ECON5120 Bayesian Data Analysis

Week 3: MCMC Methods

Santiago Montoya-Blandón

University of Glasgow
Adam Smith Business School

January 29, 2025

Outline

Posterior expectations

- Remember that our objects of interest will be quantities of the form

$$\mathbb{E}_{p(\theta|y)}[f(\theta)] = \int f(\theta)p(\theta|y)d\theta$$

for some function $f(\theta)$ where

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

- We can easily evaluate $p(y|\theta)p(\theta)$ for any θ, but the integral $\int p(y|\theta)p(\theta)d\theta$ is usually difficult
- Monte Carlo integration: if we can sample from $p(\theta|y)$, we can compute an approximation to $\mathbb{E}_{p(\theta|y)}[f(\theta)]$

Monte Carlo

- Monte Carlo methods we have discussed so far
  - Analytical solutions exist for only certain distributions
  - Rejection sampling
    - Idea: sample from a proposal and keep only appropriate draws
    - Cons: Works mostly in 1D or for special cases such as truncation (even then, problems exist)
  - Importance sampling
    - Idea: Draw from proposal and weigh draws
    - Cons: Choice of proposal, infinite variance, reliable in special cases

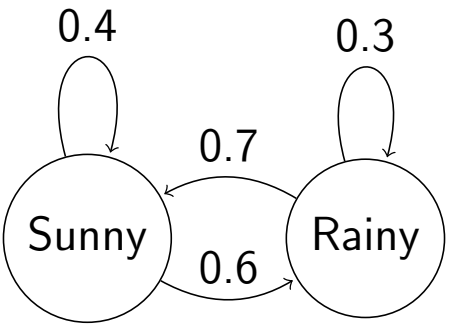
Monte Carlo

- Monte Carlo methods we have discussed so far
  - Analytical solutions exist for only certain distributions
  - Rejection sampling
    - Idea: sample from a proposal and keep only appropriate draws
    - Cons: Works mostly in 1D or for special cases such as truncation (even then, problems exist)
  - Importance sampling
    - Idea: Draw from proposal and weigh draws
    - Cons: Choice of proposal, infinite variance, reliable in special cases
- What to do in high dimensions?
  - Markov chain Monte Carlo (this class, BDA Ch. 11–12)
  - Laplace*, Variational*, EP* (BDA Ch. 4, 13*)

Markov chain

- So far, we have worked with independent sampling
- Markov chains are one of the simplest examples of *dependent* processes but they are remarkably useful
- Intuition: a process that moves across different states with a given probability
- Probability of moving only depends on current and previous state

Markov chain: example



Markov chain

Technical definition

Set of random variables $\theta^{(1)}, \theta^{(2)}, \ldots$, so that for all values of s, $\theta^{(s)}$ depends only on the previous $\theta^{(s-1)}$

$$p(\theta^{(s)}|\theta^{(1)}, \ldots, \theta^{(s-1)}) = p(\theta^{(s)}|\theta^{(s-1)})$$

Markov chain

Technical definition

Set of random variables $\theta^{(1)}, \theta^{(2)}, \ldots$, so that for all values of s, $\theta^{(s)}$ depends only on the previous $\theta^{(s-1)}$

$$p(\theta^{(s)}|\theta^{(1)}, \ldots, \theta^{(s-1)}) = p(\theta^{(s)}|\theta^{(s-1)})$$

- Chain has to be initialized with some starting point $\theta^{(0)}$
- Transition distribution: $p(\theta^{(s)}|\theta^{(s-1)}) = T_s(\theta^{(s)}|\theta^{(s-1)})$ (may depend on s)

Markov chain

Technical definition

Set of random variables $\theta^{(1)}, \theta^{(2)}, \ldots$, so that for all values of s, $\theta^{(s)}$ depends only on the previous $\theta^{(s-1)}$

$$p(\theta^{(s)}|\theta^{(1)}, \ldots, \theta^{(s-1)}) = p(\theta^{(s)}|\theta^{(s-1)})$$

- Chain has to be initialized with some starting point $\theta^{(0)}$
- Transition distribution: $p(\theta^{(s)}|\theta^{(s-1)}) = T_s(\theta^{(s)}|\theta^{(s-1)})$ (may depend on s)
- *Stationary distribution*: a distribution π such that $\pi(\theta^{(s)}) = \pi(\theta^{(s-1)})$
- We will exploit some properties of Markov chains to construct one whose stationary distribution is our object of interest $p(\theta|y)$

Markov chain Monte Carlo (MCMC)

MCMC methods

Produce S draws $(\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(S)})$ by drawing $\theta^{(s)}$ given $\theta^{(s-1)}$ from a Markov chain, which has been constructed so that its equilibrium distribution is $p(\theta|y)$.

