Markov Chain Monte Carlo and Applied Bayesian Statistics: *a short course*

Chris Holmes

Professor of Biostatistics
Oxford Centre for Gene Function

Objectives of Course

- To introduce the Bayesian approach to statistical data modelling
- To discuss Markov chain Monte Carlo (MCMC), a stochastic simulation technique that is extremely useful for computing inferential quantities.
- To introduce the software package “WinBugs”, a tool for setting up Bayesian models and performing inference via MCMC

Key References:

Gelman, A. *et al*. *Bayesian Data Analysis. 2nd Ed*. (2004). Chapman & Hall

Robert, C. P. and Casella, G. *Monte Carlo Statistical Methods*. (2004/1999). Springer

Gilks, W. R. *et al*. eds. *Markov chain Monte Carlo in Practice*. (1996). Chapman & Hall.

Acknowledgements:

Nicky Best for WinBugs help and examples.

1 Introduction and extended overview

Bayesian methods are becoming increasingly popular as techniques for modelling “systems”.

At the heart of Bayesian procedures is the following philosophy:

Bayesian inference is about the quantification and propagation of uncertainty, defined via a probability, in light of observations of the system. From Prior $\longrightarrow$ Posterior.

This is fundamentally different to classical inference which tends to be concerned with parameter *estimation*

Most classical models can be cast in a Bayesian framework, for example, normal linear regression, ARMA, GLMs, etc

Aspects of Bayes that people like

- Uncertainty propagation
  - Quantify *all* aspects of uncertainty via **probability**.
  - **Probability** is the central tool.
- Axiomatic
  - Bayesian statistics has an axiomatic foundation, from which all procedures then follow.
  - It is prescriptive in telling you how to act coherently
  - “If you don’t adopt a Bayesian approach you must ask yourself which of the axioms are you prepared to discard?” *See disadvantages*.
- Empirical evidence of effectiveness

  - There is mounting evidence that Bayesian procedures often lead to more accurate models, in terms of predictive performance, than non-Bayes approaches.
  - Especially true for complex (highly parameterised) models.
- Unified framework.
  - Random effects, Hierarchical models, Missing variables, Nested and Non-nested models. All handled in the same framework.
- Intuitive
  - For a model parameterised by θ, what is the interpretation of $Pr(\theta < 0|y)$, where y denotes “data”?
  - Compare with confidence intervals for θ
  - When testing a null hypothesis H_0 against an alternative H_1, in

Bayesian statistics we find

$$Pr(H_0|Data) = 1 - Pr(H_1|Data)$$

so that if your null became your alternative their is symmetry.

This is not true of frequentist testing using p-values

Some aspects that people don't like

- Using prior probability
  - Bayes requires a joint distribution on observable, y, AND, parameters, θ.
  - But where does this knowledge come from? And how can we expect non-technical users to formulate probability statements?
- Inference is *subjective*
  - You and I on observing the same data will be lead to different conclusions
  - This seems to fly in the face of scientific reasoning
- Bayesian inference is a closed hypothesis space.
  - Broadly speaking, there is no official room for model checking or

validation. Your prior is your prior.

- Hence, you can never learn about models (hypotheses) outside of your prior (which was specified before the data arrived)

Pragmatically there is room for both Bayesian and non-Bayes procedures, and using both often leads to more informative conclusions about the data.

1.1 Bayesian data analysis

Broadly speaking there are three steps to Bayesian data analysis

1. Setting up of a full *joint* probability distribution for both observable, y, and parameters, θ,

$$p(y, \theta) = p(y|\theta)p(\theta)$$

2. Conditioning on data, $p(\theta|y)$

3. Model checking

   – Note, this is not consistent with purist Bayesians

1.2 Bayes Theorem

In what follows, we shall use y to denote a $n \times 1$ vector of *observed* values of a system and let θ denote some *unobserved* parameter(s) for the model of the system.

We will assume that the data are *exchangeable*, in that,

$$p(y_1, \ldots, y_n)$$

is invariant to permutations of the indecies. This is a key assumption.

In BDA almost all inference is made using probability statements, regarding $\theta|y$ or $\tilde{y}|y$, for future unobserved $\tilde{y}$

To begin we specify a *joint* model, $p(y, \theta)$ which is factorised as,

$$p(y, \theta) = p(y|\theta)p(\theta)$$

where $p(y|\theta)$ is the sampling distribution and $p(\theta)$ is your prior

- the prior $p(\theta)$ elicits a model space and a distn on that model space via a distn on parameters
- reflects beliefs about dependence structures in the data

Interested in conditioning on observable, $\theta|y$, which follows

$$\begin{aligned} p(\theta|y) &= \frac{p(y,\theta)}{p(y)} \\ &= \frac{p(y|\theta)p(\theta)}{p(y)} \end{aligned}$$

BAYES THEOREM

$p(\theta|y)$ contains all of the information combining prior knowledge and observations

1.3 Three key quantities of Interest

There are 3 key quantities that we are often interested in

(1) **Prior predictive**, $p(y)$

The normalising constant in Bayes Theorem, $p(y)$, is a very important quantity,

$$p(y) = \int p(y, \theta) d\theta = \int p(y|\theta) p(\theta) d\theta$$

It represents the "evidence" for a particular model, defined by $p(\theta)$

Known as the *prior predictive* as it represents the probability of observing the data that was observed before it was observed

Also known as the *evidence* or *marginal likelihood*

(2) **Marginal effects** of a subset of parameters in a multivariate model

Let $\theta = (\theta_1, \ldots, \theta_p)$ denote a p dimensional model

Suppose we are interested in $p(\theta_i|y)$, for some subset $\theta_i \in \theta$.

