Markov Chain Monte Carlo and Applied Bayesian Statistics

Hilary Term 2007

Prof. Gesine Reinert

Markov chain Monte Carlo is a stochastic simulation technique that is very useful for computing inferential quantities. It is often used in a Bayesian context, but not restricted to a Bayesian setting.

Outline

1. Review of Bayesian inference

2. Monte Carlo integration

3. Markov chains

4. MCMC in Bayesian inference: ideas

5. MCMC in Bayesian inference: algorithms

6. Output analysis and diagnostics

7. Another example

8. Concluding remarks

Reading

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

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

3. Robert, C.P. and Casella, G. (2004) *Monte Carlo Statistical Methods.* 2nd ed. Springer.

4. Norris, J.R. (1997). *Markov Chains.* Cambridge University Press.

5. (Chen, M-H. et al. (2000). *Monte Carlo Methods in Bayesian Computation.* Springer.)

Lectures will take place Mondays 12-1 and Wednesdays 11-12, Weeks 5-7, in the Department of Statistics. There will be a practical session, using the software package `WinBUGS`, Friday week 5, 1:30 -3 pm, 3 - 4:30 pm, and 4:30 - 6 pm.

Acknowledgement: Chris Holmes for providing his lecture notes and examples, which are partly due to Nicky Best.

1. Review of Bayesian inference

Data $\mathbf{y} = y_1, y_2, \ldots, y_n$, realisations of random variables $Y_1, Y_2, \ldots, Y_n$, with distribution (model)

$$f(y_1, y_2, \ldots, y_n|\theta)$$

$L(\theta|\mathbf{y}) = f(\mathbf{y}|\theta)$ is the likelihood of $\mathbf{y}$ if θ is the true parameter (vector)

Parameter (vector) $\theta = (\theta_1, \ldots, \theta_p)$ has a prior distribution $\pi(\theta)$

Inference is based on the posterior distribution

$$\begin{aligned}\pi(\theta|\mathbf{y}) &= \frac{f(\mathbf{y}|\theta)\pi(\theta)}{\int f(\mathbf{y}|\theta)\pi(\theta)d\theta} \\ &= \frac{L(\theta|\mathbf{y})\pi(\theta)}{\int L(\theta|\mathbf{y})\pi(\theta)d\theta} \\ &\propto L(\theta|\mathbf{y})\pi(\theta)\end{aligned}$$

i.e.

$$Posterior \propto Likelihood \times Prior$$

Three quantities of interest are

1. Prior predictive distribution

$$p(\mathbf{y}) = \int f(\mathbf{y}|\theta)\pi(\theta)d\theta$$

represents the probability of observing the data that was observed *before* it was observed

2. Marginal effects of a subset of parameters in a multivariate model: Suppose that we are interested in $\pi(\theta_i|\mathbf{y})$, for some subset $\theta_i \in \theta$ (here and in the following we abuse notation by using $\theta = \{\theta_1, \ldots, \theta_p\}$ to denote a set as well as a vector). Then

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

where $\theta_{-i} = \theta \setminus \theta_i$ denotes the vector θ with θ_i removed. This distribution is also called the *marginal likelihood*.

3. Posterior predictive distribution: Let $\tilde{y}$ denote some future unobserved response, then the posterior predictive distribution is

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

For the last step we used that $\tilde{y}, \mathbf{y}$ are conditionally independent given θ, though clearly unconditionally they are dependent.

Example (See Statistical Theory, MT 2006)

$X_1, \ldots, X_n$ random sample $\mathcal{N}(\theta, \sigma^2)$, where σ^2 known

prior $\pi(\theta) \sim \mathcal{N}(\mu, \tau^2)$, where μ, τ^2 known

$$f(x_1, \ldots, x_n|\theta) = (2\pi\sigma^2)^{-\frac{n}{2}} exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\frac{(x_i-\theta)^2}{\sigma^2}\right\}$$

so

$$\pi(\theta|\mathbf{x}) \ \propto \ exp\left\{-\frac{1}{2}\left(\sum_{i=1}^{n}\frac{(x_i-\theta)^2}{\sigma^2} + \frac{(\theta-\mu)^2}{\tau^2}\right)\right\}$$

Let

$$\begin{aligned} a &= \frac{n}{\sigma^2} + \frac{1}{\tau^2} \\ b &= \frac{1}{\sigma^2}\Sigma\, x_i + \frac{\mu}{\tau^2} \end{aligned}$$

Calculate:

$$\pi(\theta|x) \sim \mathcal{N}\left(\frac{b}{a}, \frac{1}{a}\right)$$

and the predictive distribution for x is $\mathcal{N}(\mu, \sigma^2 + \tau^2)$.

