Diagnostics in MCMC

Hoff Chapter 7

October 7, 2009

Convergence to Posterior Distribution

Theory tells us that if we run the Gibbs sampler long enough the samples we obtain will be samples from the joint posterior distribution (target or stationary distribution). This does not depend on the starting point (forgets the past).

- How long do we need to run the Markov Chain to adequately explore the posterior distribution?
- Mixing of the chain plays a critical role in how fast we can obtain good results. How can we tell if the chain is mixing well (or poorly)?

Three Component Mixture Model

Posterior for μ:

$$\mu \mid Y \sim 0.45\,\mathrm{N}(-3, 1/3) + 0.10\,\mathrm{N}(0, 1/3) + 0.45\,\mathrm{N}(3, 1/3)$$

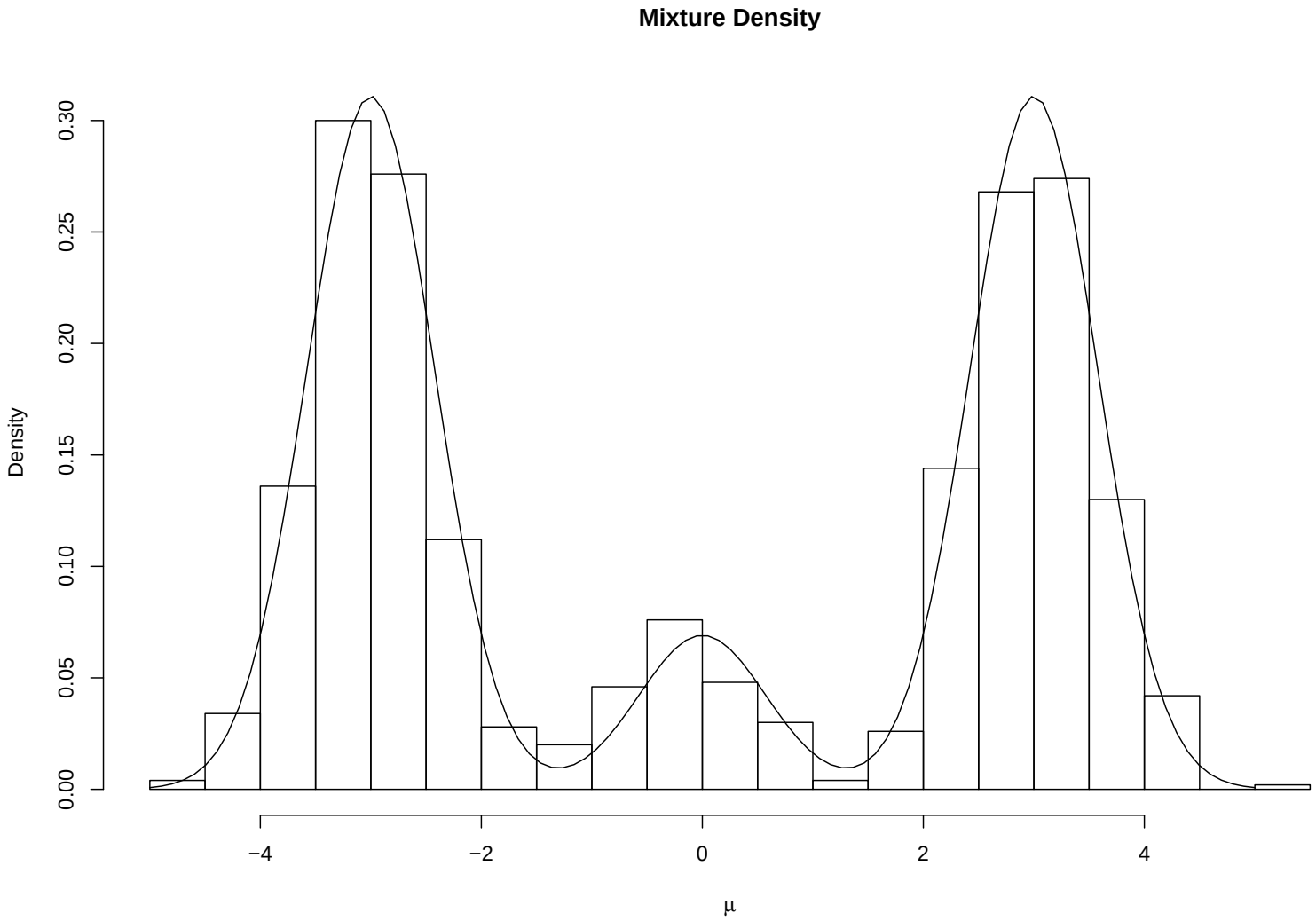
How can we draw samples from the posterior?