Then,

$$\begin{aligned} p(\theta_i|y) &= \int p(\theta_i, \theta_{-i}|y) d\theta_{-i} \\ &= \int p(\theta_i|\theta_{-i}, y) p(\theta_{-i}|y) d\theta_{-i} \end{aligned}$$

where $\theta_{-i} = \theta \backslash \theta_i$ denotes the vector θ with θ_i removed

(3) **Posterior Predictions**

Let $\tilde{y}$ denote some future unobserved response of the system.

Then the posterior predictive $p(\tilde{y}|y)$ follows,

$$\begin{aligned} p(\tilde{y}|y) &= \int p(\tilde{y}|\theta, y)p(\theta|y)d\theta \\ &= \int p(\tilde{y}|\theta)p(\theta|y)d\theta \end{aligned}$$

Note that $\tilde{y}, y$ are conditionally independent given θ; though clearly $p(\tilde{y}, y)$ are dependent

Again, note that all 3 quantities are defined via probability statements on the unknown variable of interest

Example: Normal Linear Regression

Consider a normal linear regression,

$$y = x\beta + \epsilon$$

where ϵ $N(0, \sigma^2 I)$. Alternatly,

$$y \sim N(y|x\beta, \sigma^2 I)$$

for now assume that σ is known

Classically, we would wish to *estimate* the regression coefficients, β, given a data set, $\{y_i, x_i\}_{i=1}^n$, say using MLE

$$\hat{\beta} = (x'x)^{-1}x'y$$

Bayesian modelling proceeds by constructing a joint model for the

data and unknown parameters,

$$\begin{aligned} p(y, \beta | x, \sigma^2) &= p(y|x, \beta, \sigma^2) p(\beta | x, \sigma^2) \\ &= N(y|x\beta, \sigma^2 I) p(\beta) \end{aligned}$$

where we assume, for now, that the prior $p(\beta)$ is independent of $\{x, \sigma^2\}$

Suppose we take,

$$p(\beta) = N(\beta | 0, v)$$

Then,

$$\begin{aligned} p(\beta|y) &\propto p(y|\beta)p(\beta) \\ &\propto \sigma^{-n/2}\exp[-\frac{1}{2\sigma^2}(y-x\beta)'(y-x\beta)] \times \\ &\quad |v|^{-1/2}\exp[-\beta' v\beta] \end{aligned}$$

which can be written,

$$\begin{aligned} p(\beta|y) &= N(\beta|\hat{\beta},\hat{v}) \\ \hat{\beta} &= (x'x+v^{-1})^{-1}x'y \\ \hat{v} &= (x'x+v^{-1})^{-1} \end{aligned}$$

Note

- β again follows a normal distribution: Prior (normal) $\longrightarrow$ Posterior (normal)

For new data, $\{y_0, x_0\}$, predictive densities follow,

$$\begin{aligned} p(y_0|x_0, y) &= \int p(y_0|x_0, \beta, y)p(\beta|y)d\beta \\ &= \int N(y_0|x_0\beta, \sigma^2)N(\beta|\hat{\beta}, \hat{v})d\beta \\ &= N(y_0|x_0\hat{\beta}, \sigma^2(1 + x_0\hat{v}x_0')) \end{aligned}$$

1.4 Prior: where does $p(\theta)$ come from?

- Formally it represents your subjective beliefs, via a probability statement, about likely values of unobserved θ before you've observed y
- Practically, there are often standard and well used forms for the set $\{p(y|\theta), p(\theta)\}$
- In the example above the choice of $p(\beta) = N(\beta|0, v)$ lead to easy (closed form) calculations of $p(\beta|y)$ and $p(\tilde{y}|y)$

- Often we will use forms for $p(\theta)$ that depend on $p(y|\theta)$ and which make these calculations easy
  - when the *prior* and the *posterior* are from the same family of distributions (such as normal-normal) then the prior is termed conjugate
  - Note, from a purest perspective this is putting the cart before the horse

1.5 Computational tools and MCMC

- In Section **??** all three major quantities of interest required integration over a possibly high dimension parameter space $\theta \in \Theta$
- Indeed, in BDA *integration* is the principle inferential operation; as opposed to *optimisation* in classical statistics
- Historically, the need to evaluate integrals was a major stumbling block for the take up of Bayesian methods
- Severely restricts the type of models, $p(y, \theta)$, that could be implemented
- Then, around 15 years ago a numerical technique known as

Markov chain Monte Carlo (**MCMC**) was popularised by a paper of Gelfand & Smith (1990)

- G&S (1990) is one the top three most cited papers in mathematics in the last 20 years
- other statisticians, for example, (Ripley, Besag, Tanner, Geman), were using MCMC before this

MCMC: what's the big deal?

