, denoted $p_{ij}^{(2)}$
- We can easily show that

$$p_{ij}^{(2)} = \sum_k p_{ik} p_{kj}.$$

- and we can verify that the matrix form of this is given by $PP = P^2$.

Stationarity of the Markov Chain

- Then, by induction, we can show that the values of $p_{ij}^{(n)}$ are the ijth entries of the matrix P^n, where n is any integer.

Stationarity of the Markov Chain

- Then, by induction, we can show that the values of $p_{ij}^{(n)}$ are the ijth entries of the matrix P^n, where n is any integer.
- We are then, with regard to the use of MCMC methods, concerned with what happens to $p_{ij}^{(n)}$ as n becomes large.

Stationarity of the Markov Chain

- Then, by induction, we can show that the values of $p_{ij}^{(n)}$ are the ijth entries of the matrix P^n, where n is any integer.
- We are then, with regard to the use of MCMC methods, concerned with what happens to $p_{ij}^{(n)}$ as n becomes large.
- A key requirement for the application of MCMC methods is the convergence of the chain to a stationary distribution.

Stationarity of the Markov Chain

- Then, by induction, we can show that the values of $p_{ij}^{(n)}$ are the ijth entries of the matrix P^n, where n is any integer.
- We are then, with regard to the use of MCMC methods, concerned with what happens to $p_{ij}^{(n)}$ as n becomes large.
- A key requirement for the application of MCMC methods is the convergence of the chain to a stationary distribution.
- A probability distribution $\pi = (\pi_1, \pi_2, \ldots, \pi_s)'$ is an invariant or stationary distribution of a chain with transition probabilities P if $\pi' = \pi' P$ or

$$\pi_j = \sum_i \pi_i p_{ij}.$$

Convergence to the stationary distribution

- If the stationary distribution π exists and $lim_{n\to\infty}\pi P^n = \pi$, then, independently of the initial distribution of the chain, P^n will approach π, as $n \to \infty$.

Convergence to the stationary distribution

- If the stationary distribution π exists and $lim_{n\to\infty}\pi P^n = \pi$, then, independently of the initial distribution of the chain, P^n will approach π, as $n \to \infty$.
- For example, using the 2x2 P defined above, from $\pi' = \pi' P$, we have

$$(\pi_1 \pi_2) \begin{bmatrix} 0.750 & 0.250 \\ 0.125 & 0.875 \end{bmatrix} = (\pi_1 \pi_2),$$

so

$$0.750\pi_1 + 0.125\pi = \pi_1,$$

and since $\pi_2 = 1 - \pi_1$,

$$0.75\pi_1 + 0.125(1 - \pi_1) = \pi_1,$$

which implies $\pi = (1/3, 2/3)$.

Convergence to the stationary distribution (cont.)

- The above result tells us that there is a unique invariant/stationary distribution, and also that $p_{ij}^{(n)}$ **converges at a geometric rate** to π_j.

Convergence to the stationary distribution (cont.)

- The above result tells us that there is a unique invariant/stationary distribution, and also that $p_{ij}^{(n)}$ **converges at a geometric rate** to π_j.
- Further, for large enough n, the initial state i plays almost no role.

Convergence to the stationary distribution (cont.)

- The above result tells us that there is a unique invariant/stationary distribution, and also that $p_{ij}^{(n)}$ **converges at a geometric rate** to π_j.
- Further, for large enough n, the initial state i plays almost no role.
- So, in other words, P^n converges quickly to a matrix whose rows are all π'.

Example of speed of convergence

- See **markov.R**:

```
# markov.R - Markov chain example
P  <- matrix(c(0.75,0.25,0.125,0.875), nrow = 2, ncol=2, byrow=TRUE
n = 10 # calculate P^n
Z = P
for (i in 1:(n-1)) {
Z = Z%*%P  }
Z
```

- This result, in more general form, is the basis for MCMC methods.

Ergodicity

- Ergodicity concerns ensuring that the chain will visit all possible values under the support of the distribution of interest (the stationary distribution) with nonzero probability.

Ergodicity

- Ergodicity concerns ensuring that the chain will visit all possible values under the support of the distribution of interest (the stationary distribution) with nonzero probability.
- A chain is ergodic if it is aperiodic (so it cannot get stuck cycling in one subregion of the parameter space), and positive recurrent (which essentially means that as $n \to \infty$, the probability of visiting every possible state is nonzero).

Ergodicity

- Ergodicity concerns ensuring that the chain will visit all possible values under the support of the distribution of interest (the stationary distribution) with nonzero probability.
- A chain is ergodic if it is aperiodic (so it cannot get stuck cycling in one subregion of the parameter space), and positive recurrent (which essentially means that as $n \to \infty$, the probability of visiting every possible state is nonzero).
- For a Markov chain, $\theta = (\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(n)})$, the ergodic average of a real-valued function of θ, $h(\theta)$ is the average $\overline{h}_n = (1/n) \sum_{t=1}^{n} h(\theta^{(t)})$.