Introduce "mixture component indicator" δ, an unobserved latent variable which simplifies sampling

- $\delta = 1$ then $\mu \mid \delta, Y \sim \mathrm{N}(-3, 1/3)$ and $P(\delta = 1 \mid Y) = 0.45$
- $\delta = 2$ then $\mu \mid \delta, Y \sim \mathrm{N}(0, 1/3)$ and $P(\delta = 2 \mid Y) = 0.10$;
- $\delta = 3$ then $\mu \mid \delta, Y \sim \mathrm{N}(3, 1/3)$ and $P(\delta = 3 \mid Y) = 0.45$

Monte Carlo sampling: Draw δ; Given δ, draw μ

MC density



Histogram of μ from 1000 MC draws with posterior density as a solid line

MC Variation

If we want to find the posterior mean of $g(\mu)$, then the Monte Carlo estimate based on M MC samples is

$$\hat{g}_{MC} = \frac{1}{M} \sum_{m} g(\mu^{(m)}) \rightarrow \mathrm{E}[g(\mu) \mid Y]$$

with variance

$$\mathrm{Var}[\hat{g}_{MC}] = \mathrm{E}[\hat{g}_{MC} - \mathrm{E}[\hat{g}_{MC}]]^2 = \frac{\mathrm{Var}[g(\mu) \mid Y]}{M}$$

leading to Monte Carlo Standard Error $\sqrt{\mathrm{Var}[\hat{g}_{MC}]}$.

We expect the posterior mean of $g(\mu)$ should in the interval $\hat{g}_{MC} \pm 2\sqrt{\mathrm{Var}[\hat{g}_{MC}]}$ for roughly 95% of repeated MC samples.

To increase accuracy, increase M.