It is fair to say that MCMC has revitalised (perhaps even revolutionised) Bayesian statistics. Why?

- MCMC is a general method that simulataniously solves inference of $\{p(\theta|y), p(\theta_i|y), p(\tilde{y}|y)\}$
- Only requires evaluation of joint distn.,

$$p(y, \theta) \propto p(y|\theta)p(\theta)$$

  up to proportionality, pointwise for any $\theta \in \Theta$
- Allows modeller to concentrate on modelling. That is, to use models, $p(y, \theta)$, that you believe represent the true dependence structures in the data, rather than those that are simple to compute

How does it work?

- MCMC methods construct a Markov chain on the state space, $\theta \in \Theta$, whose steady state distribution is the posterior distn. of interest $p(\theta|y)$ - this sounds hard, however,
- There are simple and general procedures for constructing Markov chains to (automatically) match **any** $p(\theta|y)$
- MCMC simulation approximates the true posterior density, $p(\theta|y)$, using a bag of samples drawn from the density...

- That is, MCMC procedures return a collection of M samples, $\{\theta^{(1)}, \ldots, \theta^{(M)}\}$ where each sample can be assumed to be drawn from $p(\theta|y)$, (with slight abuse of notation....)

$$Pr(\theta^{(i)} \in A) = p(\theta \in A|y)$$

for any set $A \in \Theta$, or,

$$\theta^{(i)} \sim p(\theta|y) \quad \text{for } i = 1, \ldots, M$$

Example: Normal Linear Regression

We have seen that for the normal linear regression with known noise variance and prior, $p(\beta) = N(0, v)$, then the posterior is,

$$\begin{aligned} p(\beta|y) &= N(\beta|\hat{\beta}, \hat{v}) \\ \hat{\beta} &= \hat{v}x'y \\ \hat{v} &= (x'x + v^{-1})^{-1} \end{aligned}$$

MCMC would approximate this distribution with M samples drawn from the posterior,

$$\{\beta^{(1)}, \ldots, \beta^{(M)}\} \sim N(\hat{\beta}, \hat{v})$$

1.6 Simulation based inference

- Recall: all the information (needed for, say, predictions, marginals, etc) is contained in the posterior distn. $p(\theta|y)$
- However, $p(\theta|y)$ may not be quantifiable as a standard distribution.
- Trick: suppose we are able to draw samples, $\theta^{(1)}, \ldots, \theta^{(M)}$, from $p(\theta|y)$, so that,

$$\theta^{(i)} \sim p(\theta|y)$$

- Then most inferential quantities of interest are solvable using the bag of samples, $\{\theta^{(i)}\}_{i=1}^{M}$, as a proxy for $p(\theta|y)$.

Examples:

(1) Suppose we are interested in $Pr(\theta < a|y)$. Then,

$$Pr(\theta < a|y) \approx \frac{1}{M} \sum_{i=1}^{M} I(\theta^{(i)} < a)$$

where $I(\cdot)$ is the logical indicator function. More generaly, for a set $A \in \Theta$

$$Pr(\theta \in A|y) \approx \frac{1}{M} \sum_{i=1}^{M} I(\theta^{(i)} \in A)$$

(2) Prediction: Suppose we are interested in $p(\tilde{y}|y)$, for some future $\tilde{y}$. Then,

$$\begin{aligned} p(\tilde{y}|y) &\approx \frac{1}{M}\sum_{i=1}^{M} p(\tilde{y}|\theta^{(i)}, y) \\ &\approx \frac{1}{M}\sum_{i=1}^{M} p(\tilde{y}|\theta^{(i)}) \end{aligned}$$

(3) Inference of marginal effects: Suppose, θ is multivariate and we are interested in the subvector $\theta_j \in \theta$ (for example a particular parameter in a normal linear regression model). Then,

$$F_{\theta_j}(a) \approx \frac{1}{M} \sum_{i=1}^{M} I(\theta_j^{(i)} \leq a)$$

where $F(\cdot)$ denotes the distribution function; More generaly for any set $A_j \in \Theta_j$, the lower dimensional parameter space,

$$Pr(\theta_j \in A_j | y) \approx \frac{1}{M} \sum_{i=1}^{M} I(\theta_j^{(i)} \in A_j)$$

This last point is particularly usefull.

Note that all these quantities can be computed from the same bag of samples. That is, we can first collect $\theta^{(1)}, \ldots, \theta^{(M)}$ as a proxy for $p(\theta|y)$ and then use the same set of samples over and again for whatever we are subsequently interested in.

Finally, a word of warning.....MCMC is a numerial technique and hence subject to approximation error. As we shall see, it is (most) often impossible to quanitify exactly how large this error is. Hence, MCMC is most definatly not a panacea and should be used with caution, almost as a last resort. It is just that often we are at the last resort for interesting models $p(y, \theta)$.