Law of large numbers for Markov chains

- If the chain is ergodic and $E_\pi[h(\theta)] < \infty$ for the unique limiting distribution π, then
- $$\overline{h}_n \xrightarrow{a.s.} E_\pi[h(\theta)] \text{ as } n \to \infty. \tag{1}$$

Law of large numbers for Markov chains

- If the chain is ergodic and $E_\pi[h(\theta)] < \infty$ for the unique limiting distribution π, then
- $$\overline{h}_n \xrightarrow{a.s.} E_\pi[h(\theta)] \text{ as } n \to \infty. \quad (1)$$
- This result is a Markov chain equivalent of the Law of Large Numbers (see Gamerman and Lopes 2006, p.125).

CLT for Markov chains

- If a chain is uniformly (geometrically) ergodic and $h^2(\theta)(h^{2+\epsilon}(\theta))$ is integrable with respect to π for some $\epsilon > 0$, then we can obtain a Central Limit Theorem for Markov chains,

$$\sqrt{n}\frac{\overline{h}_n - E_\pi[h(\theta)]}{\tau} =$$

$$\sqrt{n_{\text{eff}}}\frac{\overline{h}_n - E_p[h(\theta)]}{\sigma} \xrightarrow{d} N(0,1) \text{ as } n \to \infty, \tag{2}$$

where ...

CLT distribution parameters

- the variance of the limiting distribution π is given by,

$$\sigma^2 = \mathrm{var}(h(\theta))$$

CLT distribution parameters

- the variance of the limiting distribution π is given by,

$$\sigma^2 = \mathrm{var}(h(\theta))$$

- the limiting sample variance of the estimate $\overline{h}_n$ is,

$$\tau^2 = \sigma^2(1 + 2\sum_{k=1}^{\infty} \rho_k)$$

Relative numerical efficiency

- The inefficiency factor due to autocorrelation in the Markov chain is,

$$n_{\text{eff}} = n/(1 + 2\sum_{k=1}^{\infty} \rho_k) \tag{3}$$

where the autocorrelation is indicated by

$$\rho_k = \text{cov}(h(\theta^{(t)}, h(\theta^{(t-k)})/\sigma^2.$$

Relative numerical efficiency

- The inefficiency factor due to autocorrelation in the Markov chain is,

$$n_{\text{eff}} = n/(1 + 2\sum_{k=1}^{\infty} \rho_k) \tag{3}$$

where the autocorrelation is indicated by

$$\rho_k = \text{cov}(h(\theta^{(t)}, h(\theta^{(t-k)})/\sigma^2.$$

- The inefficiency factor n_{eff} is used in practice to measure the 'effective' random iid sample size of the MCMC chain by replaced the theoretical autocorrelations, ρ_k, with consistent sample estimates.

The detailed balance equation

- Equation (1) provides theoretical support for evaluating ergodic averages as estimates, and equation (2) supports evaluating approximate confidence intervals.

The detailed balance equation

- Equation (1) provides theoretical support for evaluating ergodic averages as estimates, and equation (2) supports evaluating approximate confidence intervals.
- One further point: a chain is said to be reversible if

$$\pi(x)P(x,y) = \pi(y)P(y,x) \,\text{for all}\, x, y \in S, \tag{4}$$

where the state space S is the appropriate subset of $\mathbb{R}^n$ representing the support of x, y.

The key theoretical result

- Equation (4) is known as the 'detailed balance equation' because it equates the rates of moves through states (so balanced) for every possible pair of states (hence detailed).

The key theoretical result

- Equation (4) is known as the 'detailed balance equation' because it equates the rates of moves through states (so balanced) for every possible pair of states (hence detailed).
- This leads to the key result:
- If there is a distribution p satisfying the detailed balance equation, (4), for an irreducible chain, then the chain is positive recurrent and reversible with stationary distribution π.

The key result for MCMC

- Metropolis et al. (1953) showed that it is then ***always*** possible to construct a Markov chain with stationary distribution π by finding transition probabilities $P(x, y)$ satisfying (4).

The key result for MCMC

- Metropolis et al. (1953) showed that it is then ***always*** possible to construct a Markov chain with stationary distribution π by finding transition probabilities $P(x, y)$ satisfying (4).
- This provides an algorithm for constructing Markov chains that has weak requirements and so has wide applicability.

The key result for MCMC

- Metropolis et al. (1953) showed that it is then ***always*** possible to construct a Markov chain with stationary distribution π by finding transition probabilities $P(x, y)$ satisfying (4).
- This provides an algorithm for constructing Markov chains that has weak requirements and so has wide applicability.
- The above results, in particular, convergence to the limiting distribution, the ergodic theorem and the central limit theorem, all hold for continuous state spaces with only minor technical modifications required (see Gamerman and Lopes, 2006).

The key result for MCMC

- Metropolis et al. (1953) showed that it is then ***always*** possible to construct a Markov chain with stationary distribution π by finding transition probabilities $P(x, y)$ satisfying (4).
- This provides an algorithm for constructing Markov chains that has weak requirements and so has wide applicability.
- The above results, in particular, convergence to the limiting distribution, the ergodic theorem and the central limit theorem, all hold for continuous state spaces with only minor technical modifications required (see Gamerman and Lopes, 2006).
- The above theory provides the means by which sampling from **virtually any** posterior distribution π can be achieved.

MCMC as a useful algorithm

- The basic Metropolis algorithm is to set the posterior distribution of interest, π, as the limiting distribution of an ergodic Markov chain with transition kernel P.

