Stat314 Term 4: Metropolis and Metropolis-Hastings algorithms for posterior computation

Patrick Graham

September, 2021

The basic idea of Metropolis type algorithms for sampling the posterior: Informal description

i Start at some point in the parameter space.

ii Propose a jump to another point by drawing a proposed value from some "jumping density."

iii If the proposed jump is to a point with higher posterior density than the current point, move to the proposed value (accept the proposal) If the proposed jump is to a point with lower posterior density than the current value, accept the proposed jump with a probability related to the ratio of the unnormalised posterior densities at the proposed and current values. If the proposed jump is rejected stay at the current value.

iv repeat steps (ii) to (iii) until the desired simulation size is reached.

Summary of the basic idea of Metropolis sampling

- "exploring the parameter space" but in a manner guided by the (unnormalized) posterior density. Always move to a point with higher posterior density; sometimes move to a point with lower density.
- As this process evolves over a large number of iterations we end up with a large number of points from areas of high posterior density and a small number of points from areas of low posterior density.
- Exactly what we want!

animation
`https://chi-feng.github.io/mcmc-demo/`

Background on Metropolis and Metropolis-Hastings algorithm (i)

- Metropolis algorithm named after Nicolas Metropolis - pioneer of Monte Carlo methods in Physics along with Stanislaw Ulam. Ideas developed in 1940s, but general application to Bayesian computation not recognised until 1990s.
- This algorithm assumes a symmetric jumping density which is a density satisfying $J_t(\boldsymbol{\theta}^{(a)}|\boldsymbol{\theta}^{(b)}) = J_t(\boldsymbol{\theta}^{(b)}|\boldsymbol{\theta}^{(a)}), \forall \boldsymbol{\theta}^{(a)}, \boldsymbol{\theta}^{(b)}$ where by $\boldsymbol{\theta}^{(a)}, \boldsymbol{\theta}^{(b)}$ we mean two particular values of $\boldsymbol{\theta}$ (i.e two points in the parameter space).

Background on the Metropolis and Metropolis-Hastings algorithm (ii)

- The normal density, with fixed precision, is the most commonly used example of a symmetric jumping density.

$$[\theta^{(a)}|\theta^{(b)}] \sim N(\theta^{(b)}, \tau) \tag{1}$$

(where τ is the precision) implies

$$p(\boldsymbol{\theta}^{(a)}|\boldsymbol{\theta}^{(b)}) = \frac{\sqrt{\tau}}{\sqrt{2\pi}} \exp\left(-\frac{\tau(\boldsymbol{\theta}^{(a)} - \theta^{(b)})^2}{2}\right) \tag{2}$$

$$= \frac{\sqrt{\tau}}{\sqrt{2\pi}} \exp\left(-\frac{\tau(\boldsymbol{\theta}^{(b)} - \boldsymbol{\theta}^{(a)})^2}{2}\right) \tag{3}$$

so $p(\boldsymbol{\theta}^{(a)}|\boldsymbol{\theta}^{(b)}$ is symmetric in $\boldsymbol{\theta}^{(a)}$ and $\boldsymbol{\theta}^{(b)}$.

- The Metropolis-Hastings allows algorithm allows asymmetric jumping densities. Introduced by Hastings (1970).

The Metropolis -Hastings algorithm

1. draw $\boldsymbol{\theta}^{(0)}$ from an initial density $g_0(\boldsymbol{\theta})$; we require $p(\boldsymbol{\theta}^{(0)}|\text{data}) > 0$ (posterior has positive density at the generated initial point).
2. for $t = 1, 2, \ldots,$
   sample a proposal, $\boldsymbol{\theta}^{(*)}$ from a jumping (or proposal) distribution, $J_t(\boldsymbol{\theta}^{(*)}|\boldsymbol{\theta}^{(t-1)})$,
3. calculate the Metropolis-Hastings acceptance ratio

$$r_{MH}(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{(t-1)}) = \frac{q(\boldsymbol{\theta}^*|\text{data})/J_t(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(t-1)})}{q(\boldsymbol{\theta}^{(t-1)}|\text{data})/J_t(\boldsymbol{\theta}^{(t-1)}|\boldsymbol{\theta}^*)} \tag{4}$$