Markov chain Monte Carlo (MCMC)

MCMC methods

Produce S draws $(\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(S)})$ by drawing $\theta^{(s)}$ given $\theta^{(s-1)}$ from a Markov chain, which has been constructed so that its equilibrium distribution is $p(\theta|y)$.

- Pros
  - generic
  - chain goes where most of the posterior mass is
  - asymptotically chain spends the $\alpha\%$ of time where $\alpha\%$ posterior mass is
  - central limit theorem holds for computing expectations
- Cons
  - draws are dependent
  - efficiency and convergence

Monte Carlo integration with dependent draws

- Using draws $(\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(S)})$, we can estimate $\mathbb{E}_{p(\theta|y)}[f(\theta)]$ as

$$\hat{\mu}_f = \hat{\mathbb{E}}_{p(\theta|y)}[f(\theta)] = \frac{1}{S}\sum_{s=1}^{S} f(\theta^{(s)})$$

- The law of large numbers still applies, i.e., $\hat{\mu}_f \xrightarrow{p} \mathbb{E}_{p(\theta|y)}[f(\theta)]$
- However, positive dependence makes the estimator less accurate for given S
- $\text{Var}(\hat{\mu}_f)$ is usually larger when using dependent draws compared to the independent case
- We will come back to this issue when we look at autocorrelation as a measure of dependence

Gibbs sampling

- One of the simplest and most powerful MCMC algorithms is *Gibbs sampling*
- Idea: alternate sampling from conditional posteriors

Gibbs sampling

- One of the simplest and most powerful MCMC algorithms is *Gibbs sampling*
- Idea: alternate sampling from conditional posteriors
- 2D example: Want to sample from $p(\theta_1, \theta_2|y)$. At s-th draw:
  1. Draw $\theta_1^{(s)}$ from $p(\theta_1|\theta_2^{(s-1)}, y)$
  2. Draw $\theta_2^{(s)}$ from $p(\theta_2|\theta_1^{(s)}, y)$

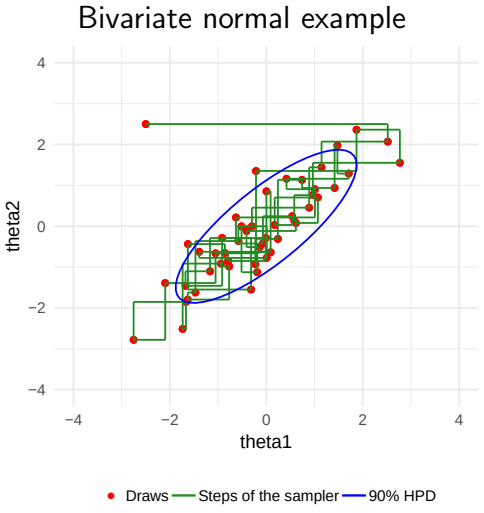
Gibbs sampling

- One of the simplest and most powerful MCMC algorithms is *Gibbs sampling*
- Idea: alternate sampling from conditional posteriors
- 2D example: Want to sample from $p(\theta_1, \theta_2|y)$. At s-th draw:
  1. Draw $\theta_1^{(s)}$ from $p(\theta_1|\theta_2^{(s-1)}, y)$
  2. Draw $\theta_2^{(s)}$ from $p(\theta_2|\theta_1^{(s)}, y)$
- In general, to sample from $p(\theta_1, \ldots, \theta_p|y)$, at the s-th draw:

$$\text{Draw } \theta_j^{(s)} \text{ from } p(\theta_j|\theta_{-j}^{(s-1)})$$

where $\theta_{-j}^{(s-1)} = (\theta_1^{(s)}, \ldots, \theta_{j-1}^{(s)}, \theta_{j+1}^{(s-1)}, \ldots, \theta_p^{(s-1)})$. Repeat for $j = 1, \ldots, p$

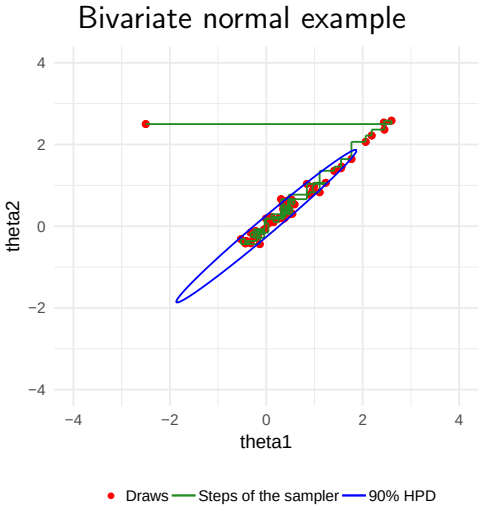
Gibbs sampling



Gibbs sampling

- With *conditionally* conjugate priors, sampling from the conditional distributions is easy for wide range of models (even if joint posterior is not closed-form)
- No algorithm parameters to tune (e.g., proposal distributions)
- If conditionals are not known analytically, can use any other sampling method that we have discussed
- *Blocking*: several parameters can be updated in blocks (e.g., all coefficients in linear regression)
- Slow if parameters are highly correlated in the posterior