MCMC as a useful algorithm

- The basic Metropolis algorithm is to set the posterior distribution of interest, π, as the limiting distribution of an ergodic Markov chain with transition kernel P.
- The various algorithms that build on this, in particular Gibbs sampling and Metropolis-Hastings (MH), are concerned with various methods of providing proposal distributions P to be sampled from.

MCMC as a useful algorithm

- The basic Metropolis algorithm is to set the posterior distribution of interest, π, as the limiting distribution of an ergodic Markov chain with transition kernel P.
- The various algorithms that build on this, in particular Gibbs sampling and Metropolis-Hastings (MH), are concerned with various methods of providing proposal distributions P to be sampled from.
- So, Markov chain Monte Carlo (MCMC) is the generic algorithm that ensures the Markov chain's stationary distribution *is* the posterior distribution of the associated parameter/unknown quantity.

MCMC as a useful algorithm

- The basic Metropolis algorithm is to set the posterior distribution of interest, π, as the limiting distribution of an ergodic Markov chain with transition kernel P.
- The various algorithms that build on this, in particular Gibbs sampling and Metropolis-Hastings (MH), are concerned with various methods of providing proposal distributions P to be sampled from.
- So, Markov chain Monte Carlo (MCMC) is the generic algorithm that ensures the Markov chain's stationary distribution *is* the posterior distribution of the associated parameter/unknown quantity.
- The MCMC algorithm is iterative, but it does not converge in the usual, numerical analysis sense of converging to a solution. It converges in the statistical sense that each realization $X^{(r)}$ is distributed according to the stationary distribution of the Markov chain.

Treating the MCMC draws as a random sample

- Further, we can obtain **as large an MCMC sample as we like** just by making more draws using the MCMC algorithm.

Treating the MCMC draws as a random sample

- Further, we can obtain **as large an MCMC sample as we like** just by making more draws using the MCMC algorithm.
- We know, from the law of large numbers, that the sample mean converges on the actual mean.

Treating the MCMC draws as a random sample

- Further, we can obtain **as large an MCMC sample as we like** just by making more draws using the MCMC algorithm.
- We know, from the law of large numbers, that the sample mean converges on the actual mean.
- so as **the number of draws in the MCMC sample**, $R \to \infty$,

$$\frac{\sum_{i=1}^{R} \theta_j^{(i)}}{R} \to E(\theta_j|D),$$

where D is the data sample used to obtain the posterior distribution.

Treating the MCMC draws as a random sample

- Further, we can obtain **as large an MCMC sample as we like** just by making more draws using the MCMC algorithm.
- We know, from the law of large numbers, that the sample mean converges on the actual mean.
- so as **the number of draws in the MCMC sample**, $R \to \infty$,

$$\frac{\sum_{i=1}^{R} \theta_j^{(i)}}{R} \to E(\theta_j|D),$$

where D is the data sample used to obtain the posterior distribution.

- So we can just calculate means, standard deviations, etc. in the usual way because **we have a sample drawn from the posterior distribution** of θ_j.

Source of confusion: the data sample size vs. the MCMC sample size

- As **the number of observations in the data sample**, $n \to \infty$, the posterior distribution will collapse upon the true parameter value, i.e.

$$E(\theta_j|D) \to \theta$$

and

$$\mathrm{Var}(\theta_j|D) \to 0$$

The data sample size vs. the MCMC sample size (cont.)

- whereas as **the number of draws in the MCMC sample**, $R \to \infty$,

$$\bar{\theta} = \frac{\sum_{i=1}^{R} \theta_j^{(i)}}{R} \to E(\theta_j|D) \neq \theta,$$

and

$$\frac{\sum_{i=1}^{R} (\theta_j^{(i)} - \bar{\theta})^2}{R} \to \mathrm{Var}(\theta_j|D) \neq 0,$$

where D is the data sample used to obtain the posterior distribution.

The MCMC chain as a sample of observations

- We use the output of the MCMC algorithm as a sample of observations for the unknown quantity of interest

The MCMC chain as a sample of observations

- We use the output of the MCMC algorithm as a sample of observations for the unknown quantity of interest
- It can be shown under remarkably general conditions, that the MCMC chain converges to draws from the marginal posterior distribution for each variable in the chain.

The MCMC chain as a sample of observations

- We use the output of the MCMC algorithm as a sample of observations for the unknown quantity of interest
- It can be shown under remarkably general conditions, that the MCMC chain converges to draws from the marginal posterior distribution for each variable in the chain.
- That is, by drawing iteratively from the conditional posteriors for θ_1 and θ_2, given by $p(\theta_1|\theta_2, y)$ and $p(\theta_2|\theta_1, y)$, we obtain a sample for each that approximates values from the marginal posteriors $p(\theta_1|y)$ and $p(\theta_2|y)$.

The MCMC chain as a sample of observations

- We use the output of the MCMC algorithm as a sample of observations for the unknown quantity of interest
- It can be shown under remarkably general conditions, that the MCMC chain converges to draws from the marginal posterior distribution for each variable in the chain.
- That is, by drawing iteratively from the conditional posteriors for θ_1 and θ_2, given by $p(\theta_1|\theta_2, y)$ and $p(\theta_2|\theta_1, y)$, we obtain a sample for each that approximates values from the marginal posteriors $p(\theta_1|y)$ and $p(\theta_2|y)$.
- The longer we run the chain, the closer the approximation, so, since **we choose *R***, the number of MCMC draws, we can get **arbitrarily close** by running the chain for long enough.