4. If $r_{MH}(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{(t-1)}) > 1$ set $\boldsymbol{\theta}^{(t)} = \boldsymbol{\theta}^*$; else set

$$\boldsymbol{\theta}^{(t)} = \begin{cases} \boldsymbol{\theta}^{(*)} \text{ with probability } r_{MH}(\boldsymbol{\theta}^{(*)}, \boldsymbol{\theta}^{(t-1)}) \\ \boldsymbol{\theta}^{(t-1)} \text{ otherwise} \end{cases}$$

Simplification r_{MH} for symmetric jumping density

For symmetric $J(\theta|\theta^{(t-1)})$

$$r_{MH}(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{(t-1)}) = \frac{q(\boldsymbol{\theta}^*|\text{data})/J_t(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(t-1)})}{q(\boldsymbol{\theta}^{(t-1)}|\text{data})/J_t(\boldsymbol{\theta}^{(t-1)}|\boldsymbol{\theta}^*)} \tag{5}$$

$$= \frac{q(\boldsymbol{\theta}^*|\text{data})}{q(\boldsymbol{\theta}^{(t-1)}|\text{data})} \tag{6}$$

So the M-H acceptance ratio only depends only on the ratio of the values of unnormalized density at the proposed new point and the current point.

Notes on implementation of the Metropolis algorithm

- To determine whether a proposal is accepted draw $U \sim Uniform(0, 1)$; if $U \leq r_{MH}(\boldsymbol{\theta}^{(*)}, \boldsymbol{\theta}^{(t)})$ then accept $\boldsymbol{\theta}^{(t)} = \boldsymbol{\theta}^{(*)}$; else $\boldsymbol{\theta}^{(t)} = \boldsymbol{\theta}^{(t-1)}$.
- Early draws in the sequence will not be from the posterior distribution.
- Run the algorithm until convergence; discard early draws; subsequent draws can be regarded as draws from the target posterior.
- The draws are correlated - this means the effective Monte Carlo sample size, is less than the actual size.
- detecting convergence is difficult. Best approach is to run multiple chains from different starting points; inspect traceplots; use the the Gelman-Rubin $\hat{R}$ statistic.

Diagnosing convergence of MCMC procedures

- run multiple chains from different starting values; discard early draws (e.g first 50%; compare within and between chain variation, formalised in the Gelman-Rubin statistic ($\hat{R}$) - see next slide
- Inspect trace plots (see below for examples)
- if only a single chain is run, look at trace plot; experiment with different starting values; experiment with different burn-in periods.

illustration of trace plots (1)