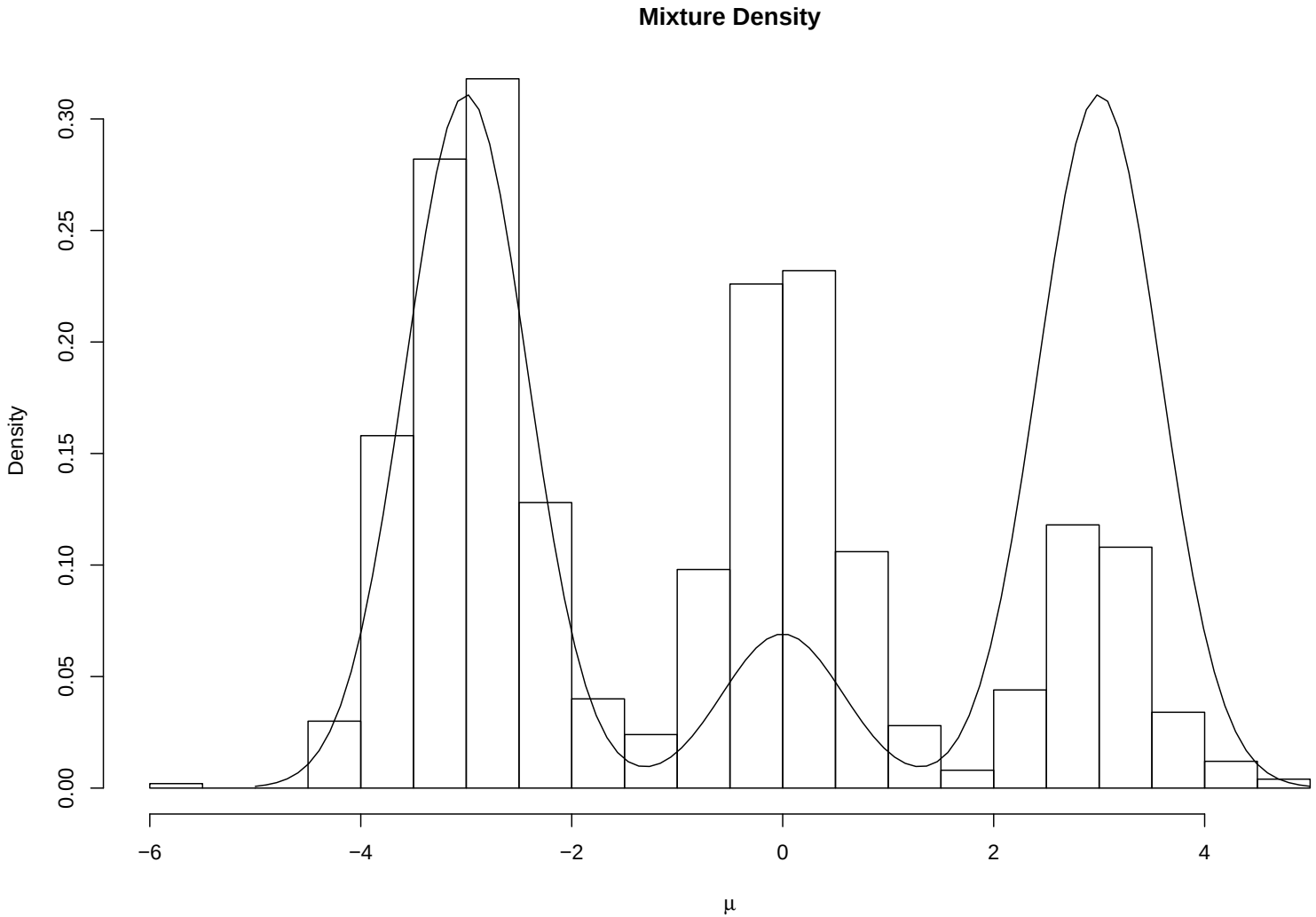
Markov Chain Monte Carlo

- Full conditional for $\mu \mid \delta, Y$
  - $\mu \mid \delta = 1, Y \sim \mathrm{N}(-3, 1/3)$
  - $\mu \mid \delta = 2, Y \sim \mathrm{N}(0, 1/3)$
  - $\mu \mid \delta = 3, Y \sim \mathrm{N}(3, 1/3)$
- Full conditional for $\delta \mid \mu, Y$ (use Bayes Theorem):

$$P(\delta = d \mid \mu, Y) = \frac{p(\delta = d \mid Y)\,\mathrm{dnorm}(\mu, m_d, s_d^2)}{\sum_{d=1}^{3} p(\delta = d \mid Y)\,\mathrm{dnorm}(\mu, m_d, s_d^2)}$$