MCMC convergence

- Typically, especially for the more standard, well understood models, the MCMC converges reasonably quickly to the marginal posteriors.

MCMC convergence

- Typically, especially for the more standard, well understood models, the MCMC converges reasonably quickly to the marginal posteriors.
- We drop the first part of the chain, called the 'burn-in period' because it is likely that the chain hasn't converged at the beginning of the process

MCMC convergence

- Typically, especially for the more standard, well understood models, the MCMC converges reasonably quickly to the marginal posteriors.
- We drop the first part of the chain, called the 'burn-in period' because it is likely that the chain hasn't converged at the beginning of the process
- This allows some time for the chain to converge, so that the approximation is better (though theoretically we can keep the entire sample).

MCMC convergence

- Typically, especially for the more standard, well understood models, the MCMC converges reasonably quickly to the marginal posteriors.
- We drop the first part of the chain, called the 'burn-in period' because it is likely that the chain hasn't converged at the beginning of the process
- This allows some time for the chain to converge, so that the approximation is better (though theoretically we can keep the entire sample).
- We can evaluate convergence by plotting the chain, looking at numerical standard errors, etc.

MCMC convergence

- Typically, especially for the more standard, well understood models, the MCMC converges reasonably quickly to the marginal posteriors.
- We drop the first part of the chain, called the 'burn-in period' because it is likely that the chain hasn't converged at the beginning of the process
- This allows some time for the chain to converge, so that the approximation is better (though theoretically we can keep the entire sample).
- We can evaluate convergence by plotting the chain, looking at numerical standard errors, etc.
- For example, **bayesm** in R has a function to calculate n_{eff}, etc., called numEff.

Using the MCMC sample

- We can thus use the MCMC sample as we would a sample of observations on any observable.

Using the MCMC sample

- We can thus use the MCMC sample as we would a sample of observations on any observable.
- We can calculate mean, median, mode, standard deviation, confidence intervals, etc.

Using the MCMC sample

- We can thus use the MCMC sample as we would a sample of observations on any observable.
- We can calculate mean, median, mode, standard deviation, confidence intervals, etc.
- A frequency plot of the sample (histogram) is an empirical estimate of the entire marginal posterior density.

Using the MCMC sample

- We can thus use the MCMC sample as we would a sample of observations on any observable.
- We can calculate mean, median, mode, standard deviation, confidence intervals, etc.
- A frequency plot of the sample (histogram) is an empirical estimate of the entire marginal posterior density.
- The more observations we have, the more accurate the estimates.

Using the MCMC sample

- We can thus use the MCMC sample as we would a sample of observations on any observable.
- We can calculate mean, median, mode, standard deviation, confidence intervals, etc.
- A frequency plot of the sample (histogram) is an empirical estimate of the entire marginal posterior density.
- The more observations we have, the more accurate the estimates.
- We can typically run an MCMC chain for tens of thousands of iterations very quickly.

Using the MCMC sample

- We can thus use the MCMC sample as we would a sample of observations on any observable.
- We can calculate mean, median, mode, standard deviation, confidence intervals, etc.
- A frequency plot of the sample (histogram) is an empirical estimate of the entire marginal posterior density.
- The more observations we have, the more accurate the estimates.
- We can typically run an MCMC chain for tens of thousands of iterations very quickly.
- If we require higher numerical accuracy, we run the algorithm for longer.

Simulation Based Inference and MCMC

- What are you doing when you look up values in a statistical table? Answer: numerical integration!

Simulation Based Inference and MCMC

- What are you doing when you look up values in a statistical table? Answer: numerical integration!
- Only very simple problems could be dealt with in this way (from a numerical perspective - they were often very complicated analytically)

$$E(g(X)|D) = \int g(X)p(X|D)dX, \tag{5}$$

where we assume that the moment exists and D represents the conditioning information available (data and other 'prior' information).

Simulation Based Inference and MCMC

- What are you doing when you look up values in a statistical table? Answer: numerical integration!
- Only very simple problems could be dealt with in this way (from a numerical perspective - they were often very complicated analytically)

$$E(g(X)|D) = \int g(X)p(X|D)dX, \tag{5}$$

where we assume that the moment exists and D represents the conditioning information available (data and other 'prior' information).

- Every summary of $p(X|D)$ can be formulated as a moment $E(g(X)|D)$. The CDF $= F(x) = Pr(X \leq x) = E(I(X \leq x)$, where $I()$ is the indicator function, $E(X) =$ the mean, $E(X^2) - (E(X))^2 =$ the variance, etc.

Using the law of large numbers and CLT to our advantage

- If $X^{(1)}, \ldots, X^{(R)}$ are simulated to produce *iid* realizations from $p(X|D)$, then by the strong law of large numbers, this can be approximated by

$$E(g(\hat{X})|D) = (1/R) \sum_{r=1}^{R} g(X^{(r)}), \tag{6}$$

where **the approximation improves as R increases**.