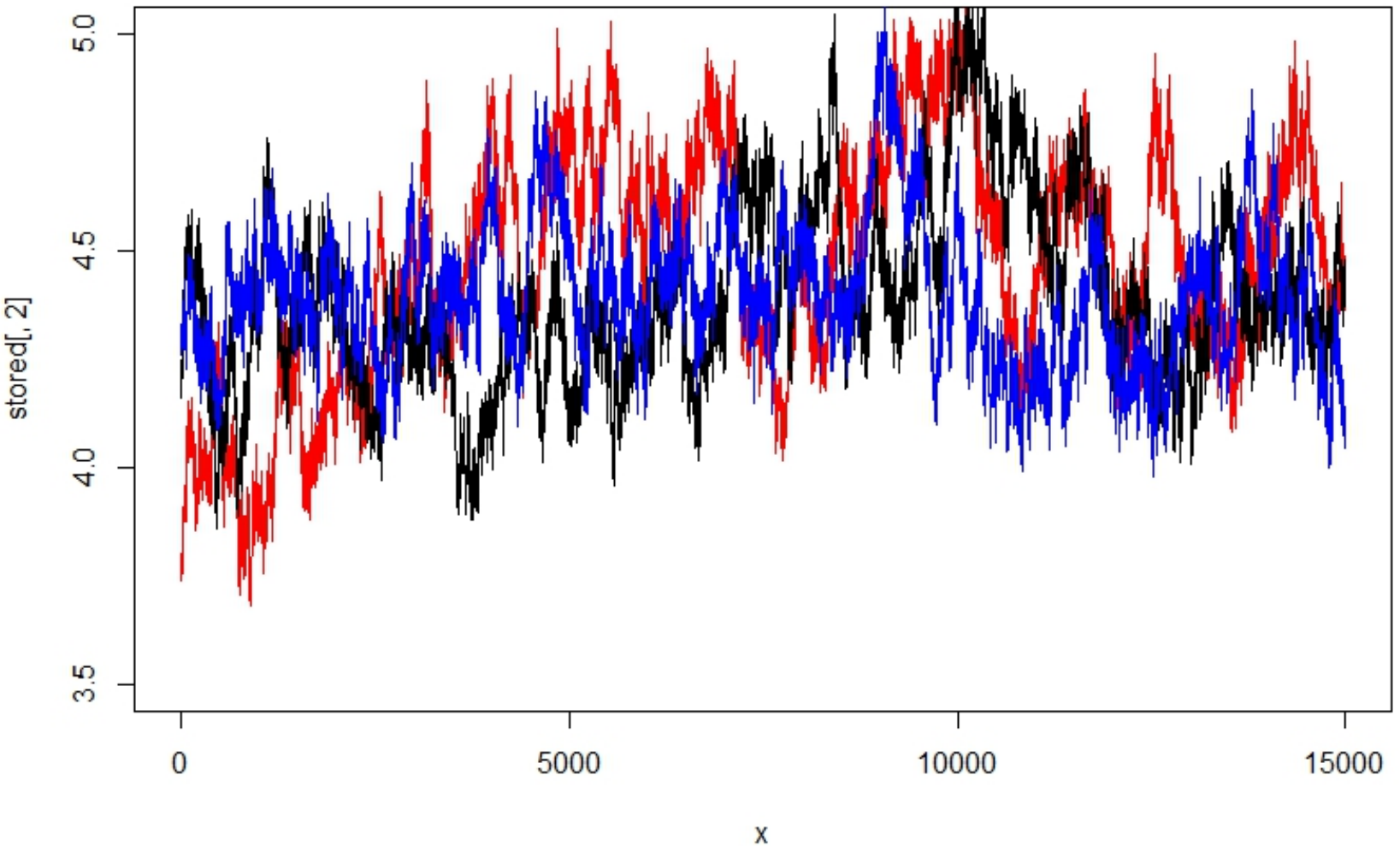
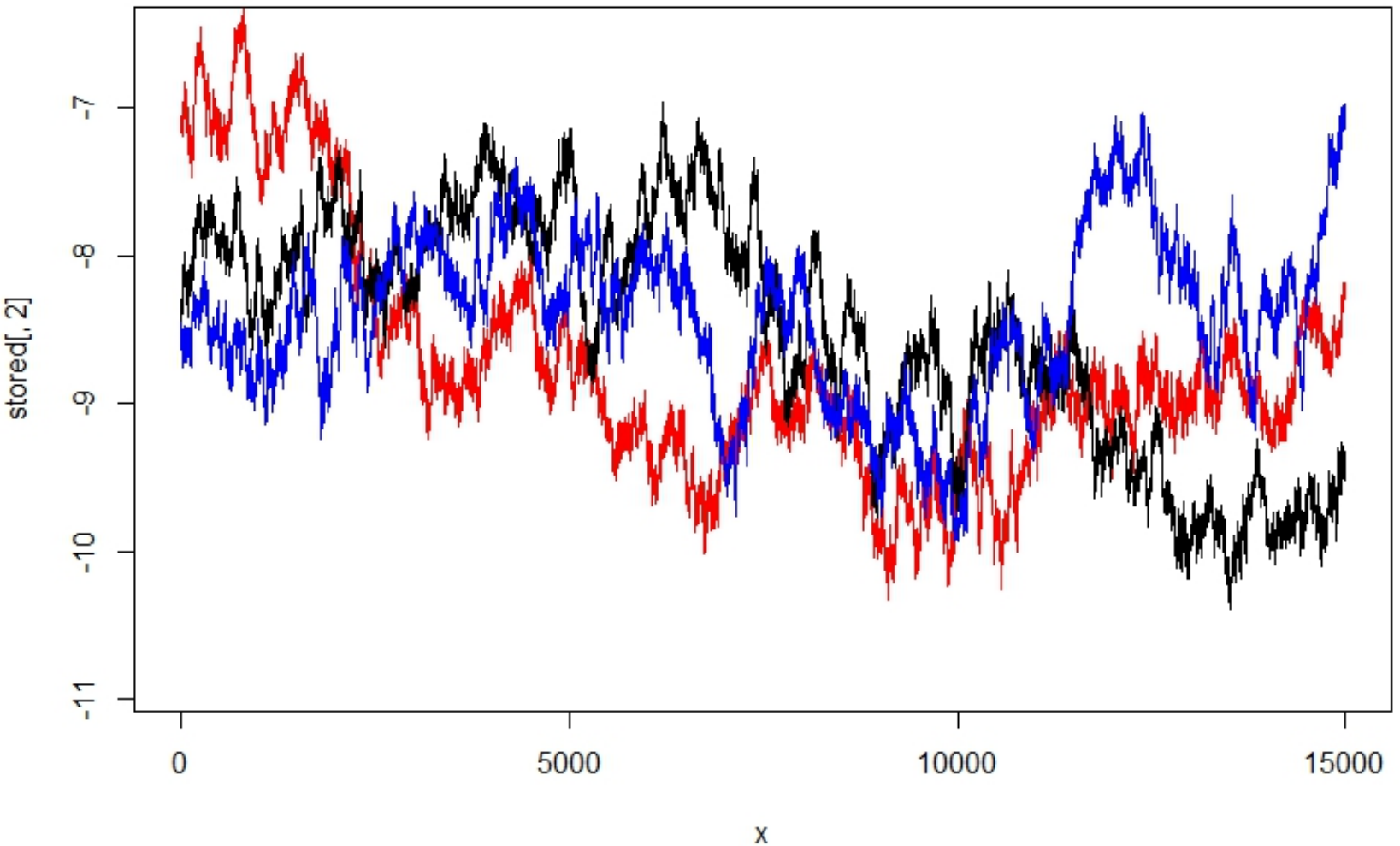