2 Modelling Data

In the previous chapter we discussed some of the advantages and disadvantages of Bayesian data analysis

We showed that MCMC is a powerful simulation technique for inference that is especially useful when we have **non-conjugacy**; when the combination of prior and sampling distribution do not lead to a standard form for the posterior, $p(\theta|y)$

In this chapter we explore how MCMC works in more detail

2.1 MCMC simulation

- As the name suggests, MCMC works by simulating a descrete-time Markov chain.
- That is, it produces a dependent sequence (a chain) of random variables, $\{\theta^{(i)}\}_{i=1}^{M}$, with approximate distribution,

$$p(\theta^{(i)}) \approx p(\theta|y)$$

- The chain is initialised with a user defined starting value, $\theta^{(0)}$
- The Markov property then specifies that the distribution of $\theta^{(i+1)}|\theta^{(i)}, \theta^{(i-1)}, \ldots,$ depends only on the current state of the chain $\theta^{(i)}$

- A Markov chain is specified by it's **transition kernel** defined as,

$$P(\theta^{(i)}, A) =_{\text{def}} P(\theta^{(i+1)} \in A | \theta^{(i)})$$

for any set $A \in \Theta$, which specifies the conditional distribution of $\theta^{(i+1)}$ given the current state $\theta^{(i)}$. If the chain is independent of i it is termed **homogeneous**. We shall allways consider homogeneous chains.

- The n-step transition kernel is,

$$P^n(\theta^{(0)}, A) =_{\text{def}} P(\theta^{(n)} \in A | \theta^{(0)})$$

- MCMC works by constructing the Markov chain in such a way that,

$$P^n(\theta^{(0)}, A) \approx P(\theta \in A|y)$$

for some n, **irrespective of** $\theta^{(0)}$

- Moreover the approximation improves at each step in that,

$$\sup_{A \in \Theta} |P^n(\theta^{(0)}, A) - P(\theta \in A|y)| \to 0, \qquad \text{as} \quad n \to \infty$$

That is the distribution of the state of the chain, $p(\theta^{(i)})$, converges to the target density, $p(\theta|y)$ as i gets “large”

- Broadly speaking, when n is small then $p(\theta^{(n)})$ can often be “far” from $p(\theta|y)$ (given an arbitary starting value $\theta^{(0)}$)
- In this case, we will want to discard the initial set of T samples, $\{\theta^{(0)}, \ldots, \theta^{(T-1)}\}$, as being unrepresentative of the steady state of the chain, $p(\theta|y)$. The time (iteration number) T is known as the **burn-in**
- Knowing when to start collecting samples is a non-trivial task. We shall deal with this later.

- It would be useful at this point to illustrate this with some examples. In this chapter we shall examine some applied problems when MCMC is useful

Example: Logistic Regression - Titanic data

- The data relates to $1,316$ passengers who sailed on the Titanic's maiden and final voyage
- We have data records on whether each passenger survived or not, $y_i \in \{$survived, died $\}$, as well as three attributes of the passenger

(1) Ticket class: $\{$ first, second, third $\}$

(2) Age: $\{$child, adult $\}$

(3) Sex: $\{$female, male $\}$

- We wish to perform a Bayesian analysis to see if there is association between these attributes and survival probability
- As stated before, the Bayesian analysis begins with the specification of a sampling distribution and prior

Sampling density for Titanic survivals

- Let, $y_i \in \{0, 1\}$, denote an indicator of whether the ith passenger survived or not
- We wish to relate the probability of survival, $P(y_i = 1)$, to the passengers covariate information, $x_i = \{\text{class, age, sex}\}$ for the ith passenger
- That is we wish to build a probability model for,

$$p(y_i|x_i)$$

- A popular approach is to use a **Generalised Linear Model (GLM)** which defines this association to be linear on an appropriate scale, for instance,

$$\begin{aligned} P(y_i = 1|x_i) &= g(\eta_i) \\ \eta_i &= x_i\beta \end{aligned}$$

  where $x_i\beta = \sum_j x_{ij}\beta_j$ and $g(\cdot)$ is a monotone **link function**, that maps the range of the **linear predictor**, $\eta_i \in [-\infty, \infty]$, onto the appropriate range, $P(y_i|x_i) \in [0, 1]$

- There is a seperate **regression coefficient**, β_j, associated with each predictor, in our case, $\beta = (\beta_{\text{class}}, \beta_{\text{age}}, \beta_{\text{sex}})'$

- The most popular link function for binary regression (two-class classification) $y_i \in \{0, 1\}$ is the **logit link**, as it quantifies the **Log-odds**

$$\text{logit}(\eta_i) = \frac{1}{1 + \exp(-\eta_i)} = \log\left(\frac{P(y_i = 1|x_i)}{P(y_i = 0|x_i)}\right)$$

where we note, $\text{logit}(\eta_i) \to 0$ as $\eta_i \to -\infty$, $\text{logit}(\eta_i) \to 1$ as $\eta_i \to \infty$

- In this case, the value of the regression coefficients β quantifies the change in the log-odds for unit change in associated x
- This is attractive as clearly β is unknown, **and hence we shall adopt a prior**, $\pi(\beta)$

- It is usual to write the model in hierarchical form,

$$\begin{aligned} p(y_i|x_i) &= g(\eta_i) \\ \eta_i &= x_i\beta \\ \beta &\sim \pi(\beta) \end{aligned}$$

- We are interested in quantifying the statistical association between the survival probability and the attributes, via the posterior density,

$$\begin{aligned} p(\beta|y,x) &\propto p(y|x,\beta)p(\beta) \\ &\propto \left[\prod_{i=1}^{N} p(y_i|x_i,\beta)\right]\pi(\beta) \end{aligned}$$

which is not of standard form

- To infer this we shall use a package known as WinBugs, a Windows version of BUGS (Bayesian analysis Using the Gibbs Sampler)

3 MCMC Algorithms

In the previous chapter we presented an example of using MCMC for simulation based inference.