Using the law of large numbers and CLT to our advantage

- If $X^{(1)}, \ldots, X^{(R)}$ are simulated to produce *iid* realizations from $p(X|D)$, then by the strong law of large numbers, this can be approximated by

$$E(g(\hat{X})|D) = (1/R) \sum_{r=1}^{R} g(X^{(r)}), \tag{6}$$

  where **the approximation improves as R increases**.
- We can characterize the distribution of the approximation using the central limit theorem.

Using the law of large numbers and CLT to our advantage

- If $X^{(1)}, \ldots, X^{(R)}$ are simulated to produce *iid* realizations from $p(X|D)$, then by the strong law of large numbers, this can be approximated by

$$E(g(\hat{X})|D) = (1/R)\sum_{r=1}^{R} g(X^{(r)}), \tag{6}$$

  where **the approximation improves as R increases**.
- We can characterize the distribution of the approximation using the central limit theorem.
- The use of (6), with large R, to approximate (5) replaces analytical derivations of $g(.)$.

Using the law of large numbers and CLT to our advantage

- If $X^{(1)}, \ldots, X^{(R)}$ are simulated to produce *iid* realizations from $p(X|D)$, then by the strong law of large numbers, this can be approximated by

$$E(g(\hat{X})|D) = (1/R)\sum_{r=1}^{R} g(X^{(r)}), \tag{6}$$

  where **the approximation improves as R increases**.
- We can characterize the distribution of the approximation using the central limit theorem.
- The use of (6), with large R, to approximate (5) replaces analytical derivations of $g(.)$.
- Approximations to the examples above are straightforward to compute: $\hat{F(x)} = E(I(\hat{X} \leq x)$, mean $= E(\hat{X})$, etc.

The Gibbs sampler

- The simplest MCMC, both to describe and to implement, is the Gibbs sampler (GS).

The Gibbs sampler

- The simplest MCMC, both to describe and to implement, is the Gibbs sampler (GS).
- Using the GS, we simulate successively from the conditional probability distributions.

The Gibbs sampler

- The simplest MCMC, both to describe and to implement, is the Gibbs sampler (GS).
- Using the GS, we simulate successively from the conditional probability distributions.
- Suppose we have two sets of parameters, say covariate coefficients, θ and variance matrix, Ω.

The Gibbs sampler

- The simplest MCMC, both to describe and to implement, is the Gibbs sampler (GS).
- Using the GS, we simulate successively from the conditional probability distributions.
- Suppose we have two sets of parameters, say covariate coefficients, θ and variance matrix, Ω.
- For $r = 1, \ldots R$, we choose starting values $\theta^{(0)}$ and $\Omega^{(0)}$ and sample iteratively from

$$p(Y^{(r)}|\theta^{(r-1)}, \Omega^{(r-1)}, Z),$$

$$p(\theta^{(r)}|, Y^{(r)}\Omega^{(r-1)}, Z),$$

$$p(\Omega^{(r)}|Y^{(r)}, \theta^{(r)}, Z).$$

Output from the Gibbs sampler

- At each step , the latest values obtained from the previous steps are used in the conditioning arguments. This defines a Markov chain with stationary distribution that is the posterior distribution $p(Y, \theta, \Omega | Z)$.

Output from the Gibbs sampler

- At each step , the latest values obtained from the previous steps are used in the conditioning arguments. This defines a Markov chain with stationary distribution that is the posterior distribution $p(Y, \theta, \Omega | Z)$.
- The final result is a simulation of R observations for each of the elements of Y, θ and Ω.

Output from the Gibbs sampler

- At each step , the latest values obtained from the previous steps are used in the conditioning arguments. This defines a Markov chain with stationary distribution that is the posterior distribution $p(Y, \theta, \Omega|Z)$.
- The final result is a simulation of R observations for each of the elements of Y, θ and Ω.
- Hence the optimal predictor $E(Y|Z)$ and marginal posteriors $p(\theta|Z)$, $p(\Omega|Z)$, etc. can be approximated by (6).

Output from the Gibbs sampler

- At each step , the latest values obtained from the previous steps are used in the conditioning arguments. This defines a Markov chain with stationary distribution that is the posterior distribution $p(Y, \theta, \Omega|Z)$.
- The final result is a simulation of R observations for each of the elements of Y, θ and Ω.
- Hence the optimal predictor $E(Y|Z)$ and marginal posteriors $p(\theta|Z)$, $p(\Omega|Z)$, etc. can be approximated by (6).
- Another measure of uncertainty is the Bayesian credible interval [see CW, p.47].

MCMC with Normal-gamma prior

- Likelihood for n observations

$$y \sim N_n(X\beta, \sigma^2 I_n)$$

MCMC with Normal-gamma prior

- Likelihood for n observations

$$y \sim N_n(X\beta, \sigma^2 I_n)$$

- Prior distribution for β **independent of** σ^2:

$$\beta \sim N_k(\beta_0, V_0)$$

MCMC with Normal-gamma prior

- Likelihood for n observations

$$y \sim N_n(X\beta, \sigma^2 I_n)$$

- Prior distribution for β **independent of** σ^2:

$$\beta \sim N_k(\beta_0, V_0)$$

- Prior for the precision τ,

$$\tau \sim Gamma(v_0/2, \delta_0/2)$$

or, if you prefer to specify the variance σ^2,

$$\sigma^2 \sim IG(v_0/2, \delta_0/2).$$

MCMC with Normal-gamma prior

- Likelihood for n observations

$$y \sim N_n(X\beta, \sigma^2 I_n)$$

- Prior distribution for β **independent of** σ^2:

$$\beta \sim N_k(\beta_0, V_0)$$

- Prior for the precision τ,

$$\tau \sim Gamma(v_0/2, \delta_0/2)$$

or, if you prefer to specify the variance σ^2,

$$\sigma^2 \sim IG(v_0/2, \delta_0/2).$$

- Notation: some authors use δ_0, others use $n_0 S_0$ or some other notation.