illustration of trace plots (2)



Convergence problems: Mixing and stationarity

See Figure 11.3 Gelman et BDA 3rd edition, p 283.

Gelman-Rubin diagnostic for detecting MCMC sampler convergence

- Suppose we have m chains of length K after discarding the burn-in period.
- Split each chain in two, so now we have $2m$ chains of length $K/2$. We work with these split - chains;
- If chains have reached their stationary distribution by the end of the burn-in, the two-halves of the post burn-in chain should be similar
- Let W be the within chain variance (compute variance of draws for each chain and average over chains), and B the between chain variance ($K/2$ times variance in chain means).

$$V^+ = \frac{K/2-1}{K/2}W + \frac{1}{K/2}B$$

$$\hat{R} = \sqrt{\frac{V^+}{W}} \to 1 \text{ as } K \to \infty$$

- Values close to 1 suggest convergence.
- Supported by asymptotic arguments (see Gelman et al BDA)

Choice of initial values

- Not important in terms of final posterior sample and hence posterior inferences. The sampler will move away from points with low support in the posterior. We discard early draws.
- However initial values should be plausible to avoid numerical issues in early stages of the sampling.
- Ideally, in the multiple chain setting, starting values should be generated from an overdispersed approximation to the posterior. Theory behind $\hat{R}$ is based on this idea. Sometimes it is hard to find an approximation to the posterior.

Choice of initial values cont'd

- If we can maximize the unnormalized posterior density wrt the parameters to find the mode (same as mode of normalized posterior) an approximation can often be based on a normal or t distribution centered on the mode with variance determined by the shape of the posterior near the mode – related to the inverse second -derivative matrix. See the R function laplace() in LearnBayes.