Gibbs sampling



Metropolis algorithm

- Gibbs sampler can be used if the conditional posteriors are available
- In many applications, they are not; e.g., linear regression with general covariance matrices

Metropolis algorithm

- Gibbs sampler can be used if the conditional posteriors are available
- In many applications, they are not; e.g., linear regression with general covariance matrices
- *Metropolis algorithm* can be used in such cases
- Idea: sample from a proposal distribution depending on previous step and accept with a carefully constructed probability

Metropolis algorithm

1. Set a starting point $\theta^{(0)}$
2. For $s = 1, 2, \ldots, S$:
   2.1 Draw a candidate θ^* from a proposal distribution $g_s(\theta^*|\theta^{(s-1)})$, which is assumed to be symmetric; i.e., for all θ_a, θ_b

$$g_s(\theta_a|\theta_b) = g_s(\theta_b|\theta_a)$$

   2.2 Calculate the acceptance probability

$$\alpha(\theta^*, \theta^{(s-1)}) = \min\left\{\frac{p(\theta^*|y)}{p(\theta^{(s-1)}|y)}, 1\right\}$$

   2.3 Set

$$\theta^{(s)} = \begin{cases} \theta^* & \text{with probability } \alpha(\theta^*, \theta^{(s-1)}) \\ \theta^{(s-1)} & \text{otherwise} \end{cases}$$

Metropolis algorithm

- Note that if $p(\theta^*|y) > p(\theta^{(s-1)}|y)$, then $\alpha(\theta^*, \theta^{(s-1)}) = 1$
- That is, all moves that go to a place with higher probability mass are accepted
- There is still chance of moving to a place with less probability mass. This is done with probability $p(\theta^*|y)/p(\theta^{(s-1)}|y)$
- As before, step 2.3 is implement by drawing u from $\mathcal{U}(0,1)$ and accepting when $u \leq \alpha(\theta^*, \theta^{(s-1)})$
- Note that the normalization constant cancels out of the acceptance probability, so only the kernel is needed!

$$\frac{p(\theta^*|y)}{p(\theta^{(s-1)}|y)} = \frac{p(y|\theta^*)p(\theta^*)}{p(y|\theta^{(s-1)})p(\theta^{(s-1)})}$$

Metropolis algorithm: Details

Why does the Metropolis algorithm work?

- Intuitively: more draws from the higher density areas as jumps to higher density are always accepted and only some of the jumps to the lower density are accepted

Metropolis algorithm: Details

Why does the Metropolis algorithm work?

- Intuitively: more draws from the higher density areas as jumps to higher density are always accepted and only some of the jumps to the lower density are accepted
- Theoretically:
  1. Prove that simulated series is a Markov chain which has unique stationary distribution
  2. Prove that this stationary distribution is the desired target distribution $p(\theta|y)$

Metropolis algorithm: Details

1. To prove that a simulated series is a Markov chain with a unique stationary distribution we must show the chain has 3 properties
   (i) Irreducible

   (ii) Aperiodic

   (iii) (Positive) Recurrent

Metropolis algorithm: Details

1. To prove that a simulated series is a Markov chain with a unique stationary distribution we must show the chain has 3 properties
   - (i) Irreducible
     - Positive probability of eventually reaching any state from any other state
   - (ii) Aperiodic
     - It does not visit a set of states with a given frequency
   - (iii) (Positive) Recurrent
     - Intuitively: probability of returning to any state is 1 (equivalent to saying that all states are visited infinitely often)
     - Technically: Expected number of steps to reach back a state is finite

Metropolis algorithm: Details

2. To prove that the stationary distribution is $p(\theta|y)$:
   - Derive the algorithm's transition probability to go from $\theta^{(s-1)}$ to $\theta^{(s)}$:

$$p(\theta^{(s)}|\theta^{(s-1)}) = g_s(\theta^{(s)}|\theta^{(s-1)})\alpha(\theta^{(s)}, \theta^{(s-1)})$$

   - Let θ_a and θ_b be distributed according to $p(\theta|y)$ with $p(\theta_b|y) \geq p(\theta_a|y)$
   - Then, we can get the unconditional transition probabilities as

$$\begin{aligned} p(\theta^{(s)} = \theta_b, \theta^{(s-1)} = \theta_a) &= g_s(\theta_b|\theta_a)\alpha(\theta_b, \theta_a)p(\theta_a|y) \\ &= g_s(\theta_b|\theta_a)p(\theta_a|y) \end{aligned}$$