Conditional posterior densities

$$\beta|\sigma^2, y, X \sim N_k(\beta_1, V_1),$$

$$\sigma^2|\beta, y, X \sim IG(v_1/2, \delta_1/2),$$

where

$$V_1 = (\sigma^{-2}X'X + V_0^{-1})^{-1},$$

$$\beta_1 = V_1(\sigma^{-2}X'y + V_0^{-1}\beta_0),$$

$$v_1 = v_0 + n,$$

$$\delta_1 = \delta_0 + (y - X'\beta)'(y - X'\beta).$$

MCMC Gibbs algorithm

- This leads to the following MCMC (Gibbs) algorithm [Greenberg Algorithm 8.1, p.112]

MCMC Gibbs algorithm

- This leads to the following MCMC (Gibbs) algorithm [Greenberg Algorithm 8.1, p.112]
- a) Choose a starting value $\sigma^{2(0)}$,

MCMC Gibbs algorithm

- This leads to the following MCMC (Gibbs) algorithm [Greenberg Algorithm 8.1, p.112]
- a) Choose a starting value $\sigma^{2(0)}$,
- b) At the gth iteration, draw

$$\beta^{(g)}|\tau\sigma^2 \sim N_k(\beta_1^{(g)}, V_1^{(g)}),$$

$$\sigma^{2(g)}|\beta \sim IG(v_1/2, \delta_1^{(g)}/2).$$

MCMC Gibbs algorithm

- This leads to the following MCMC (Gibbs) algorithm [Greenberg Algorithm 8.1, p.112]
- a) Choose a starting value $\sigma^{2(0)}$,
- b) At the gth iteration, draw

$$\beta^{(g)}|\tau\sigma^2 \sim N_k(\beta_1^{(g)}, V_1^{(g)}),$$

$$\sigma^{2(g)}|\beta \sim IG(v_1/2, \delta_1^{(g)}/2).$$

- c) Repeat step b) until $g = B + G$ where B is the burn-in sample and G is the desired sample size.

Conditional distributions parameters

$$V_1^{(g)} = (\sigma^{-2(g-1)} X'X + V_0^{-1})^{-1},$$

$$\beta_1^{(g)} = V_1(\sigma^{-2(g-1)} X'y + V_0^{-1}\beta_0),$$

$$v_1 = v_0 + n,$$

$$\delta_1^{(g)} = \delta_0 + (y - X'\beta^{(g)})'(y - X'\beta^{(g)}).$$

Conditional distributions parameters

$$V_1^{(g)} = (\sigma^{-2(g-1)} X'X + V_0^{-1})^{-1},$$

$$\beta_1^{(g)} = V_1(\sigma^{-2(g-1)} X'y + V_0^{-1}\beta_0),$$

$$v_1 = v_0 + n,$$

$$\delta_1^{(g)} = \delta_0 + (y - X'\beta^{(g)})'(y - X'\beta^{(g)}).$$

- When using R or Matlab (or a different package), care must be taken to set the correct value for δ_0 and δ_1 when drawing from the Gamma distribution using rgamma or gam_rnd.

MCMC Gibbs sampler examples

- **Bayesregexample.R** provides examples using the bayesm in R.
- **linregmcmceg1.R** does essentially the same as above, just a shorter version.
- **mcmclinreg.R** is a "DIY" version to demonstrate the nuts and bolts of the algorithm, i.e. what's going on 'under the hood' (too many idioms?).
- **Koop-hprice-ch4.R** uses the data HPRICE.txt to reproduce the house prices example in Koop, chapter 4
- **chapter4a.m** is the matlab code from the Koop website to reproduce the above example.
- More examples with heteroskedastic errors below.

GLM transformation (reminder)

- In general, for $y = X\beta + e$, $e \sim N(0, V)$, using the transformation with $\Lambda'\Lambda = V^{-1}$ and we premultiplying the regression model we get,

$$\Lambda y = \Lambda X\beta + \Lambda e,$$

$$y_\lambda = X_\lambda \beta + e_\lambda.$$

$$e_\lambda | X, \beta, V, \sigma^2 \sim N(0, \sigma^2 I_n).$$

Bayesian inference for GLM

- With a uniform prior for β and $\ln\sigma^2$, the conditional posterior density is normal with mean and variance = the GLS estimators,

$$\hat{\beta}_{GLS} = (X_\lambda' X_\lambda)^{-1} X_\lambda' y_\lambda == (X' V^{-1} X)^{-1} X' V^{-1} y.$$

$$\mathrm{var}_{GLS}(\beta) = \sigma^2 (X_\lambda' X_\lambda)^{-1} = \sigma^2 (X' V^{-1} X)^{-1}$$