Binomial-logit-normal example

Six Binomial experiments, each of size 10. Number of successes is $Y = (6, 7, 5, 5, 4, 8)$

$$Y_i | N, \boldsymbol{\theta} \overset{\text{indep}}{\sim} \text{Binomial}(\boldsymbol{\theta}, N)$$

$$[\log(\frac{\boldsymbol{\theta}}{1-\boldsymbol{\theta}})] \sim \text{Normal}(\mu, \sigma^2)$$

Equivalently, could write this as an intercept only logistic regression model:

$$[\log(\frac{\boldsymbol{\theta}}{1-\boldsymbol{\theta}})] = \beta_0 \tag{7}$$

$$\beta_0 \sim \text{Normal}(\mu, \sigma^2) \tag{8}$$

Note

$$\eta = \log(\frac{\boldsymbol{\theta}}{1-\boldsymbol{\theta}}) \Leftrightarrow \boldsymbol{\theta} = \frac{\exp(\eta)}{1+\exp(\eta)} = \frac{1}{1+\exp(-\eta)} \tag{9}$$

We write $\text{logit}(\boldsymbol{\theta})$ for $\log(\frac{\boldsymbol{\theta}}{1-\boldsymbol{\theta}})$ and $\text{invlogit}(\eta)$ for $\frac{\exp(\eta)}{1+\exp(\eta)} = \frac{1}{1+\exp(-\eta)}$

Likelihood and posterior for Binomial parameterised in terms of the logit (η)

Suppose

$$[Y_i|\theta, N_i] \overset{\text{indep}}{\sim} \text{Binomial}(\theta, N_i),\ i = 1, \ldots, K$$
$$\eta = \text{logit}(\theta);\ \eta \sim \text{Normal}(\mu, \sigma^2) \tag{10}$$

$$p(\mathbf{Y}|\eta, \mathbf{N}) = \prod_i \binom{N_i}{\text{invlogit}(\eta)} \text{invlogit}(\eta)^{Y_i}(1 - \text{invlogit}(\eta))^{N_i - Y_i}$$
$$\propto \prod_i \text{invlogit}(\eta)^{Y_i}(1 - \text{invlogit}(\eta))^{N_i - Y_i}$$

so

$$p(\eta|\mathbf{Y}, \mathbf{N}) \propto \left[\prod_i \text{invlogit}(\eta)^{Y_i}(1 - \text{invlogit}(\eta))^{N_i - Y_i}\right] \times$$
$$\times \text{Normal}(\eta|\mu, \sigma^2)$$

Not a conjugate model.

Setting up the Metropolis sample for the Binomial - logit-normal model

- We will draw initial values from a normal approx based on the the posterior mode and inverse of negative second derivative of the log-posterior, V_{approx}
- Jumping distribution will be a normal centred on the current value (hence symmetric)

$$J_t(\eta^{(t)}|\eta^{(t-1)}) = normal(\eta^{(t)}|\eta^{(t-1)}, v)$$

  for some suitable chosen variance v (part of the art of setting up Metropolis samplers). We will use $v = 2.4^2 V_{approx}$
- accept jumps with probability

$$\min\left(1, \frac{q(\eta^{(t)}|\mathbf{Y}, \mathbf{N})}{q(\eta^{(t-1)}|\mathbf{Y}, \mathbf{N})}\right)$$

See R code for example of computations

`MH_example1_binomlogit_2021.r`

Tuning Metropolis-Hastings algorithms (i)

- If jumps are too small the chain will move too slowly through the space; may not traverse the full space; convergence may be too slow. Depending on the circumstances, high acceptance rates can be an indicator that the jumps are too small.
- if jumps are too big acceptance rates will be low and the sampler inefficient.
- For approximately normal posteriors Gelman et al (BDA) suggest a (multivariate) normal jumping density centred on the current value with variance given by $V_{jump} \approx (2.4^2/d)\Sigma$ where Σ is the variance of the approximate normal posterior.
- This provides another reason to try and construct an approximation based on on the mode and shape of the posterior near the mode.