Bayesian analysis might then continue by calculating the posterior mean, the posterior variance, credible intervals, or using Bayesian hypothesis testing.

Note that in the above example, the posterior again follows a normal distribution:

Prior (normal) $\rightarrow$ Posterior (normal)

conjugate prior: when prior and posterior are in the same family

Computationally even evaluating the posterior distribution, the prior predictive distribution, the marginal likelihoods, and the posterior predictive distribution is not an easy task, in particular if we do not have conjugate priors.

Historically, the need to evaluate integrals was a major stumbling block for the take up of Bayesian methods.

Around 15 years ago or so, a numerical method known as *Markov chain Monte Carlo* (**MCMC**) was popularized by a paper of Gelfand and Smith (1990); other statisticians such as Ripley, Besag, Tanner, Geman were using MCMC before.

2. Monte Carlo integration

In general, when X is a random variable with distribution π, and h is a function, then evaluating

$$E_\pi[h(X)] = \int h(x)\pi(x)dx$$

can be difficult, in particular when X is high-dimensional. However, if we can draw samples

$$x^{(1)}, x^{(2)}, \ldots, x^{(n)} \sim \pi$$

then we can estimate

$$E_\pi[h(X)] \approx \frac{1}{n} \sum_{i=1}^{n} h(x^{(i)}).$$

This is *Monte Carlo integration*

For independent samples, by the law of large numbers we have that, in probability

$$\frac{1}{n}\sum_{i=1}^{n} h(x^{(i)}) \rightarrow E_{\pi}[h(X)] \text{ as } n \rightarrow \infty \tag{1}$$

and the Central Limit Theorem holds under weak assumptions on the distribution of $h(X)$

Application to Bayesian inference

Recall: all the information (needed for, say, predictions, marginals, etc) is contained in the posterior $\pi(\theta|\mathbf{y})$

However, $\pi(\theta|\mathbf{y})$ may not be quantifiable as a standard distribution.

Suppose we are able to draw samples, $\theta^{(1)}, \ldots, \theta^{(M)}$, from $\pi(\theta|\mathbf{y})$, so that,

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

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

Examples:

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

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

where $I(\cdot)$ is the logical indicator function.

More generaly, for a set $A \in \Theta$

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

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

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

(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 generally 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 useful.

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 $\pi(\theta|\mathbf{y})$ and then use the same set of samples over and again for whatever we are subsequently interested in.

Warning: Monte Carlo integration is a last resort; if we can calculate expectations and probabilities analytically, then that would be much preferred.

Independent sampling from $\pi(x)$ may be difficult. Fortunately (1) still applies if we generate samples using a Markov chain, provided some conditions apply - in that case (1) is called the *Ergodic Theorem*. To state the Ergodic Theorem properly, we recall some Markov chain concepts.

3. Markov chains

Suppose that $X_1, X_2, \ldots$ is a sequence of (discrete) random vectors such that, for all $t, \mathbf{x}$,

$$\begin{aligned} & P(X_{t+1} = x_{t+1} | X_t = x_t, X_{t-1} = x_{t-1}, \ldots) \\ = \; & P(X_{t+1} = x_{t+1} | X_t = x_t) \end{aligned}$$

then $(X_t)_{t=0,1,\ldots}$ is called a *Markov chain*. Note that X_{t+1} depends on the past $X_0, X_1, \ldots, X_t$ only through X_t.

A *homogeneous* Markov chain $(X_t)_{t=0,1,\ldots}$ is generated by sampling from a transition kernel $P(y, x)$; if $X_t = x_t$, then $X_{t+1} \sim P(x_t, x)$, for $t = 0, 1, 2, \ldots$; more generally, for any set A,

$$P(x_t, A) := P(X_{t+1} \in A | X_t = x_t).$$

If the transition probabilities depended on t, the chain would be called *inhomogeneous*.

Example. Consider the AR(1) process

$$X_t = \alpha X_{t-1} + \epsilon_t,$$

where the ϵ_t's are independent, identically distributed. Then $(X_t)_{t=0,1,\ldots}$ is a homogeneous Markov chain.