Bayesian inference for GLM

- With a uniform prior for β and $\ln\sigma^2$, the conditional posterior density is normal with mean and variance = the GLS estimators,

$$\hat{\beta}_{GLS} = (X_\lambda' X_\lambda)^{-1} X_\lambda' y_\lambda == (X' V^{-1} X)^{-1} X' V^{-1} y.$$

$$\mathrm{var}_{GLS}(\beta) = \sigma^2 (X_\lambda' X_\lambda)^{-1} = \sigma^2 (X' V^{-1} X)^{-1}$$

- Similarly, conditional posterior for σ^2 given V is

$$\sigma^2 | V \sim \mathrm{Gamma}(\nu/2, (y^\lambda - X^\lambda \hat{\beta}_{GLS})'(y^\lambda - X^\lambda \hat{\beta}_{GLS})/2).$$

Conditional posterior for V

- The conditional posterior for V given β and σ^2 and a prior, $p(V)$, has the form

$$p(V|y, X, \beta, \sigma^2) \propto p(V)|V|^{-1/2} \exp\left[-\frac{1}{2\sigma^2}(y - X\beta)'V^{-1}(y - X\beta)\right].$$

Conditional posterior for V

- The conditional posterior for V given β and σ^2 and a prior, $p(V)$, has the form

$$p(V|y, X, \beta, \sigma^2) \propto p(V)|V|^{-1/2} \exp\left[-\frac{1}{2\sigma^2}(y - X\beta)'V^{-1}(y - X\beta)\right].$$

- Given a uniform prior for the elements of $\alpha = (\alpha_1, \alpha_2)'$, in $\sigma_i^2 = \exp(\alpha_1 + \alpha_2 z_i)$, prior becomes part of the normalizing constant, and $|V| = \prod_{i=1}^n \sigma_i^2$. The above conditional is not in a known/convenient distributional form from which to draw, so a MH step is required.

Some alternative specifications

- Alternatively, to simplify a little, we can drop the scaling factor in the variance, σ^2, i.e. specify $e \sim N(0, V)$ instead of $e \sim N(0, \sigma^2 V)$, eliminating the need to draw from $\sigma^2 | V$.

Some alternative specifications

- Alternatively, to simplify a little, we can drop the scaling factor in the variance, σ^2, i.e. specify $e \sim N(0, V)$ instead of $e \sim N(0, \sigma^2 V)$, eliminating the need to draw from $\sigma^2 | V$.
- Another alternative that allows GS at all steps is to draw each σ_i^2 independently from a Gamma and construct the diagonal V matrix at each iteration. It would be interesting to evaluate how each of these alternatives performs.

Random walk MH step

- Using a random walk MH algorithm to obtain draws, the above conditional is evaluated each MCMC round, r,

$$p(V^{(r)}|\beta^{(r)}, \sigma^{2(r)}, y, X),$$

and the previous and the candidate draw are used to calculate the acceptance probability as follows.

Random walk HM algorithm

- The idea is to take a random step away from the current location, drawing

$$V^{(r)} = V^{(r-1)} + e, \quad e \sim N(0, s^2\Sigma)$$

where s^2 is the tuning parameter chosen to obtain an acceptance rate in the range 0.3 to 0.5, and either $\Sigma = I$ or $\Sigma =$ an approximate estimate of the posterior covariance matrix, then

Random walk HM algorithm

- The idea is to take a random step away from the current location, drawing

$$V^{(r)} = V^{(r-1)} + e, \quad e \sim N(0, s^2\Sigma)$$

where s^2 is the tuning parameter chosen to obtain an acceptance rate in the range 0.3 to 0.5, and either $\Sigma = I$ or $\Sigma =$ an approximate estimate of the posterior covariance matrix, then

- Evaluate the conditional posterior at the old and new locations and accept the new draw with probability

$$p(accept) = \min\{1, \frac{p(V^{(r)}|\beta^{(r)}, \sigma^{2(r)}, y, X)}{p(V^{(r-1)}|\beta^{(r)}, \sigma^{2(r)}, y, X)}.$$

MCMC for GLM with heteroskedasticity of known form

- For the model with the ith diagonal element of V given by $\sigma_i^2 = \exp(\alpha_1 + \alpha_2 z_i)$

```
# Initial values x = X
ols   = lm(y~x-1)
b     = ols$coef
e     = y-x%*%b
q <- log(e^2)
z <- cbind(rep(1,n),x2)
a <- solve(t(z)%*%z)%*%t(z)%*%q

# MCMC
sd = 0.1; burn = 1000; R = 20000; niter = burn+R
bs = matrix(0,niter,p); as = matrix(0,niter,q)

for (iter in 1:niter){
```

MCMC updating draws

```
# new mean and variance for b distribution using GLS tra
A  = exp(z%*%a/2)
y1 = y/A
x1 = x/matrix(A,n,p)
V1 = solve(iV0+t(x1)%*%x1)
b1 = V1%*%(iV0b0+t(x1)%*%y1)

# GS update for regression coefficients b
b  = b1+t(chol(V1))%*%rnorm(p)
```