Metropolis algorithm: Details

2. To prove that the stationary distribution is $p(\theta|y)$:
   - Derive the algorithm's transition probability to go from $\theta^{(s-1)}$ to $\theta^{(s)}$:

$$p(\theta^{(s)}|\theta^{(s-1)}) = g_s(\theta^{(s)}|\theta^{(s-1)})\alpha(\theta^{(s)}, \theta^{(s-1)})$$

   - Let θ_a and θ_b be distributed according to $p(\theta|y)$ with $p(\theta_b|y) \geq p(\theta_a|y)$
   - Then, we can get the unconditional transition probabilities as

$$\begin{aligned} p(\theta^{(s)} = \theta_a, \theta^{(s-1)} = \theta_b) &= g_s(\theta_a|\theta_b)\frac{p(\theta_a|y)}{p(\theta_b|y)}p(\theta_b|y) \\ &= g_s(\theta_b|\theta_a)p(\theta_a|y) \end{aligned}$$

Metropolis algorithm: Details

2. To prove that the stationary distribution is $p(\theta|y)$:
   - Derive the algorithm's transition probability to go from $\theta^{(s-1)}$ to $\theta^{(s)}$:

$$p(\theta^{(s)}|\theta^{(s-1)}) = g_s(\theta^{(s)}|\theta^{(s-1)})\alpha(\theta^{(s)}, \theta^{(s-1)})$$

   - Let θ_a and θ_b be distributed according to $p(\theta|y)$ with $p(\theta_b|y) \geq p(\theta_a|y)$
   - Then, we can get the unconditional transition probabilities as

$$\begin{aligned} p(\theta^{(s)} = \theta_b, \theta^{(s-1)} = \theta_a) &= g_s(\theta_b|\theta_a)\alpha(\theta_b, \theta_a)p(\theta_a|y) \\ &= g_s(\theta_b|\theta_a)p(\theta_a|y) \\ p(\theta^{(s)} = \theta_a, \theta^{(s-1)} = \theta_b) &= g_s(\theta_a|\theta_b)\frac{p(\theta_a|y)}{p(\theta_b|y)}p(\theta_b|y) \\ &= g_s(\theta_b|\theta_a)p(\theta_a|y) \end{aligned}$$

   - We have shown that the joint distribution of $\theta^{(s)}$ and $\theta^{(s-1)}$ is symmetric

Metropolis algorithm: Details

2. To prove that the stationary distribution is $p(\theta|y)$:
   - We have shown

$$p(\theta^{(s)} = \theta_b, \theta^{(s-1)} = \theta_a) = p(\theta^{(s)} = \theta_a, \theta^{(s-1)} = \theta_b)$$

   - As the joint is symmetric, we can show

$$p(\theta^{(s)}) = p(\theta^{(s-1)})$$

   - Remember we started with $p(\theta^{(s)} = \theta) = p(\theta|y)$ for all s
   - This is the definition of a stationary distribution!

Metropolis-Hastings algorithm

Metropolis-Hastings

Generalization of Metropolis algorithm for non-symmetric proposal distributions.

- We no longer require $g_s(\theta_a|\theta_b) = g_s(\theta_b|\theta_a)$
- To make everything work, we modify the acceptance rate:

$$\alpha(\theta^*, \theta^{(s-1)}) = \min\left\{\frac{p(\theta^*|y)g_s(\theta^{(s-1)}|\theta^*)}{p(\theta^{(s-1)}|y)g_s(\theta^*|\theta^{(s-1)})}, 1\right\}$$

- Acceptance ratio now includes ratio of proposal distributions
- Order of evaluation of ratio of targets (posteriors) is opposite that of the proposals!

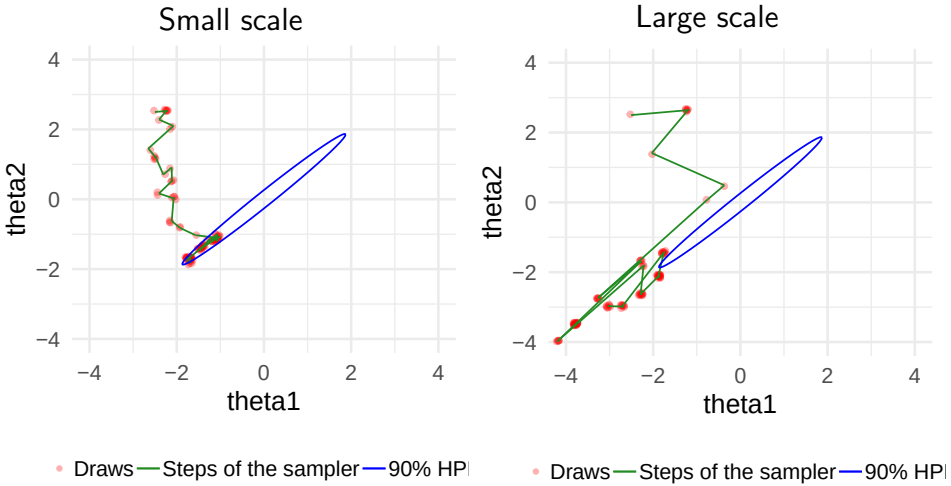
Metropolis-Hastings algorithm: Proposal choice