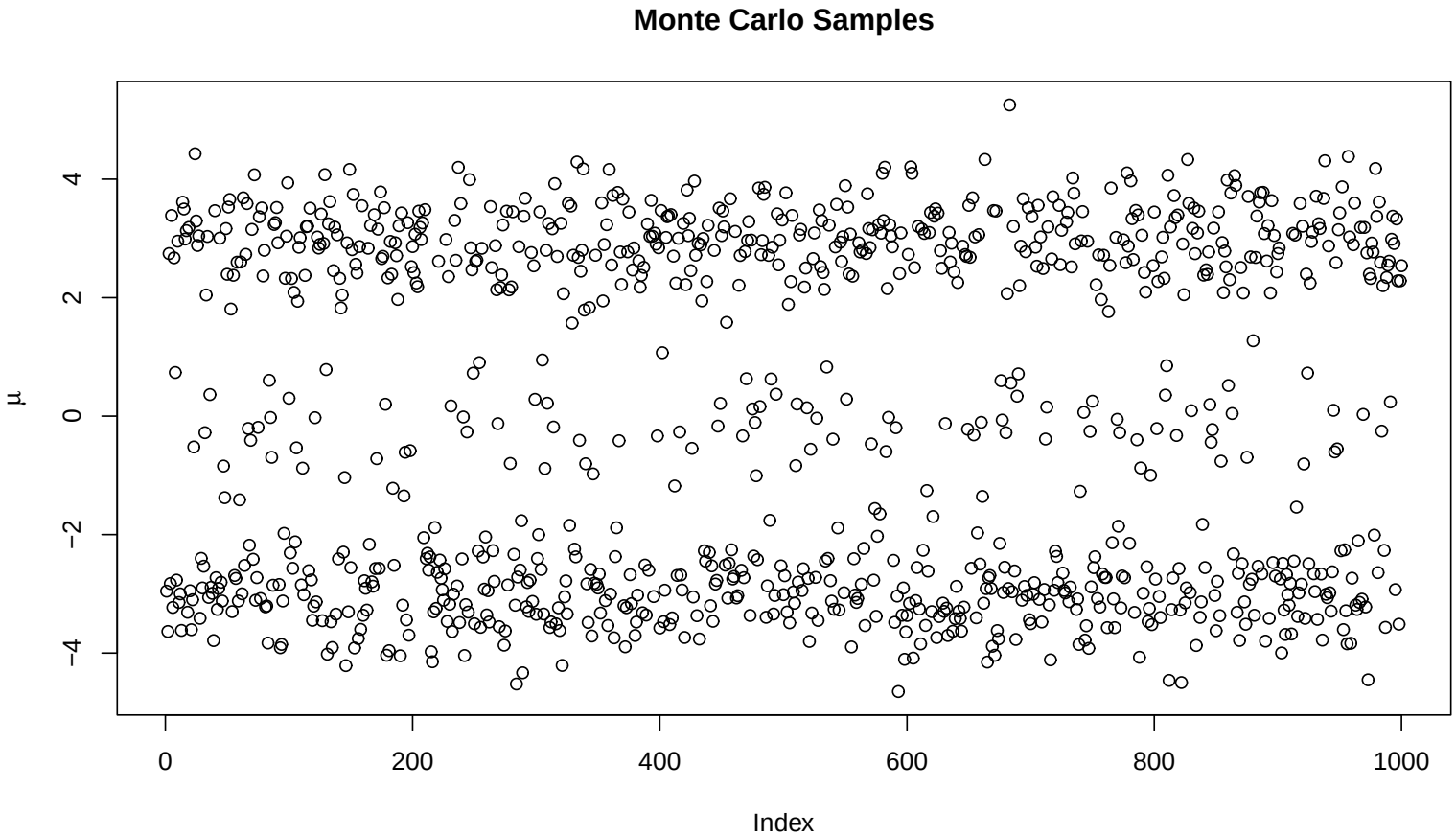
where dnorm is the normal density.