Up to now we haven’t discussed the algorithms that lie behind MCMC and generate the samples

First, recall that MCMC is an iterative procedure, such that given the current state of the chain, $\theta^{(i)}$, the algorithm makes a **probabilistic** update to $\theta^{(i+1)}$

The general algorithm is

–MCMC Algorithm–

$\theta^{(0)} \leftarrow x$

For i=1 to M

$$\theta^{(i)} = f\left(\theta^{(i-1)}\right)$$

End

where $f\left(\cdot\right)$ outputs a draw from a conditional probability density

- The update, $f(\cdot)$, is made in such a way that the distribution $p(\theta^{(i)}) \rightarrow p(\theta|y)$, the target distribution, as $i \rightarrow \infty$, for any starting value $\theta^{(0)}$

We shall consider two of the most general procedures for MCMC simulation from a target distribution, namely, the **Metropolis-Hastings** algorithm and, the **Gibbs sampler**

3.1 Metropolis-Hastings algorithm

- Let the current state of the chain be $\theta^{(i)}$
- Consider a (any) conditional density $q(\tilde{\theta}|\theta^{(i)})$, defined on $\tilde{\theta} \in \Theta$ (with the same dominating measure as the model)
- We call $q(\cdot|\theta^{(i)})$ the **proposal density** for reasons that will become clear
- We shall use $q(\cdot|\theta^{(i)})$ to update the chain as follows

–M-H Algorithm–

$\theta^{(0)} \leftarrow x$

For i=0 to M

Draw $\tilde{\theta} \sim q(\tilde{\theta}|\theta^{(i)})$

Set $\theta^{(i+1)} \leftarrow \tilde{\theta}$ with probability $\alpha(\theta^{(i)}, \tilde{\theta})$

Else set $\theta^{(i+1)} \leftarrow \theta^{(i)}$, where

$$\alpha(a,b) = \min\left\{1, \frac{p(b|y)q(a|b)}{p(a|y)q(b|a)}\right\}$$

End

Note:

- There is a positive probability of remaining in the same state, $1 - \alpha(\theta^{(i)}, \tilde{\theta})$; and this counts as an extra iteration.
- The process looks like a stochastic hill climbing algorithm. You always accept the proposal if $\frac{p(b|y)q(a|b)}{p(a|y)q(b|a)} > 1$ else you accept with that probability (defined by the ratio)
- The acceptance term corrects for the fact that the proposal density is not the target distribution

- To accept with probability $\frac{p(b|y)q(a|b)}{p(a|y)q(b|a)}$,

  First, draw a uniform random variable, say U, uniform on $[0, 1]$.

  IF $U < \alpha(\theta^{(i)}, \tilde{\theta})$;

  THEN accept $\tilde{\theta}$;

  ELSE reject and chain stays at $\theta^{(i)}$

- The ratio of densities means that the normalising constant $p(y) = \int p(y|\theta)p(\theta)d\theta$ cancels, top and bottom. Hence, we can use MCMC when this is unknown (as is often the case)

- In the special case of a symmetric proposal density **(Hastings algorithm)**, $q(a|b) = q(b|a)$, for example $q(a|b) = N(a|b, 1)$, then the ratio reduces to that of the probabilities

$$\alpha(a, b) = \min \left\{ 1, \frac{p(b|y)}{p(a|y)} \right\}$$

- The proposal density, $q(a|b)$, is user defined. It is more of an art than a science.

- Pretty much any $q(a|b)$ will do, so long as it gets you around the state space Θ. However different $q(a|b)$ lead to different levels of performance in terms of convergence rates to the target distribution and exploration of the model space

Choices for $q(a|b)$

- Clearly $q(a|b) = p(\theta|y)$ leads to an acceptance probability of 1 for all moves and the samples are iid from the posterior
- Of course, the reason we are using MCMC is that we don't know how to draw from $p(\theta|y)$
- It is usual to "centre" the proposal density around the current state and make "local" moves

- There is a trade off: we would like “large” jumps (updates), so that the chain explores the state space, but large jumps usually have low acceptance probability as the posterior density can be highly peaked (and you jump off the mountain side)
- As a rule of thumb, we set the spread of $q()$ to be as large as possible without leading to very small acceptance rates, say < 0.1

- Finally, $q(a|b)$ should be easy to simulate and evaluate: don't make life hard for yourself
- A popular choice when θ is real valued is to take $q(a|b) = b + N(a|0, V)$ where V is user specified. That is, a normal density centred at the current state b.