- Ideal proposal is the posterior itself: $g(\theta^*|\theta) = p(\theta^*|y)$ for all θ
  - Acceptance probability is 1
  - Draws are independent
  - Not feasible
- Good proposal distribution resembles the target distribution
- In practice, we usually use a normal or t distribution with moments selected to match the shape of the posterior

Metropolis-Hastings algorithm: Proposal choice

- After shape has been selected, it is important to select the scale (variance)
  - Small scale $\Rightarrow$ many steps accepted, but the chain moves slowly due to small steps
  - Large scale $\Rightarrow$ long steps proposed, but many of those rejected and again chain moves slowly
- Generic rule for rejection rate is 60-90% (but depends on dimensionality and a specific algorithm variation)
- Can use blocking and all other ideas that apply to Gibbs sampling

Metropolis-Hastings algorithm: Proposal choice



Gibbs sampling: Digression

We can now show that Gibbs sampling is a special case of the Metropolis-Hastings algorithm

- Imagine there were only two parameters θ_1 and θ_2
- Idea: use Metropolis-Hastings to draw from conditional posteriors $p(\theta_1|\theta_2, y)$ and $p(\theta_2|\theta_1, y)$
- Proposal distributions are conditional posteriors:

$$g(\theta_1^{(s)}|\theta_1^{(s-1)}, \theta_2) = p(\theta_1^{(s)}|\theta_2, y)$$
$$g(\theta_2^{(s)}|\theta_2^{(s-1)}, \theta_1) = p(\theta_2^{(s)}|\theta_1, y)$$

- As before, we know this means acceptance probability is 1
- It also proves the validity of Gibbs sampling

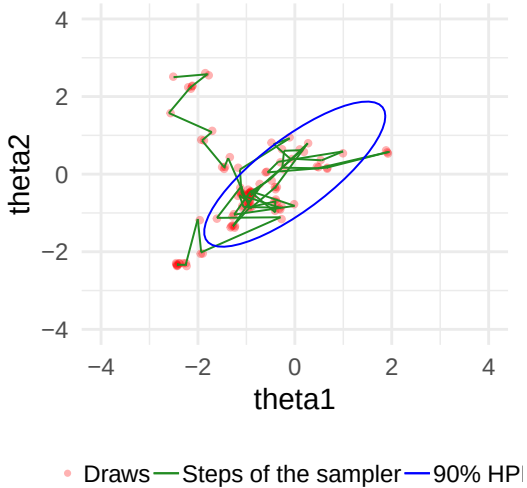
Practical considerations: Burn-in and Thinning

- Asymptotically, chain spends the $\alpha\%$ of time where $\alpha\%$ posterior mass is
- However, in finite time, the initial part of the chain may not be representative of the target distribution
- The simplest solution is to discard initial draws of the chain
- The draws we discard are known as the *burn-in* or warm-up draws
- Can also discard draws to deal with autocorrelation, process known as *thinning* (will return to this shortly)
- Cost is increased simulation time

Practical considerations: Convergence

- There is also the issue of *convergence*: can we confidently say we are finally obtaining draws from the target distribution?
- Convergence diagnostics: akin to hypothesis tests for convergence
  1. Geweke: test for equality of means of first and last parts of chain
  2. Raftery and Lewis: sample required to estimate a pre-specified quantile with given accuracy
  3. Heidelberger and Welch: two-part test of stationarity and achieved accuracy
  4. Gelman and Rubin: Potential Scale Reduction Factor for multiple chain output (see below)
- Keep in mind: passing a diagnostic $\neq$ convergence

Burn-in



Dependence measures: Autocorrelation

- Remember that draws from MCMC methods are dependent
- As we saw, positive dependence introduces a larger variance and thus less information per draw (compared to independent draws)
- We can use the same tools as in time series analysis to examine the extent of dependence in a chain
- The main tool for this objective is the *autocorrelation* function
  - Describes the correlation between $\theta^{(s)}$ and $\theta^{(s-j)}$ given a certain lag j
  - Can be used to compare efficiency of MCMC algorithms and parameterizations