Tuning Metropolis-Hastings algorithms (ii)

- Gelman et al's suggested jumping density is not a hard and fast rule, however.
- It may be necessary to change the jumping density as the sampler evolves -reduce jumping density variance if acceptance rates are too low; increase the jumping density variance if acceptance rates are too high.
- One possibility is to:
  1. start the sampler with Gelman et al's recommended jumping density
  2. after some number of draws change the variance of the jumping density to be proportional to the variance of the draws generated so far;
  3. increase or decrease the jumping density variance depending on acceptance rates;

- If tuning is necessary, think in terms of an adaptive phase where the jumping density is tuned; followed by a fixed phase where the tuned jumping density is used and the sampler is run for a long period - only draws from the fixed phase should be used for inference.

Final notes on Metropolis-Hastings

- The posterior sample obtained comprises correlated draws from the posterior; The effective simulation sample size is therefore less than the nominal simulation sample size; sometimes dramatically so.
- Computation of the effective sample size for correlated draws from a distribution is tricky but some time-series concepts can be used; won't discuss here but the `coda` package contains an `effectiveSize()` function which adjusts nominal size for correlation.
- The chains can be thinned to get closer to independent draws if needed.
- Two broad classes of jumping densities:
  1. random walk style - mostly what we have been considering; often these are (multivariate) normal densities centred on the current value.
  2. Draws from an approximate posterior density - the Metropolis-Hastings acceptance probabilities then act to correct the approximation. In this case we would like a high acceptance rate

Outline proof of the convergence of the Metropolis algorithm to the posterior distribution

(based on Gelman et al BDA, 3rd edn. pp. 279)

1. Recognise the sequence of draws as a Markov chain; the states of the chain are points in the parameters space
2. If the Markov Chain is aperiodic, not transient and irreducible then it converges to a unique stationary distribution. In different ways these conditions are all getting at the idea that each state (point) should be eligible to be visited at each iteration. The first two conditions are virtually guaranteed for a random walk on a proper distribution (assuming the jumping distribution is alsoproper); irreducibility captures the idea that each state should able to be reached (eventually) from any other state. This amounts to a condition on the jumping distribution - it must eventually be able to jump to all states with positive probability.

Outline proof continued

3 Assuming the sequence of draws constitute an aperiodic, non-transient and irreducible Markov Chain, if we can show that the target posterior distribution is the stationary distribution of the chain, then, in view of the uniqueness of the stationary distribution of the chain, we are done.

Outline proof continued: Definition of stationarity

Definition of stationarity
if $g(\boldsymbol{\theta})$ is some density such that if $\boldsymbol{\theta}^{(t-1)}, \boldsymbol{\theta}^{(t)}$ are successive values from the Markov Chain $\tilde{\boldsymbol{\theta}}$,

$$p(\boldsymbol{\theta}^{(t-1)} = \boldsymbol{\theta}_a) = g(\boldsymbol{\theta}_a) \Rightarrow p(\boldsymbol{\theta}^{(t)} = \boldsymbol{\theta}_a) = g(\boldsymbol{\theta}_a) \, \forall \boldsymbol{\theta}_a \tag{11}$$

then $g()$ is a stationary distribution of the chain. That is if $\boldsymbol{\theta}^{(t-1)}$ being a draw from the density $g()$ *implies* $\boldsymbol{\theta}^{(t)}$ is also a draw from the density $g()$ then $g()$ is a stationary distribution of the Markov Chain.

Outline proof of convergence of Metropolis algorithm - cont'd

Since the marginal distributions of exchangeable random variables are identical (see Assignment 4, question 1), if we can show $\boldsymbol{\theta}^{(t-1)}$ being a draw from $p(\boldsymbol{\theta}|\text{data})$ implies $\boldsymbol{\theta}^{(t-1)}, \boldsymbol{\theta}^{(t)}$ are exchangeable then we are done, since this implies $\boldsymbol{\theta}^{(t)}$ is also a draw from $p(\boldsymbol{\theta}|\text{data})$.

See separate notes for details