The Metropolis-Hastings algorithm is a general approach to sampling from a target density, in our case $p(\theta|y)$. However, it requires a user specified proposal density $q(a|b)$ and the acceptance rates must be **continuously** monitored for low and high values. This is not good for automated models (software)

3.2 The Gibbs Sampler

An important alternative approach is available in the following circumstances

- Suppose that the multidimensional θ can be partitioned into p subvectors, $\theta = \{\theta_1, \ldots, \theta_p\}$, such that the conditional distribution,

  $$p(\theta_j|\theta_{-j}, y)$$

  is easy to sample from; where $\theta_{-j} = \theta \backslash \theta_j$

- Iterating over the p subvectors and updating each subvector in turn using $p(\theta_j|\theta_{-j}, y)$ leads to a valid* MCMC scheme known as the **Gibbs Sampler**

- provided the state space remains *connected* (irreducible); which is simple to rectify if it is not

–Gibbs Sampler –

$\theta^{(0)} \leftarrow x$

For i=0 to M

Set $\tilde{\theta} \leftarrow \theta^{(i)}$

For j=1 to p

Draw $X \sim p(\theta_j | \tilde{\theta}_{-j}, y)$

Set $\tilde{\theta}_j \leftarrow X$

End

Set $\theta^{(i+1)} \leftarrow \tilde{\theta}$

End

Note:

- The Gibbs Sampler is a special case of the Metropolis-Hastings algorithm using the ordered sub-updates, $q(\cdot) = p(\theta_j|\theta_{-j}, y)$
- All proposed updates are accepted (there is no accept-reject step)
- θ_j may be multidimensional or univariate
- Often, $p(\theta_j|\theta_{-j}, y)$ will have standard form even if $p(\theta|y)$ does not

Example: normal linear regression

Consider again the normal linear regression model discussed in Chapter 1

$$y = x\beta + \epsilon$$

where $\epsilon \sim N(0, \sigma^2 I)$. Alternately,

$$y \sim N(y|x\beta, \sigma^2 I)$$

we now assume that σ is **unknown**

As before we construct a joint model for the data and unknown parameters,

$$\begin{aligned} p(y, \beta, \sigma^2|x) &= p(y|x, \beta, \sigma^2)p(\beta, \sigma^2|x) \\ &= N(y|x\beta, \sigma^2 I)p(\beta)p(\sigma^2) \end{aligned}$$

where we have assumed that the joint prior for β, σ^2 is independent

Suppose we take,

$$\begin{aligned} p(\beta) &= N(\beta|0, v) \\ p(\sigma^2) &= IG(\sigma^2|a, b) \end{aligned}$$

where $IG(\cdot|a, b)$ denotes the Inverse-Gamma density,

$$IG(x|a, b) \propto x^{-(a-2)/2} \exp(-b/(2x))$$

Then the joint posterior density is,

$$\begin{aligned} p(\beta, \sigma^2|y) &\propto p(y|\beta)p(\beta)p(\sigma^2) \\ &\propto \sigma^{-n/2}\exp[-\frac{1}{2\sigma^2}(y-x\beta)'(y-x\beta)] \times \\ &\quad |v|^{-1/2}\exp[-\beta' v\beta] \times \\ &\quad (\sigma^2)^{-(a-2)/2}\exp(-b/(2\sigma^2)) \end{aligned}$$

This is NOT a standard distribution!

However, the full conditionals ARE known, and moreover,

$$\begin{aligned} p(\beta|y,\sigma^2) &= N(\beta|\hat{\beta},\hat{v}) \\ \hat{\beta} &= (\sigma^{-2}x'x + v^{-1})^{-1}\sigma^{-2}x'y \\ \hat{v} &= (\sigma^{-2}x'x + v^{-1})^{-1} \end{aligned}$$

and

$$\begin{aligned} p(\sigma^2|\beta, y) &= IG(\sigma^2|a+n, b+SS) \\ SS &= (y-x\beta)'(y-x\beta) \end{aligned}$$

Hence the Gibbs sampler can be adopted:

–Gibbs Sampler, normal linear regression–

$(\beta, \sigma^2)^{(0)} \leftarrow x$

For i=0 to M

Set $(\tilde{\beta}, \tilde{\sigma}^2) \leftarrow (\beta, \sigma^2)^{(i)}$

Draw $\tilde{\beta}|\sigma^2 \sim N(\beta|\hat{\beta}, \hat{V})$

Draw $\tilde{\sigma^2}|\tilde{\beta} \sim IG(\sigma^2|a+n, b+SS)$

Set $(\beta, \sigma^2)^{(i)} \leftarrow (\tilde{\beta}, \tilde{\sigma}^2)$

End

Example: hierarchical normal linear regression

Consider again the normal linear regression model discussed in Chapter 1

$$y = x\beta + \epsilon$$

where $\epsilon \sim N(0, \sigma^2 I)$.

we now assume that **BOTH** σ *and prior variance* v *of* $p(\beta)$ *are* **unknown**