For a Markov chain with finite state space I we can calculate n-step transition probabilities by matrix iteration: If $p_{ij}^{(n)} = Pr(X_n = j | X_0 = i)$, for $i, j \in I$, then

$$(p_{ij}^{(n)})_{i,j \in I} = P^n.$$

Example. A two-state Markov chain $(X_t)_{t=0,1,\ldots}$ has transition matrix

$$P = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}.$$

From the equation $P^{n+1} = P^n P$ we have, for example,

$$p_{11}^{(n+1)} = (1-\alpha)p_{11}^{(n)} + \beta p_{12}^{(n)};$$

this corresponds to conditioning on the nth step:

$$
\begin{aligned}
p_{11}^{(n+1)} &= Pr(X_{n+1}=1|X_0=1) \\
&= Pr(X_{n+1}=1|X_0=1,X_n=1)Pr(X_n=1|X_0=1) \\
&\quad +Pr(X_{n+1}=1|X_0=1,X_n=2)Pr(X_n=2|X_0=1) \\
&= Pr(X_{n+1}=1|X_n=1)p_{11}^{(n)} \\
&\quad +Pr(X_{n+1}=1|X_n=2)p_{12}^{(n)} \\
&= (1-\alpha)p_{11}^{(n)}+\beta p_{12}^{(n)}.
\end{aligned}
$$

From

$$
p_{12}^{(n)}+p_{11}^{(n)} = Pr(X_n = 1 \text{ or } 2) = 1
$$

we obtain for $n \geq 1$,

$$
p_{11}^{(n+1)} = (1-\alpha-\beta)p_{11}^{(n)}+\beta,
$$

and $p_{11}^{(0)} = 1$. Solving the system gives as unique solution

$$p_{11}^{(n)} = \begin{cases} \frac{\beta}{\alpha+\beta} + \frac{\alpha}{\alpha+\beta}(1-\alpha-\beta)^n & \text{for } \alpha+\beta > 0 \\ 1 & \text{for } \alpha+\beta = 0 \end{cases}.$$

A Markov chain has *stationary* or *invariant* distribution π if

$$\int \pi(y)P(y,x)dy = \pi(x), \text{ all } x$$

that is, once we start in the stationary distribution π, all X_t will have the distribution π

In matrix notation: $\pi P = \pi$

Interpretation: In the long run the proportion of time the chain spends in any given state x is proportional to $\pi(x)$.

Fact: If the state space I is finite and $p_{ij}^{(n)} \to \pi_j$ as $n \to \infty$ for all $j \in I$, then $\pi = (\pi_i, i \in I)$ is invariant.

Example: For the two-state Markov chain above, as $n \to \infty$,

$$P^n \to \begin{pmatrix} \frac{\beta}{\alpha+\beta} & \frac{\alpha}{\alpha+\beta} \\ \\ \frac{\beta}{\alpha+\beta} & \frac{\alpha}{\alpha+\beta} \end{pmatrix}$$

and so $\pi = (\frac{\beta}{\alpha+\beta}, \frac{\alpha}{\alpha+\beta})$ is invariant distribution.

You can also check that $\pi P = \pi$.

One can try to break a Markov chain X_n into smaller pieces. We say that $i \to j$, i *communicates* with j, if

$$P(X_n = j \text{ for some } n \geq 0 | X_0 = i) > 0.$$

A Markov chain is *irreducible* if any set of states can be reached from any other state in a finite number of moves, i.e. if $P(X_n = j \text{ for some } n \geq 0 | X_0 = i) > 0$ for all $i, j \in I$. Every state communicates with every other state.

Fact: If the chain is irreducible and if it has a stationary distribution, then the stationary distribution is unique.

A criterion for the existence of a stationary distribution is *reversibility*. The Markov chain $(X_t)_{t=0,1,...}$ as called *reversible* if there is a function π such that the *detailed balance equations* hold:

$$\pi(x_t)p_{x_t,x_{t+1}} = \pi(x_{t+1})p_{x_{t+1},x_t}; \tag{2}$$

the pairs (x_t, x_{t+1}) and (x_{t+1}, x_t) will occur on average with equal frequency in realisations of the Markov chain.

Fact: If the Markov chain is irreducible and if the π in (2) are such that $0 \leq \pi(x) \leq 1$ and $\Sigma_x \, \pi(x) = 1$, then π is the unique equilibrium distribution of the chain.

This approach can be generalised to continuous state spaces.

A state i is *aperiodic* if $p_{ii}^{(n)} > 0$ for all sufficiently large n.

Example. Consider the two-state Markov chain with transition matrix

$$P = \begin{pmatrix} 0 & 1 \\ \\ 1 & 0 \end{pmatrix}.$$

Then $P^2 = I, P^{2n} = I, P^{2n+1} = P$, so each state returns to itself at every second step: the chain is periodic.

Fact: If an irreducible Markov chain has an aperiodic state, then automatically all its states are aperiodic.

Ergodic Theorem: Assume the homogeneous Markov chain has stationary distribution π and is aperiodic and irreducible. Then (1) holds; for any function h such that $\int h(x)\pi(x)dx$ exists,

$$\frac{1}{n}\sum_{t=1}^{n} h(X_t) \to E_\pi[h(X)] = \int h(x)\pi(x)dx \text{ as } n \to \infty.$$

Here, $X \sim \pi$.