UofG

Dependence measures: Autocorrelation

- Effective sample size:

$$S_{\text{eff}} = \frac{S}{\tau}$$

- τ captures the sum of autocorrelations
- With positive autocorrelations (the usual scenario), then $\tau > 1$
- We need τ draws to obtain one independent draw of information

Dependence measures: Autocorrelation

- Effective sample size:

$$S_{\text{eff}} = \frac{S}{\tau}$$

- τ captures the sum of autocorrelations
- With positive autocorrelations (the usual scenario), then $\tau > 1$
- We need τ draws to obtain one independent draw of information
- In practice, as we control S, then we can run chains for longer
- We can also *thin* the draws by choosing only every t draws (e.g., for $t = 5$, use draws $1, 6, 11, \ldots$)
- Thinning discards information but mitigates autocorrelation
- Optimally, you should choose t to be the largest lag that still creates autocorrelation problems

Monte Carlo integration with dependent draws: Details

- For simplicity, focus on estimating the mean $\mathbb{E}_{p(\theta|y)}[\theta]$ so choose $f(\theta) = \theta$,

$$\hat{\mu}_f = \bar{\theta} = \frac{1}{S}\sum_{s=1}^{S} \theta^{(s)}$$

- The Markov chain properties guarantee stationarity:
  1. $\text{Var}(\theta^{(s)}) = \text{Var}(\theta^{(t)})$ for all $s \neq t$
  2. $\text{Cov}(\theta^{(s)}, \theta^{(t)}) = \text{Cov}(\theta^{(i)}, \theta^{(j)})$ when $|s - t| = |i - j|$
- With independent draws: variance of sum = sum of variances

$$\text{Var}(\bar{\theta}) = \frac{\text{Var}(\theta^{(s)})}{S}$$

- With dependent draws, we can show

$$\text{Var}(\bar{\theta}) = \frac{\text{Var}(\theta^{(s)})}{S} \cdot \tau$$

Monte Carlo integration with dependent draws: Details

- τ is the sum of autocorrelations

$$\tau = 1 + 2\sum_{s=1}^{S-1}(S-s)\,\text{Corr}(\theta^{(1)}, \theta^{(s+1)})$$

- $\tau > 1 \Rightarrow$ Variance with dependence $>$ Variance with independence
- For a Markov chain, you have $\text{Corr}(\theta^{(1)}, \theta^{(s+1)}) = \rho^s$ for $\rho \in (-1, 1)$

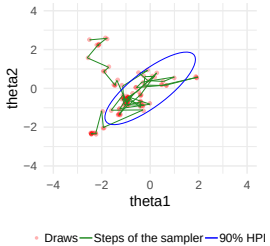
Monte Carlo integration with dependent draws: Details

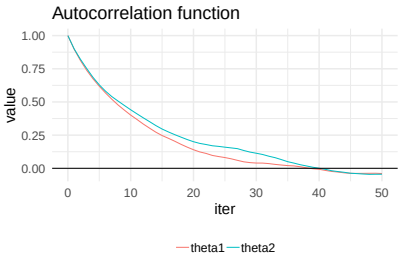
- τ is the sum of autocorrelations

$$\tau = 1 + 2\sum_{s=1}^{S-1}(S-s)\,\text{Corr}(\theta^{(1)}, \theta^{(s+1)})$$

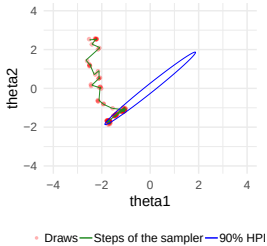
- $\tau > 1 \Rightarrow$ Variance with dependence $>$ Variance with independence
- For a Markov chain, you have $\text{Corr}(\theta^{(1)}, \theta^{(s+1)}) = \rho^s$ for $\rho \in (-1, 1)$
- We can show that $\tau > 1 \iff \rho > 0$ and that τ increases as ρ increases
  1. Positive dependence increases variance of estimated quantities
  2. The larger the dependence, the larger the increase
  3. The choice of proposal distribution affects the dependence introduced into the chains

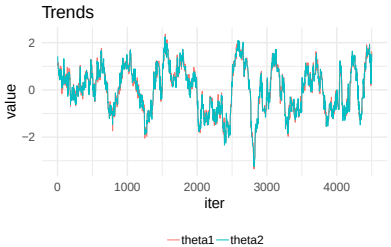
Autocorrelation: Metropolis (expected behaviour)

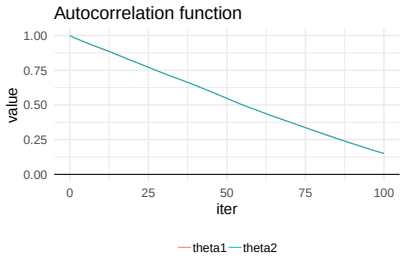




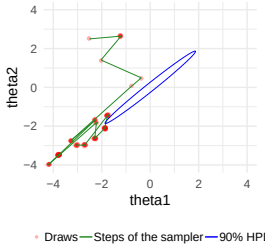
Autocorrelation: Metropolis with small steps