In hierarchical form we write,

$$\begin{aligned} y &\sim N(y|x\beta, \sigma^2 I) \\ \beta &\sim N(\beta|0, vI) \\ \sigma^2 &\sim IG(\sigma^2|a, b) \\ v &\sim IG(v|c, d) \end{aligned}$$

note the “hierarchy” of dependencies

where $IG(\cdot|a, b)$ denotes the Inverse-Gamma density,

$$IG(x|a, b) \propto x^{-(a-2)/2} \exp(-b/(2x))$$

Then the joint posterior density is,

$$\begin{aligned} p(\beta, \sigma^2|y) &\propto p(y|\beta)p(\beta)p(\sigma^2) \\ &\propto \sigma^{-n/2}\exp[-\frac{1}{2\sigma^2}(y-x\beta)'(y-x\beta)] \times \\ &\quad |v|^{-1/2}\exp[-\beta' v\beta] \times \\ &\quad (\sigma^2)^{-(a-2)/2}\exp(-b/(2\sigma^2)) \times \\ &\quad v^{-(c-2)/2}\exp(-d/(2v)) \end{aligned}$$

Again, this is NOT a standard distribution!

However, the full conditionals ARE known, and moreover,

$$
\begin{aligned}
p(\beta|y,\sigma^2,v) &= N(\beta|\hat{\beta},\hat{v}) \\
\hat{\beta} &= (\sigma^{-2}x'x + v^{-1})^{-1}\sigma^{-2}x'y \\
\hat{v} &= (\sigma^{-2}x'x + v^{-1})^{-1}
\end{aligned}
$$

and

$$\begin{aligned} p(\sigma^2|\beta, y) &= IG(\sigma^2|a+n, b+SS) \\ SS &= (y-x\beta)'(y-x\beta) \end{aligned}$$

and

$$\begin{aligned} p(v|\beta) &= IG(v|a+p, b+SB) \\ SB &= \beta'\beta \end{aligned}$$

where p is the number of predictors (length of β vector)

Hence the Gibbs sampler can be adopted:

–Gibbs Sampler, hierarchical normal linear regression–

$\{\beta, \sigma^2, v\}^{(0)} \leftarrow x$

For i=0 to M

Set $(\tilde{\beta}, \tilde{\sigma}^2, \tilde{v}) \leftarrow \{\beta, \sigma^2, v\}^{(i)}$

Draw $\tilde{\beta}|\sigma^2, v \sim N(\beta|\hat{\beta}, \hat{V})$

Draw $\tilde{\sigma^2}|\tilde{\beta} \sim IG(\sigma^2|a+n, b+SS)$

Draw $\tilde{v}|\tilde{\beta} \sim IG(v|c+p, d+SB)$

Set $\{\beta, \sigma^2, v\}^{(i)} \leftarrow (\tilde{\beta}, \tilde{\sigma}^2, \tilde{v})$

End

When the conditionals do not have standard form we can usually perform univariate updates (as there are a variety of methods for univariate sampling from a target density) namely,

- Slice sampling
- Rejection sampling
- Ratio of uniforms

Some Issues

- The Gibbs sampler is automatic (no user set parameters) which is good for software, such as WinBugs
- But, M-H is more general and if dependence in the full conditionals, $p(\theta_j|\theta_{-j}, y)$ is strong the Gibbs sampler can be very slow to move around the space, and a joint M-H proposal may be more efficient. The choice of the subvectors can affect this
- We can combine the two in a **Hybrid sampler**, updating some components using Gibbs and others using M-H

4 MCMC Output analysis

In an ideal world, our simulation algorithm would return *iid* samples from the target (posterior) distribution

However, MCMC simulation has two short-comings

1. The distribution of the samples, $p(\theta^{(i)})$ only *converges* with i to the target distribution
2. The samples are dependent

In this chapter we shall consider how we deal with these issues.

We first consider the problem of convergence

Recall that MCMC is an iterative procedure, such that

- Given the current state of the chain, $\theta^{(i)}$, the algorithm makes a **probabilistic** update to $\theta^{(i+1)}$
- The update, $f(\cdot)$, is made in such a way that the distribution $p(\theta^{(i)}) \longrightarrow p(\theta|y)$, the target distribution, as $i \longrightarrow \infty$, for any starting value $\theta^{(0)}$

Hence, the early samples are strongly influenced by the distribution of $\theta^{(0)}$, which presumably is not drawn from $p(\theta|y)$

- The accepted practice is to discard an initial set of samples as being unrepresentative of the steady-state distribution of the Markov chain (the target distribution)
- That is, the first B samples, $\{\theta^{(0)}, \theta^{(1)}, \ldots, \theta^{(B)}\}$ are discarded
- This user defined initial portion of the chain to discard is known as a **burn-in** phase for the chain
- The value of B, the length of burn-in, is determined by You using various **convergence diagnostics** which provide evidence that $p(\theta^{(B+1)})$ and $p(\theta|y)|$ are in some sense “close”

Remember, ALL possible sample paths are indeed possible

4.1 Convergence diagnostics

- WinBugs bundles a collection of convergence diagnostics and sample output analysis programs in a menu driven set of S-Plus functions, called CODA
- CODA implemenents a set of routines for
  - *graphical analysis* of samples;
  - *summary statistics*, and;
  - *formal tests for convergence*

we shall consider the graphical analysis and convergence tests, for more details see the CODA documentation at,