Also for such chains with

$$\sigma_h^2 = var_\pi[h(X)] < \infty$$

the central limit theorem holds, and convergence to the stationary distribution occurs (geometrically) fast.

So we can apply Monte Carlo integration to approximate $\int h(x)\pi(x)dx$ by simulating a Markov chain that has π as stationary distribution.

Further reading on Markov chains: J.R. Norris, *Markov chains.* Cambridge University Press, 1997.

Note: Usually it is not possible to start the chain in the stationary distribution - if it was easy to sample from that distribution directly, we would not need a Markov chain in the first place.

If we start the chain in some arbitrary value X_0, then for small n the distribution of the samples may be quite far away from the stationary distribution, and we better discard the initial set of, say, T samples as being unrepresentative.

Knowing when to start collecting samples is a nontrivial task; we shall deal with this later (watch out for *burn-in*).

4. MCMC in Bayesian inference: idea

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

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

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 $\theta^{(i)}$

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

MCMC methods construct a Markov chain on the state space, $\theta \in \Theta$, whose steady state distribution is the posterior of interest $\pi(\theta|\mathbf{y})$

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

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

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

$$\theta^{(i)} \sim \pi(\theta|\mathbf{y}) \qquad \text{for} \;\; i = 1, \ldots, M$$

We shall see that

- MCMC is a general method that simultaneously solves inference of $\{\pi(\theta|\mathbf{y}), \pi(\theta_i|\mathbf{y}), p(\tilde{y}|y)\}$
- MCMC only requires evaluation of the joint distribution

$$\pi(\mathbf{y}, \theta) \propto p(y|\theta)\pi(\theta)$$

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

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, $y_i \in \{\text{survived, died}\}$, 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. 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, $x_i = \{$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,

$$P(y_i = 1|x_i) = g(\eta_i)$$

$$\eta_i = x_i\beta$$

where $x_i\beta = \Sigma_j\, x_{ij}\beta_j$ 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 separate *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,

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

which is not of standard form

To infer this we shall use the WinBUGS package.

Example: Normal Linear Regression

Consider a normal linear regression,

$$\mathbf{y} = \mathbf{x}\beta + \epsilon$$

where $\epsilon \sim N(0, \sigma^2 I)$. Alternatively, $\mathbf{y} \sim \mathcal{N}(\mathbf{x}\beta, \sigma^2 I)$; to make the $\mathbf{y}$-dependence clearer, we write

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

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

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

Bayesian modelling proceeds by constructing a joint model for the data and unknown parameters,

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

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

Suppose we take

$$\pi(\beta) = N(\beta|0, vI),$$

where v is a scalar. Then

$$\begin{aligned}
\pi(\beta|\mathbf{y}) &\propto f(\mathbf{y}|\beta)\pi(\beta) \\
&\propto \sigma^{-n/2}\exp\left\{-\frac{1}{2\sigma^2}(\mathbf{y}-\mathbf{x}\beta)'(\mathbf{y}-\mathbf{x}\beta)\right]\times \\
&\quad |v|^{-1/2}\exp[-(2v)^{-1}\beta'\beta\Big\} \\
&\propto \exp\left[-\frac{1}{2\sigma^2}\beta'\mathbf{x}'\mathbf{x}\beta - (2v)^{-1}\beta'\beta\right. \\
&\quad \left.+\frac{1}{2\sigma^2}(\mathbf{y}'\mathbf{x}\beta + \beta'\mathbf{x}'\mathbf{y})\right].
\end{aligned}$$

We recall that the multivariate normal density $f_{N(\mu,\Sigma)}$ for some vector $\mathbf{z}$ can be written as

$$\begin{aligned} f_{N(\mu,\Sigma)}(\mathbf{z}) &\propto \exp\left\{-\frac{1}{2}(\mathbf{z}-\mu)'\Sigma^{-1}(\mathbf{z}-\mu)\right\} \\ &\propto \exp\left\{-\frac{1}{2}\mathbf{z}'\Sigma^{-1}\mathbf{z}\right. \\ &\qquad \left.+\frac{1}{2}(\mathbf{z}'\Sigma^{-1}\mu+\mu'\Sigma^{-1}\mathbf{z})\right\}. \end{aligned}$$

Matching up the densities we find

$$\Sigma^{-1} = (v^{-1} + \sigma^{-2}\mathbf{x}'\mathbf{x})I$$

so that

$$\Sigma = \sigma^2(\sigma^2 v^{-1} + \mathbf{x}'\mathbf{x})^{-1}I$$

and

$$\mu = \frac{1}{\sigma^2}\Sigma\mathbf{x}'\mathbf{y} = (\mathbf{x}'\mathbf{x} + \sigma^2 v^{-1})^{-1}\mathbf{x}'\mathbf{y}.$$

Therefore we can write

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

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

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

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

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

(and similarly for the predictive densities).