MCMC density

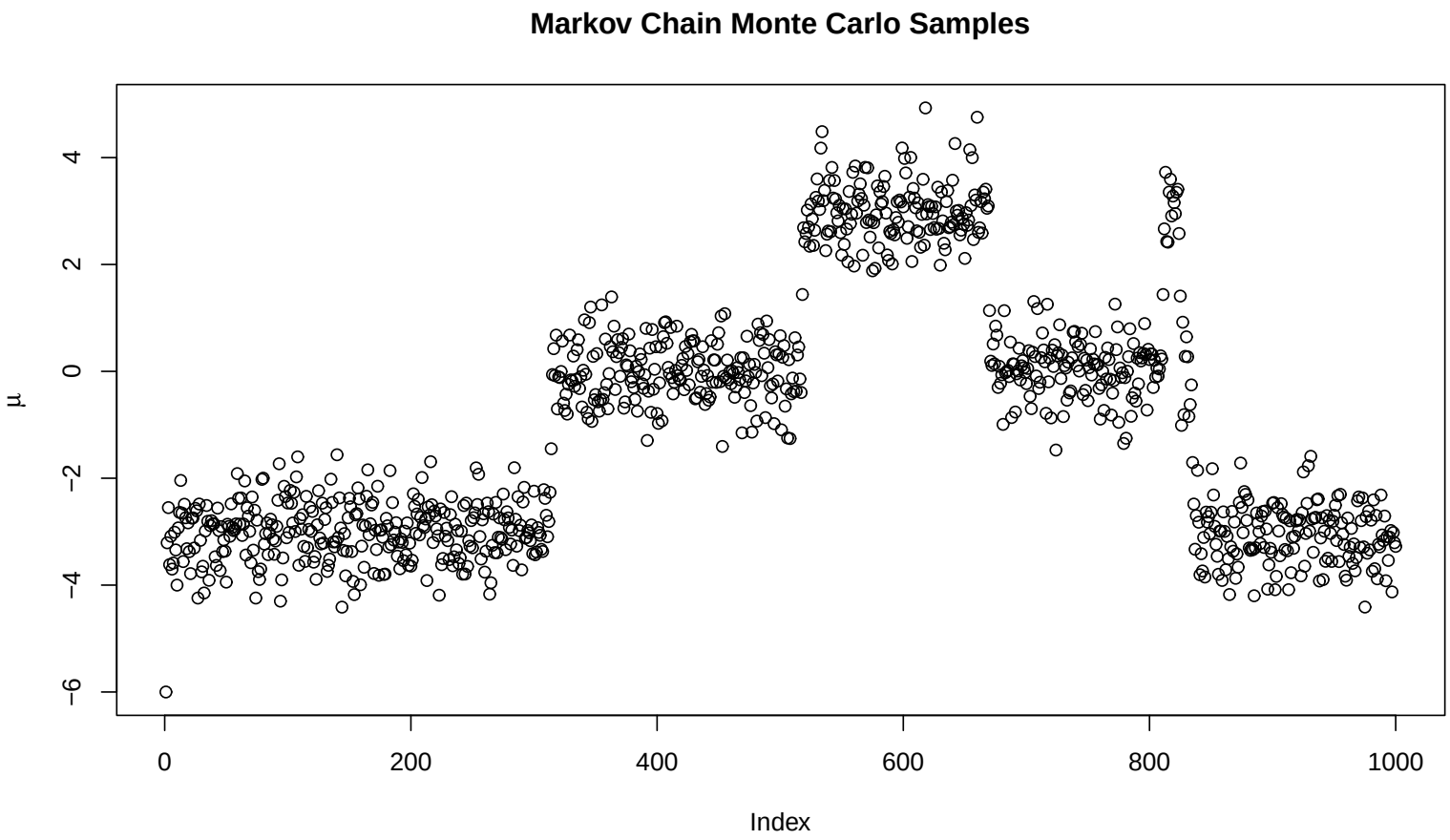


1000 draws using MCMC starting at $\mu = -6$ and $\delta = 1$

MC Trace Plots



MCMC Traceplot



Stationarity

Partition the parameter space as follows:

- $A_0 = (-\infty, -5)$
- $A_1 = (-5, -1)$
- $A_2 = (-1, 1)$
- $A_3 = (1, 5)$
- $A_4 = (5, \infty)$

Under the posterior, we should have
$P(A_3) = P(A_1) > P(A_2) > P(A_0) = P(A_4)$.
We need to have our MCMC sample size S to be big enough so that we can

1. Move out of A_2 (or another areas of low probability) into regions of high posterior regions (convergence)
2. Move between A_1 and A_3 and other high probability regions (mixing)

Stickiness

The traceplots show

- MC samples can move from one region to another in 1 step (perfect mixing)
- MCMC quickly moves away from the starting value -6
- MCMC has more difficulty moving from A_2 into higher probability regions
- MCMC has difficulty moving between the different components and tends to get "stuck" in one component for a while (stickiness)