MH step

```
# MH step for variance coefficients a
e  = y-x%*%b
a1 = rnorm(q,a,sd)
la = sum(dnorm(e,0,exp(z%*%a1/2),log=TRUE))
          -sum(dnorm(e,0,exp(z%*%a/2),log=TRUE))
if (log(runif(1))<la){a = a1}

bs[iter,]   = t(b)
as[iter,]   = t(a)
}
# final MCMC sample
bs = bs[(burn+1):R,]
as = as[(burn+1):R,]
```

- see **heterosked1.R**

Heteroskedasticity of unknown form

- Now suppose we do not know the form of the heteroskedasticity.
- If we are willing to assume instead that the error variances, σ_i^2 are drawn from a common distribution, the Gamma, i.e.

$$p(\sigma_i^2|v) = \prod_{i=1}^{n} \text{Gamma}\left(\frac{v}{2}, \frac{v}{2}\right).$$

Heteroskedasticity of unknown form

- Now suppose we do not know the form of the heteroskedasticity.
- If we are willing to assume instead that the error variances, σ_i^2 are drawn from a common distribution, the Gamma, i.e.

$$p(\sigma_i^2|v) = \prod_{i=1}^{n} \text{Gamma}\left(\frac{v}{2}, \frac{v}{2}\right).$$

- Two approaches:

1. Choose $2 < v < 30$. The closer to 30, the closer to homoskedasticity. A v of around 5 appears to be a reasonable choice (then explore what happens with different values for v).

Heteroskedasticity of unknown form

- Now suppose we do not know the form of the heteroskedasticity.
- If we are willing to assume instead that the error variances, σ_i^2 are drawn from a common distribution, the Gamma, i.e.

$$p(\sigma_i^2|\nu) = \prod_{i=1}^{n} \text{Gamma}\left(\frac{\nu}{2}, \frac{\nu}{2}\right).$$

- Two approaches:

1. Choose $2 < \nu < 30$. The closer to 30, the closer to homoskedasticity. A ν of around 5 appears to be a reasonable choice (then explore what happens with different values for ν).
2. Adopt a hierarchical prior for ν. The exponential distribution is the usual choice (see Geweke, 1993, and Koop, Poirier and Tobias (2007),

$$p(\nu) = \text{Exp}(\nu, 2) = \frac{1}{2}\exp(-\nu/2).$$

Student-t errors

- If we take either approach, we obtain a neat result - Geweke (1993) shows that with this set up, if we integrate out the σ_i^2s, we get a Student-t error linear regression model with homoskedastic error (i.e. Student-t instead of Normal errors).

Student-t errors

- If we take either approach, we obtain a neat result - Geweke (1993) shows that with this set up, if we integrate out the σ_i^2s, we get a Student-t error linear regression model with homoskedastic error (i.e. Student-t instead of Normal errors).
- Equations (13.8), (13.12), (13.20) and (13.22) in Koop *et al.* (2007) (ch. 13) provide the full set of conditional posteriors for this model.

Student-t errors

- If we take either approach, we obtain a neat result - Geweke (1993) shows that with this set up, if we integrate out the σ_i^2s, we get a Student-t error linear regression model with homoskedastic error (i.e. Student-t instead of Normal errors).
- Equations (13.8), (13.12), (13.20) and (13.22) in Koop *et al.* (2007) (ch. 13) provide the full set of conditional posteriors for this model.
- We can draw sequentially from each of these conditionals to obtain an MCMC sample and conduct inference.

Student-t errors

- If we take either approach, we obtain a neat result - Geweke (1993) shows that with this set up, if we integrate out the σ_i^2s, we get a Student-t error linear regression model with homoskedastic error (i.e. Student-t instead of Normal errors).
- Equations (13.8), (13.12), (13.20) and (13.22) in Koop *et al.* (2007) (ch. 13) provide the full set of conditional posteriors for this model.
- We can draw sequentially from each of these conditionals to obtain an MCMC sample and conduct inference.
- An MH step is required to draw from (13.22) since the density is nonstandard. The other distributions are all of known form and so a Gibbs step can be employed. See Koop *et al.* (2007) for further details.
- see **glmterrs.R**

A final word of warning

- If you explore uncharted territory:

A final word of warning

- If you explore uncharted territory:
- "When developing MCMC algorithms to simulate from the posterior distribribution, [it is important to] take great care in developing full conditional distributions, in looking for ways to increase statistical efficiency (e.g. Rao-Blackwellization) and computation efficiency (e.g. block updating), and in diagnosing the convergence of the Markov chain to its stationary distribution." [Cressie and Wilke (2011), p.48]. See also Gelman et al. (2004) and Robert and Casella (2004).

A final word of warning

- If you explore uncharted territory:
- "When developing MCMC algorithms to simulate from the posterior distribribution, [it is important to] take great care in developing full conditional distributions, in looking for ways to increase statistical efficiency (e.g. Rao-Blackwellization) and computation efficiency (e.g. block updating), and in diagnosing the convergence of the Markov chain to its stationary distribution." [Cressie and Wilke (2011), p.48]. See also Gelman et al. (2004) and Robert and Casella (2004).
- Don't reinvent the wheel though - robust, efficient algorithms for almost any model you wish to analyze has, in all likelihood, already been developed.