5. MCMC in Bayesian inference: algorithms

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

Up to now we have not 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`

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

`End`

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

The update, $f(\cdot)$, is made in such a way that the distribution $p(\theta^{(i)}) \rightarrow \pi(\theta|\mathbf{y})$, 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**

4.1 The Metropolis-Hastings (M-H) algorithm

Metropolis *et al.* (1953) give an algorithm of how to construct a Markov chain whose stationary distribution is our target distribution π; this method was generalized by Hastings (1970).

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` $\alpha(\theta^{(i)}, \tilde{\theta})$`, where`

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

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

`End`

It can be shown that the Markov chain $(\theta^{(i)}), i = 1, 2, \ldots$ will indeed have $\pi(\theta|\mathbf{y})$ as stationary distribution:

Why does it work?

The key idea is *reversibility* or *detailed balance*:

In general the target distribution π is invariant for P if for all x, y in the state space, the detailed balance equation holds:

$$\pi(x)P(x,y) = \pi(y)P(y,x).$$

We check that the M-H sampler satisfies detailed balance:

Let P be the transition matrix for the M-H chain. Then, for $a \neq b$,

$$\begin{aligned} \pi(a|\mathbf{y})P(a,b) &= \pi(a|\mathbf{y})q(b|a)\alpha(a,b) \\ &= min(\pi(a|\mathbf{y})q(b|a), \pi(b|\mathbf{y})q(a,b)) \end{aligned}$$

and this expression is symmetric in a, b, hence

$$\pi(a|\mathbf{y})P(a,b) = \pi(b|\mathbf{y})P(b,a),$$

and detailed balance is satisfied.

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 $\frac{\pi(b|y)q(a|b)}{\pi(a|y)q(b|a)}$,

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 $p(y) = \int f(y|\theta)\pi(\theta)d\theta$ cancels, top and bottom. Hence, we can use MCMC when the normalizing constant is unknown (as is often the case)

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

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

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

Example

Suppose that we want to generate a random element from the set $\mathcal{S}$ of all permutations $(x_1, \ldots, x_n)$ of the numbers $(1, \ldots, n)$ for which $\Sigma_{j=1}^{n} jx_j > a$ for a given constant a.

We say that two permutations are neighbours of each other if one results from an interchange of two of the position of the other (a *transposition*). So, $(1, 2, 3, 4)$ and $(1, 2, 4, 3)$ are neighbours, whereas $(1, 2, 3, 4)$ and $(1, 3, 4, 2)$ are not.

Let $N(s)$ denote the set of neighbours in $\mathcal{S}$ of a permutation s, then we choose

$$q(t|s) = \frac{1}{|N(s)|}, \quad t \in N(s)$$

that is, the target next state from s is equally likely to be any of its neighbours. Since the desired limiting probabilities of the chain are $\pi(s) = c$, a constant, for $s = (x_1, \ldots, x_n)$ such that $\Sigma_{j=1}^{n} jx_j > a$, and zero otherwise, it follows that

$$\alpha(s,t) = \min\left\{1, \frac{|N(s)|}{|N(t)|}\right\}$$

if s and t are neighbours and in $\mathcal{S}$, and $\alpha(s,t) = 0$ otherwise.

Choices for $q(a|b)$

Clearly $q(a|b) = \pi(\theta|y)$ leads to an acceptance probability of 1 for all moves and the samples are iid from the posterior. But the reason we are using MCMC is that we do not know how to draw from $\pi(\theta|y)$

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

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

It is usual to "centre" the proposal density around the current state and make "local" moves. 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.

Warning. The Metropolis-Hastings algorithm is a general approach to sampling from a target density, in our case $\pi(\theta|y)$. 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)

4.2 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,

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

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

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

Gibbs Sampler

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

For i=0 to M

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

For j=1 to p

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

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, $q(\cdot) = \pi(\theta_j|\theta_{-j}, y)$

All proposed updates are accepted (there is no accept-reject step)

θ_j may be multidimensional or univariate

Often, $\pi(\theta_j|\theta_{-j}, y)$ will have standard form even if $\pi(\theta|y)$ does not

Example: normal linear regression

Consider again the normal linear regression model discussed in Chapter 1

$$\mathbf{y} = \mathbf{x}\beta + \epsilon$$

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

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

we now assume that σ *is* **unknown**

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

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

assuming independence for the priors for β, σ^2

Suppose we take,

$$\pi(\beta) = N(\beta|0, vI)$$

$$\pi(\sigma^2) = IG(\sigma^2|a, b)$$

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,

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

This is not a standard distribution!