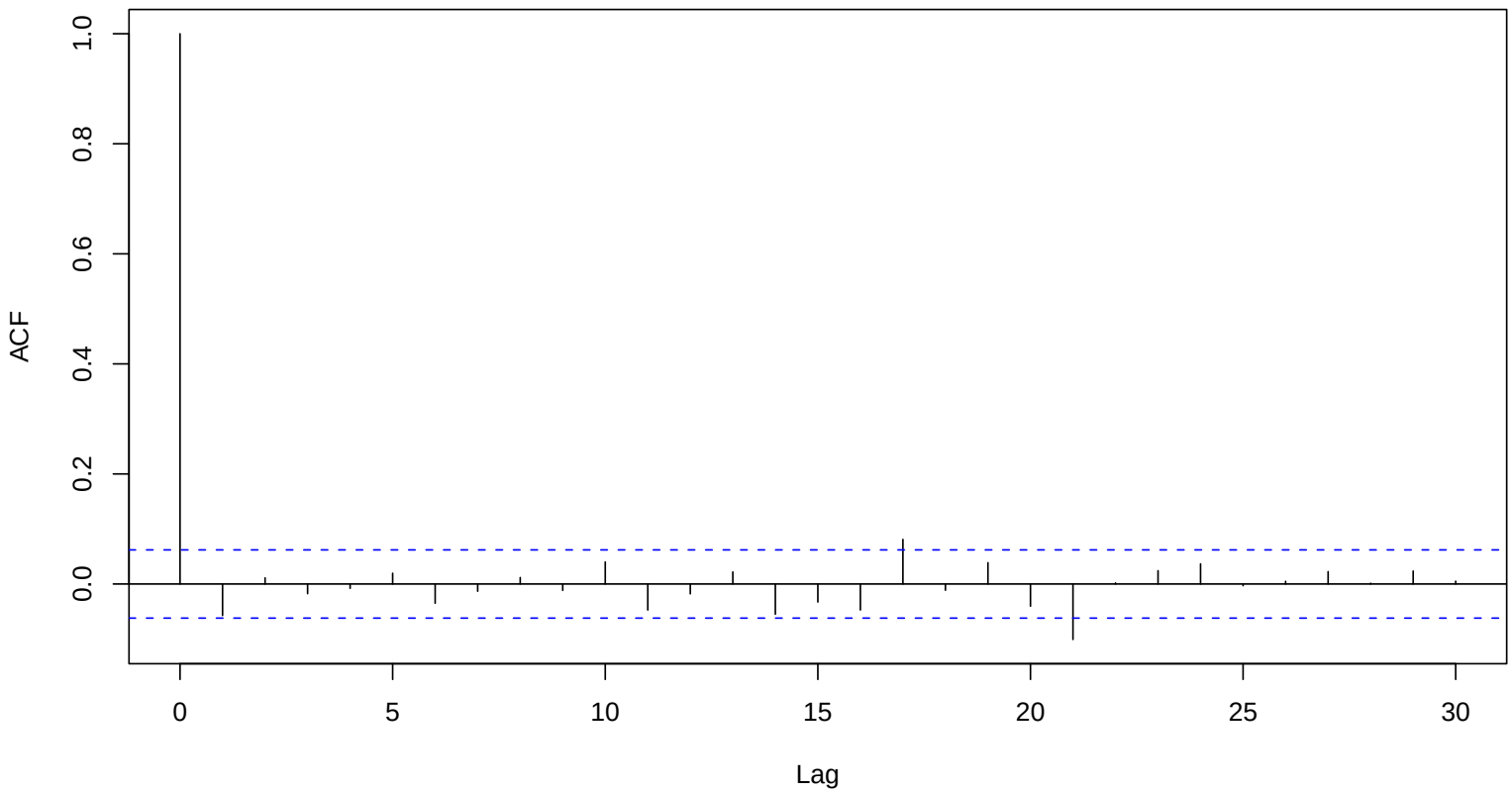
Consequences for Variance for MCMC estimates

$$\text{Var}_{\text{MCMC}}[\hat{g}] = \text{Var}_{\text{MC}}(\hat{g}) + \sum_{s \neq t} \text{E}\left[\left(g(\theta^{(s)}) - \text{E}[g(\theta)|Y]\right)\left(g(\theta^{(t)}) - \text{E}[g(\theta)|Y]\right)\right]$$