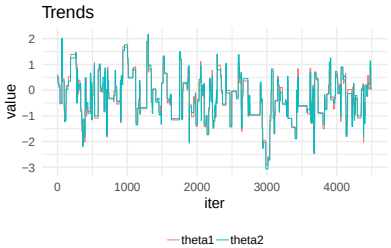


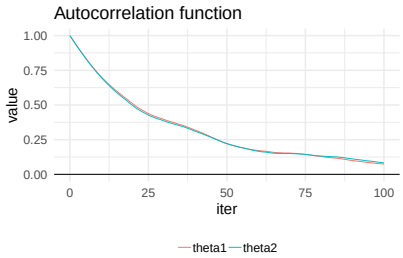




Autocorrelation: Metropolis with large steps



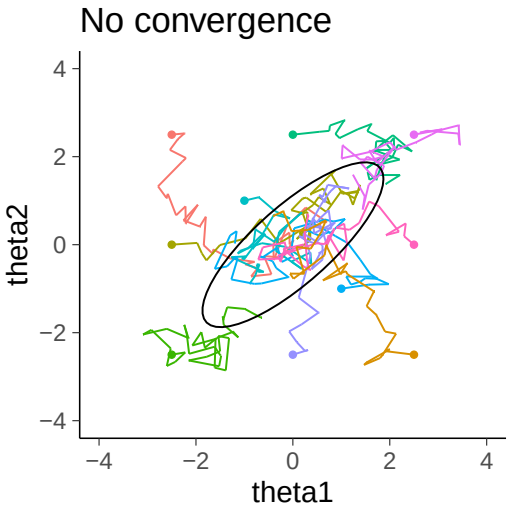




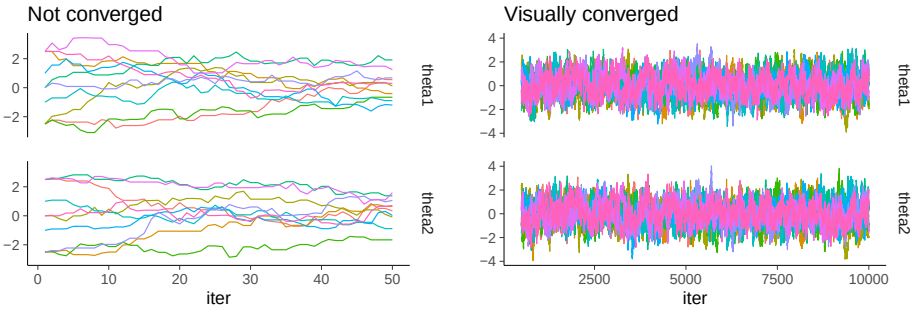
Several chains

- Use of several chains make convergence diagnostics easier
- Start chains from different starting points by changing $\theta^{(0)}$
- Preferably, use overdispersed initial values (e.g., from a distribution with larger variance that target)
- Remove draws from the beginning of the chains and run chains long enough so that it is not possible to distinguish where each chain started
- That is, try to get the chains to be well *mixed*
- Keep in mind, visual checks are not generally sufficient!

Several chains



Several chains



Diagnostic measure: $\widehat{R}$

- One of the first and most used measures for chain convergence is the *Potential Scale Reduction Factor* (PSRF) or $\widehat{R}$
- Idea: measure convergence of the chains (to the same target) by testing for equality of means
- Technically: measure the degree to which variance of the means between chains exceeds what one would expect if the chains were identically distributed

Diagnostic measure: $\widehat{R}$

- $\widehat{R}$ is computed by taking the total variance and dividing it by within chain variance W

$$\widehat{R} = \sqrt{\frac{\widehat{\text{Var}}^{+}}{W}}$$

- $\widehat{R} \to 1$, when $S \to \infty$

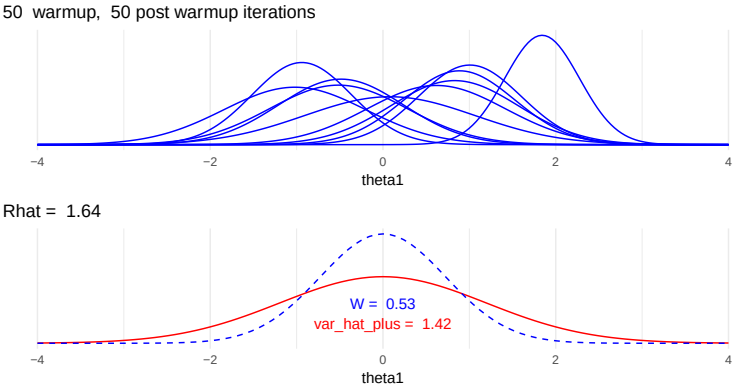
Diagnostic measure: $\widehat{R}$

- $\widehat{R}$ is computed by taking the total variance and dividing it by within chain variance W