However, the full conditionals are known, and

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

and

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

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

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

`Set` $(\beta, \sigma^2)^{(i+1)} \leftarrow (\tilde{\beta}, \tilde{\sigma}^2)$

`End`

Example: hierarchical normal linear regression

Consider again the normal linear regression model

$$\mathbf{y} = \mathbf{x}\beta + \epsilon$$

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

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

In hierarchical form we write,

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

$$v \sim IG(v|c,d)$$

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

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

note the "hierarchy" of dependencies

Then the joint posterior density is

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

Again, this is not a standard distribution!

However, the full conditionals are known, and

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

and

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

and

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

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).

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

6. Output analysis and diagnostics

In an ideal world, our simulation algorithm would return *i.i.d.* 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.

6.1 Convergence and burn-in

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 $p(\theta^{(i)}) \rightarrow \pi(\theta|\mathbf{y})$, the target distribution, as $i \rightarrow \infty$, 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 $\pi(\theta|\mathbf{y})$

The accepted practice is to discard an initial set of samples as being unrepresentative of the stationary 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 $\pi(\theta|\mathbf{y})$ are in some sense "close"

It is worth emphasising from the beginning that in practice no general exact tests for convergence exist.

Tests for convergence should more formally be called *tests for lack of convergence.* That is, as in hypothesis testing, we can usually only detect when it looks like convergence has NOT yet been met.

Remember, all possible sample paths are indeed possible.

Available 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`: 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 `www-fis.iarc.fr/coda/` and at `mrc-bsu.cam.ac.uk/bugs/documentation/coda03/cdaman03.html`.

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: **trace plot** or **history plot**

There should be

- no continuous drift

- no strong autocorrelation

in the sequence of values following burn-in (as the samples are supposed to follow the same distribution)

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 $\pi(\theta|\mathbf{y})$

If the model has converged, the trace plot will move like a snake around the mode of the distribution.

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

The trace is not easy to interpret if there are very many points

The trace can be easier to interpret if it is summarized by

- the cumulative posterior median, and upper and lower credible intervals (say, 95% level)

- moving averages.

If the model has converged, additional samples from the posterior distribution should not influence the calculation of the mean. Running means will reveal if the posterior mean has settled to a particular value.

Kernel density plots

Sometimes non-convergence is reflected in a multimodal distribution. A "lumpy" posterior may indicate non-convergence.

However, do not assume that the chain has converged just because the posteriors "look smooth".

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

Autocorrelation plots

Autocorrelation plots show the serial correlation in the chain. Some correlation between adjacent values will arise due to the Markov nature of the algorithm. Increasing run length should reduce the autocorrelation.

The presence of correlation indicates that the samples are not effective in moving around through the entire posterior distribution.

The autocorrelation will be high if

- the jump function does not jump far enough

- the jump function jumps too far, into a region of low density.

If the level of autocorrelation is high for a parameter of interest, then a trace plot will be a poor diagnostic for convergence.

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, improved by Brooks and Gelman.

Geweke's test

Geweke (1992) proposed a convergence test based on a time-series analysis approach. It is a formal way to interpret the trace.

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.

The test is similar to the 2-sample t-test
The pooled standard deviation is estimated using the time series spectrum.

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.

Gelman & Rubin's test

Gelman and Rubin (GR) (1992) proposed a convergence test based on output from **two or more multiple runs of the MCMC simulation**. This approach was improved by Brooks and Gelman (1998). BGR is perhaps the most popular diagnostic used today.

The approach uses several chains from different starting values. 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.

BGR 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.

However, it is possible that the within-variance and the between-variance are roughly equal but the pooled and the within confidence interval widths do not converge to stability. The improved BGR procedure is as follows.

1. Generate $m \geq 2$ MCMC chains, each with different initial values.

2. Exclude the burn-in period, and iterate for an n-iteration monitored period.

3. From each individual chain the empirical $(1-\alpha)$ CI-interval width is calculated; that is the difference between $\frac{\alpha}{2}$ and $(1-\alpha/2)$ empirical quantiles of the first n simulations. We obtain m within-sequence interval widths estimates.

4. From the entire set of mn observations (pooled), the empirical $(1-\alpha)$ CI-interval width is calculated.

5. $\hat{R}$ is defined as

$$\hat{R} = \frac{\text{width of pooled interval}}{\text{mean width of within-sequence intervals}}.$$

Usually for small n, $\hat{R} > 1$ if the initial values are chosen dispersed enough. The statistic $\hat{R}$ approaches to 1 as the chains converge.