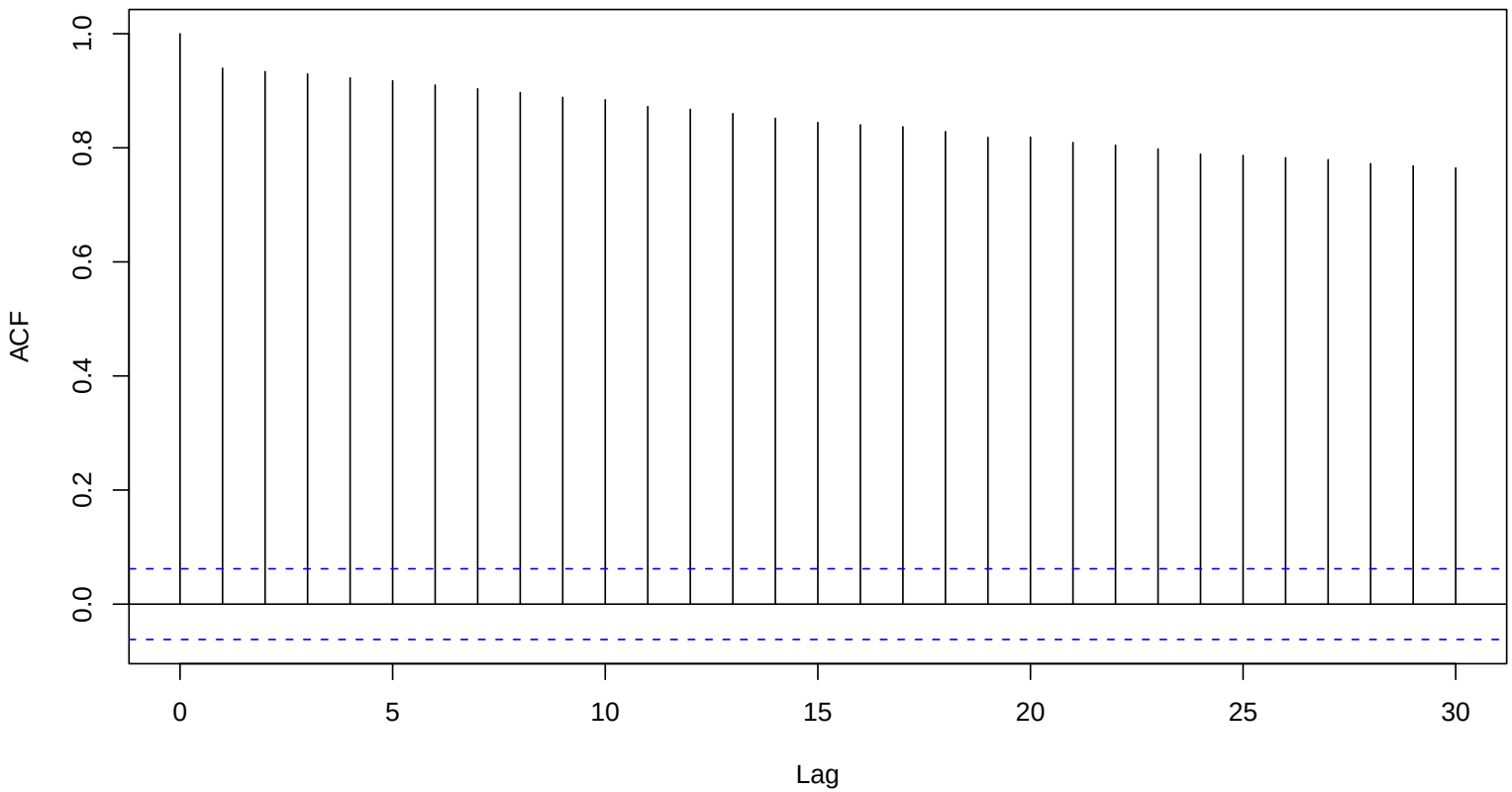
- The second term depends on the autocorrelation of samples within the Markov chain
- Autocorrelation of lag t is the correlation between $g(\theta)^{(s)})$ and $g(\theta)^{(s+t)})$ (elements that are t time steps apart)
- often positive so MCMC variance is larger than MC variance.
- high correlation is an indicator of poor mixing, so that we need a larger sample size to obtain a comparable variance
- Effective Sample Size