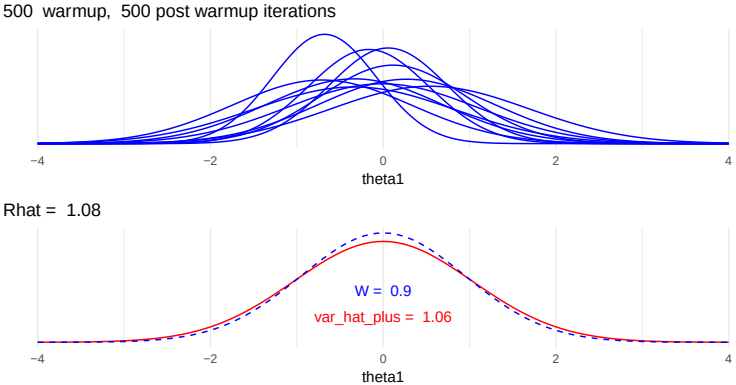
$$\widehat{R} = \sqrt{\frac{\widehat{\text{Var}}^{+}}{W}}$$

- $\widehat{R} \to 1$, when $S \to \infty$
- If $\widehat{R}$ is large (e.g., $R > 1.01$), keep sampling
- If $\widehat{R}$ close to 1, it is still possible that chains have not converged
  - If starting points were not overdispersed
  - Distribution far from normal (especially if infinite variance)
  - Just by chance as S is finite in practice
- This measure can be augmented to diagnose stationarity and to deal with large variance in the chains

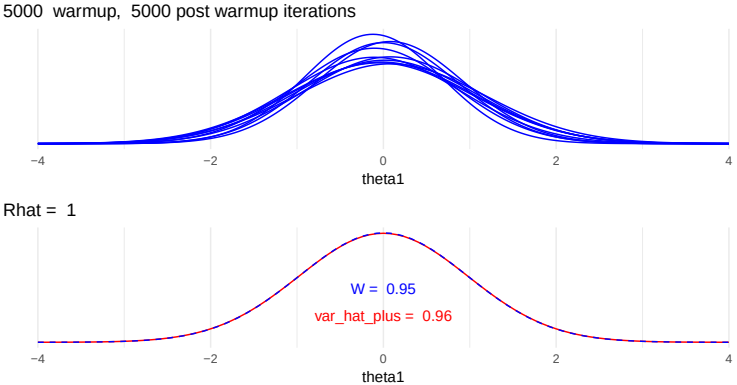
Diagnostic measure: $\widehat{R}$



Diagnostic measure: $\widehat{R}$



Diagnostic measure: $\widehat{R}$



$\widehat{R}$: Details

- M chains, each having S draws
- Define

$$\bar{\theta}_m := \frac{1}{S}\sum_{s=1}^{S}\theta_{sm}\,, \quad \bar{\theta} := \frac{1}{M}\sum_{m=1}^{M}\bar{\theta}_m\,, \quad \text{and} \quad s_m^2 := \frac{1}{S-1}\sum_{s=1}^{S}(\theta_{sm}-\bar{\theta}_m)^2$$

- Within chains variance W

$$W = \frac{1}{M}\sum_{m=1}^{M} s_m^2$$

- Between chains variance B

$$B = \frac{S}{M-1}\sum_{m=1}^{M}(\bar{\theta}_m-\bar{\theta})^2$$

$\widehat{R}$: Details

- B/S is variance of the means of the chains
- Estimate total variance $\text{Var}(\theta|y)$ as a weighted mean of W and B

$$\widehat{\text{Var}}^{+}(\theta|y) = \frac{S-1}{S}W + \frac{1}{S}B$$

- This *overestimates* marginal posterior variance if the starting points are overdispersed
- Given finite S, W *underestimates* marginal posterior variance as single chains have not yet visited all points in the distribution
- However, as $S \to \infty$, $\quad \mathbb{E}(W) \to \text{Var}(\theta|y)$
- As $\widehat{\text{Var}}^{+}(\theta|y)$ overestimates and W underestimates, compute

$$\widehat{R} = \sqrt{\frac{\widehat{\text{Var}}^{+}}{W}}$$

Problematic distributions

- Nonlinear dependencies
  - optimal proposal depends on location
- Funnels
  - optimal proposal depends on location
- Multimodal
  - difficult to move from one mode to another
- Long-tailed with non-finite variance and mean
  - central limit theorem for expectations does not hold