The option bgr diag in WinBUGS calculates the $\hat{R}$ - based diagnostics with $\alpha = 0.2$. is calculated after each 50 simulations. The width of the central 80% interval of the pooled runs is green, the average width of the 80% intervals within the individual runs is blue, and their ratio $\hat{R}$ is red - for plotting purposes the pooled and within interval widths are normalised to have an overall maximum of one. Convergence is achieved when is close to 1, and both the pooled and within interval widths are stable. The values can be listed to a window by double-clicking on the figure followed by ctrl-left-mouse-click on the window.

Other tests

Heidelberger-Welsh: tests for stationarity of the chain, based on Brownian bridge theory

Raftery-Lewis: based on how many iterations are necessary to estimate the posterior for a given quantity; consider a two-state Markov chain with 1 for exceedance of the given quantity

Further reading:

Cowles, M.K. and Carlin, B.P. (1996), Markov Chain Monte Carlo Convergence Diagnostics: a Comparative Review. *Journal of the American Statistical Association* **91**, pp.883 - 905.

and Brooks, S.P. and Roberts, G. O. (1998). Assessing Convergence of Markov Chain Monte Carlo Algorithms. *Statistics and Computing* **8**, pp.319-335.

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.

Warning: Convergence does not mean that you have a good model!

Tricks to speed up convergence

Standardize all your variables by subtracting them from their sample means and dividing by their sample standard deviations. This decreases the posterior correlation between parameters.

Example: $Y_i \sim \mathcal{N}(a + bX_i, 1)$; choose priors $\propto 1$ for a and b. The posterior correlation between a and b is:

$$\rho_{a,b} = -\frac{EX}{\sqrt{E(X) + Var(X)}}.$$

If $|E(X)|$ is large relative to the sample variance of X, then there will be a large posterior correlation between a and b and therefore slow convergence (due to a high autocorrelation in the parameter simulations).

Use WinBUGS Over-relax algorithm (tick the corresponding box in the Update part). This generates multiple samples at each iteration and then selects one that is negatively correlated with the current value. The time per iteration increases, but the within-chain correlations should be reduced, and hence fewer iterations may be necessary. However, this method is not always effective.

Pick good initial values. If your initial values are close to their posterior modes, then convergence should occur relatively quickly.

Just wait. Sometimes models just take a long time to converge.

6.2 Tests for dependence in the chain

MCMC produces a set of dependent samples (conditionally Markov)

The Theory

From the central limit result for Markov chains we have that

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

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

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

and $E[f(\theta)]$ denotes the true unknown expectation. We assume that the chain is aperiodic and irreducible, and that $\sigma_f^2 < \infty$

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 autocorrelations

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)

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 $\Sigma_{j=1}^{k} \rho(j)$ is the sum of the first k monotone sample autocorrelations

The ESS can be estimated from the sample autocorrelation function; ESS estimates the reduction in the true number of samples, compared to *i.i.d.* 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*

We call

$$Eff = \frac{1}{(1 + 2\,\Sigma_{j=1}^{k}\,\rho(j))},$$

that is the ratio of the Effective Sample Size (ESS) to the number of replicates generated (M), the *efficiency* of the MCMC.

The maximum efficiency of the MCMC is ∞ and the minimum is $-\infty$.

ESS is generally smaller than the size of the MCMC sample.

Estimating ESS and efficiency can be done only on the sample from the stationary distribution!

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

7. Another Example

For each electoral precinct $i = 1, \ldots, p$ we observe the fraction of voting-age people who turn out to vote (T_i) and who are black (X_i), along with the number of voting-age people (N_i). The quantities of interest, which remain unobserved because of the secret ballot, are the fraction of blacks who vote (β_i^b) and whites who vote (β_i^w). The proportions β_i^b and β_i^w are not observed because T_i and X_i are from different data sources (electoral results and census data, respectively).

Note that

$$T_i = X_i \beta_i^b + (1 - X_i) \beta_i^w$$

and

$$\beta_i^w = \frac{T_i}{1 - X_i} - \frac{X_i}{1 - X_i}\beta_i^b$$

Let T_i' denote the number of voting-age people who turn out to vote. We assume that T_i' follows a binomial distribution with probability of success

$$\theta_i = X_i\beta_i^b + (1 - X_i)\beta_i^w$$

and count N_i.

Next we assume that β_i^b is sampled from a beta distribution with parameters c_b and d_b and that, independently, β_i^w is sampled from a beta distribution with parameters c_w and d_w.

Finally we assume that c_b, d_b, c_w and d_w follow independent exponential distributions with mean 2.

How do we draw inference for this model?