$$\text{Var}_{\text{MCMC}} = \frac{\text{Var}_{\text{MCMC}}}{S_{\text{eff}}}$$

ACF Plots Monte Carlo Sample



ACF Plots MCMC Sample



CODA

The CODA package provides many popular diagnostics for assessing convergence of MCMC output from WinBUGS (and other programs)

```
> library(coda)
> effectiveSize(theta.MCMC)
    var1     var2
3.444007 4.390776
```

The precision of the MCMC estimate of the posterior mean based on 1000 samples is a good as taking 4 independent samples!

Running for 1 million iterations, still has an effective sample size of 1927.

Useful Diagnostics/Functions

- Effective Sample Size: `effectiveSize()`
- Autocorrelation `autocorr.plot()`
- Cross Variable Correlations:`crosscorr.plot()`
- Geweke: `geweke.diag()`
- Gelman-Rubin: `gelman.diag()`
- Heidelberg & Welch: `heidel.diag()`
- Raftery-Lewis: `raftery.diag()`

Geweke

Geweke (1992) proposed a convergence diagnostic for Markov chains based on a test for equality of the means of the first and last part of a Markov chain (by default the first 10% and the last 50%).

If the samples are drawn from the stationary distribution of the chain, the two means are equal and Geweke's statistic has an asymptotically standard normal distribution.

Gelman-Rubin

Gelman and Rubin (1992) propose a general approach to monitoring convergence of MCMC output in which $m > 1$ parallel chains are run.

- Use different starting values that are overdispersed relative to the posterior distribution.
- Convergence is diagnosed when the chains have "forgotten" their initial values, and the output from all chains is indistinguishable.
- The diagnostic is applied to a single variable from the chain. It is based a comparison of within-chain and between-chain variances (similar to a classical analysis of variance)
- Assumes that the target is normal (transformations may help)
- Values of $\hat{R}$ near 1 suggest convergence

Heidelberg-Welch

The convergence test uses the Cramer-von-Mises statistic to test the null hypothesis that the sampled values come from a stationary distribution.

- The test is successively applied, firstly to the whole chain, then after discarding the first 10%, 20%, of the chain until either the null hypothesis is accepted, or 50% of the chain has been discarded.
- The latter outcome constitutes "failure" of the stationarity test and indicates that a longer MCMC run is needed.
- If the stationarity test is passed, the number of iterations to keep and the number to discard (burn-in) are reported.

Raftery-Lewis

Calculates the number of iterations required to estimate the quantile q to within an accuracy of $\pm r$ with probability p.

- Separate calculations are performed for each variable within each chain. If the number of iterations in data is too small, an error message is printed indicating the minimum length of pilot run.
- The minimum length is the required sample size for a chain with no correlation between consecutive samples. An estimate I (the 'dependence factor') of the extent to which autocorrelation inflates the required sample size is also provided.
- Values of I larger than 5 indicate strong autocorrelation which may be due to a poor choice of starting value, high posterior correlations or stickiness of the MCMC algorithm.
- The number of burn-in iterations to be discarded at the beginning of the chain is also calculated.

Summary

- Diagnostics cannot guarantee that chain has converged
- Can indicate that it has not converged

Solutions?

- Run longer and thin output
- Reparametrize model
- "Block" correlated variables together
- integrate out variables
- Add auxiliary variables (Slice-sampler for example)
- Use "Rao-Blackwellization" in estimation