`http://www.mrc-bsu.cam.ac.uk/bugs/documentation/Download/cdaman03.pdf`

Graphical Analysis

The first step in **any** output analysis is to eyeball sample traces from various variables, $\{\theta_j^{(1)}, \ldots, \theta_j^{(M)}\}$, for a set of key variables j

There should be no continuous drift in the sequence of values following burn-in (as the samples are supposed to follow the same distribution)

For example, usually, $\theta^{(0)}$ is far away from the major support of the posterior density

Initially then, the chain will often be seen to “migrate” away from $\theta^{(0)}$ towards a region of high posterior probability centred around a mode of $p(\theta|y)$

The time taken to settle down to a region of a mode is certainly the very minimum lower limit for B

Another useful visual check is to partition the sample chain up into k blocks, $\{\{\theta^{(0)}, \ldots, \theta^{(M/k)}\}, \ldots, \{\cdot, \ldots, \theta^{(M)}\}\}$ and use kernel density estimates for the within block distributions to look for continuity/stability in the estimates

Formal convergence diagnostics

CODA offers four formal tests for convergence, perhaps the two most popular one being those reported by Geweke and those of Gelman and Rubin

4.2 Geweke

- Geweke (1992) proposed a convergence test based on a time-series analysis approach.
- Informally, if the chain has reached convergence then statistics from different portions of the chain should be close

- For a (function of the) variable of interest, the chain is sub-divided up into 2 “windows” containing the initial $x\%$ (CODA default is 10%) and the final $y\%$ (CODA default is 50%).
- If the chain is stationary, the expectations (means) of the values should be similar.
- Geweke describes a test statistic based on a standardised difference in sample means. The test statistic has a standard normal sampling distribution if the chain has converged

4.3 Gelman & Rubin

- Gelman and Rubin (G&R) (1992) proposed a convergence test based on output from **two or more multiple runs of the MCMC simulation**
- G&R is perhaps the most popular diagnostic used today
- The approach uses several chains from different starting values that are **over-dispersed** relative to the posterior distribution. This can often be achieved by sampling from the prior (if vague).
- The method compares the within and between chain variances for each variable. When the chains have “mixed” (converged) the

variance within each sequence and the variance between sequences for each variable will be roughly equal

- They derive a statistic which measures the potential improvement, in terms of the estimate of the variance in the variable, which could be achieved by running the chains to infinity
- When little improvement could be gained, the chains are taken as having converged

Formal tests for convergence should not be taken without question as evidence for convergence. Graphical plots and examining posterior distributions for stability should always be employed for key (functions of) variables of interest

Personally, I always run multiple chains from dispersed starting points to check for stability in my estimates (and use the G&R test)

We now turn to the second problem with MCMC samples.....

Dependence in the chain

MCMC produces a set of dependent samples (conditionally Markov)

What effect does this dependence have on inference?

The Theory

A central limit result for Markov chains holds that

$$\{f(\theta^{(\cdot)}) - E[f(\theta)]\} \rightarrow N(0, \sigma_f^2/M)$$

where $f(\theta^{(\cdot)})$ denotes the empirical estimate for the statistic of interest using the M MCMC samples,

$$f(\theta^{(\cdot)}) = \frac{1}{M} \sum_{i=1}^{M} f(\theta^{(i)})$$

and $E[f(\theta)]$ denotes the true unknown expectation. Note that almost all quantities of interest can be written as expectations

The variance in the estimator, σ_f^2, is given by

$$\sigma_f^2 = \sum_{s=-\infty}^{\infty} \text{cov}[f(\theta^{(i)}), f(\theta^{(i+s)})]$$

Hence, the greater the covariance between samplers, the greater the variance in the MCMC estimator (for given sample size M)

In Practice

The variance parameter σ_f^2 can be approximated using the sample autocovariances

Plots of autocorrelations within chains are extremely useful

High autocorrelations indicate slow mixing (movement around the parameter space), with increased variance in the MCMC estimators

(and usually slower convergence)

Autocorrelations should always be plotted for visual inspection and comparison!

A useful statistic is the **Effective Sample Size**

$$ESS = M/(1 + 2\sum_{j=1}^{k} \rho(j))$$

where M is the number of *post burn-in* MCMC samples and $\sum_{j=1}^{k} \rho(j)$ is the sum of the first k monotone sample autocorrelations

The ESS estimates the reduction in the true number of samples, compared to *iid* samples, due to the autocorrelation in the chain

The ESS is a good way to compare competing MCMC strategies *if you standardise for CPU run time*

If run time is not an issue, but storage is, it is useful to **thin the chain** by only saving one in every T samples - clearly this will reduce the autocorrelations in the saved samples

5 Conclusions

Bayesian data analysis treats **all** unknowns as random variables

Probability is the central tool used to quantify all measures of uncertainty

Bayesian data analysis is about propagating **uncertainty**, from prior to posterior (using Bayes theorem)

Often the posterior will not be of standard form (for example when the prior is non-conjugate)

In these circumstances, sample based simulation offers a powerful tool for inference

MCMC is (currently) the most general technique for obtaining samples from any posterior density - **though it should not be used blindly!**

WinBugs is a user friendly (free) package to construct Bayesian data models and perform MCMC

Additional (minimal) Reading

Chaps 5 and 11 of Gelman’s book (2nd edition)