Step 1: Calculate the posterior distribution:

$$
\begin{aligned}
& p(data|\beta_i^b, \beta_i^w, i = 1, \ldots, p) \\
& \quad \times p(\beta_i^b, \beta_i^w, i = 1, \ldots, p|c_b, d_b, c_w, d_w) \\
& \quad \times p(c_b, d_b, c_w, d_w) \\
& \propto \prod_{i=1}^{p} (X_i\beta_i^b + (1 - X_i)\beta_i^w)^{T_i'}(1 - X_i\beta_i^b - (1 - X_i)\beta_i^w)^{N_i - T_i'} \\
& \quad \times \prod_{i=1}^{p} \frac{\Gamma(c_b + d_b)}{\Gamma(c_b)\Gamma(d_b)}(\beta_i^b)^{c_b - 1}(1 - \beta_i^b))^{d_b - 1} \\
& \quad \times \prod_{i=1}^{p} \frac{\Gamma(c_w + d_w)}{\Gamma(c_w)\Gamma(d_w)}(\beta_i^w)^{c_w - 1}(1 - \beta_i^w))^{d_w - 1} \\
& \quad \times \exp(-2c_b) \exp(-2c_w) \exp(-2d_b) \exp(-2d_w)
\end{aligned}
$$

Obtaining the marginals is not feasible, hence we use the Gibbs sampler.

Step 2: Calculate the full conditional distributions:

$$p(\beta_i^b|\beta_i^w, c_b, d_b)$$

$$\propto (X_i\beta_i^b + (1-X_i)\beta_i^w)^{T_i'}(1 - X_i\beta_i^b - (1-X_i)\beta_i^w)^{N_i - T_i'}$$

$$\times(\beta_i^b)^{c_b-1}(1-\beta_i^b))^{d_b-1}$$

and

$$p(\beta_i^w|\beta_i^b, c_w, d_w)$$

$$\propto (X_i\beta_i^b + (1-X_i)\beta_i^w)^{T_i'}(1 - X_i\beta_i^b - (1-X_i)\beta_i^w)^{N_i - T_i'}$$

$$\times(\beta_i^w)^{c_w-1}(1-\beta_i^w))^{d_w-1}$$

and

$$p(c_b|\beta_i^b, i = 1, \ldots, p; d_b)$$

$$\propto \left(\frac{\Gamma(c_b + d_b)}{\Gamma(c_b)}\right)^p$$

$$\times \exp\{(\sum_{i=1}^{p} \log(\beta_i^b) - 2)c_b\}$$

and

$$
\begin{aligned}
& p(d_b|\beta_i^b, i = 1, \ldots, p; d_b) \\
& \quad \propto \left(\frac{\Gamma(c_b + d_b)}{\Gamma(d_b)}\right)^p \\
& \qquad \times \exp\{(\sum_{i=1}^{p} \log(1 - \beta_i^b) - 2)d_b\}
\end{aligned}
$$

and

$$
\begin{aligned}
& p(c_w|\beta_i^w, i = 1, \ldots, p; d_w) \\
& \quad \propto \left(\frac{\Gamma(c_w + d_w)}{\Gamma(c_w)}\right)^p \\
& \qquad \times \exp\{(\sum_{i=1}^{p} \log(\beta_i^w) - 2)c_w\}
\end{aligned}
$$

and

$$
\begin{aligned}
& p(d_w|\beta_i^w, i = 1, \ldots, p; d_w) \\
& \quad \propto \left(\frac{\Gamma(c_w + d_w)}{\Gamma(d_w)}\right)^p \\
& \qquad \times \exp\{(\sum_{i=1}^{p} \log(1 - \beta_i^w) - 2)d_w\}
\end{aligned}
$$

Step 3: Generate a Gibbs sampler: draw random samples from each of these full conditionals, in turn updating the variables afer each draw.

Unfortunately none of the full conditionals are standard distributions for which pre-written subroutines are available. So we use the Metropolis algorithm (acceptance-rejection) to sample from each of these distributions! Use as proposal density the uniform density with mean the current sample value and variance sufficiently large.

Example: Data from 275 counties in four U.S. States: Florida, Louisiana, North Carolina, and South Carolina, in 1968

Use posterior mean to estimate:

mean of posterior distribution for blacks is 0.60 (0.04)

mean of posterior distribution for whites is 0.85 (0.02)

Compare to fraction of registered blacks in all counties: 0.56

fraction of registered whites in all counties: 0.85

Can also detect e.g. bimodality in posterior distribution

Could include covariates in model for β_i^b, β_i^w

See *King, G., Rosen, O., and Tanner, M. (1999).* Binomial-Beta Hierarchical Models for Ecological Inference. *Sociological Methods and Research* **28**, 61–90.

8. Concluding remarks

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.