Statistics for Data Scientists: Monte Carlo and MCMC Simulations

James M. Flegal

University of California, Riverside, CA

February 12, 2016

Introduction

- Monte Carlo Simulations
  - Introduction and Motivation
  - Examples
- Bayesian Statistics
- Markov chain Monte Carlo Simulations
  - Metropolis Hastings and Gibbs Sampling
  - Examples
  - Output Analysis and Stopping Rules

Monte Carlo Integration

Monte Carlo Integration

Example: Let $X \sim \Gamma(3/2, 1)$, i.e.

$$f(x) = \frac{2}{\sqrt{\pi}} \sqrt{x} e^{-x} I(x > 0).$$

Suppose we want to find

$$\begin{aligned} \theta &= \mathsf{E}\left[\frac{1}{(X+1)\log(X+3)}\right] \\ &= \int_0^\infty \frac{1}{(x+1)\log(x+3)} \frac{2}{\sqrt{\pi}} \sqrt{x} e^{-x} dx. \end{aligned}$$

The expectation (or integral) θ is intractable, we don't know how to compute it analytically.

Monte Carlo Integration

One possible solution is to approximate θ using Monte Carlo integration. If $Y_1, Y_2, \ldots$ are i.i.d. with $\mathrm{E}|Y_1| < \infty$ then

$$\bar{y} = \sum_{i=1}^{n} Y_i \overset{a.s.}{\to} \mathrm{E}Y_1 \quad \text{(SLLN)}.$$

Suppose $X_1, X_2, \ldots$ are i.i.d $\Gamma(3/2, 1)$ and define $Y_i = [(X_i + 1)\log(X_i + 3)]^{-1}$. Then since $\mathrm{E}|Y_1| < \infty$ we have

$$\sum_{i=1}^{n} [(X_i + 1)\log(X_i + 3)]^{-1} \overset{a.s.}{\to} \mathrm{E}\left[\frac{1}{(X + 1)\log(X + 3)}\right] = \theta.$$

Monte Carlo Integration

Thus is we had a way to "generate" or "simulate" or "draw" $\Gamma(3/2, 1)$ random variables, we could obtain a large number of them and claim

$$\sum_{i=1}^{n} \left[(X_i + 1) \log(X_i + 3)\right]^{-1} \approx \theta.$$

An obvious question is how good is this approximation?

Monte Carlo Standard Error

Suppose $Y_1, Y_2, \ldots$ are i.i.d. with $\mathrm{E}|Y_1^2| < \infty$ then the CLT says

$$\frac{\sqrt{n}(\bar{y}_n - \mathrm{E}Y_1)}{\sigma} \xrightarrow{d} \mathrm{N}(0,1).$$

That is, for sufficiently large n,

$$\bar{y}_n \sim \mathrm{N}(\mathrm{E}Y_1, \sigma^2/n).$$

Further, we can estimate the standard error $\sigma/\sqrt{n}$ with $s_n/\sqrt{n}$ where s_n is the sample standard deviation.

Monte Carlo Standard Error

We can also use the CLT form a confidence interval with

$$Pr(\bar{y}_n - 1.96 s_n/\sqrt{n} < \mathrm{E}Y_1 < \bar{y}_n + 1.96 s_n/\sqrt{n}) \approx 0.95.$$

Or we could simulate until a half-width (or width) of this confidence interval is sufficiently small, say less than $\epsilon > 0$. That is, simulate until

$$1.96 s_n/\sqrt{n} < \epsilon.$$

Toy Example

Example: Let $X \sim \Gamma(3/2, 1)$, i.e.

$$f(x) = \frac{2}{\sqrt{\pi}} \sqrt{x} e^{-x} I(x > 0).$$

Suppose we want to find

$$\begin{aligned} \theta &= \mathrm{E}\left[\frac{1}{(X+1)\log(X+3)}\right] \\ &= \int_0^\infty \frac{1}{(x+1)\log(x+3)} \frac{2}{\sqrt{\pi}} \sqrt{x} e^{-x} dx. \end{aligned}$$

Further, suppose we want to estimate this quantity such that a 95% CI length is less than 0.002.

Toy Example Code

```
set.seed(500)

########################################
## Monte Carlo Toy Example
########################################

n <- 1000
x <- rgamma(n, 3/2, scale=1)
mean(x)
y <- 1/((x+1)*log(x+3))
est <- mean(y)
est
mcse <- sd(y) / sqrt(length(y))
interval <- est + c(-1,1)*1.96*mcse
interval
```

Toy Example Code

```
## Implementing the sequential stopping rule
eps <- 0.002
len <- diff(interval)
plotting.var <- c(est, interval)
while(len > eps){
new.x <- rgamma(n, 3/2, scale=1)
new.y <- 1/((new.x+1)*log(new.x+3))
y <- cbind(y, new.y)
est <- mean(y)
mcse <- sd(y) / sqrt(length(y))
interval <- est + c(-1,1)*1.96*mcse
len <- diff(interval)
plotting.var <- rbind(plotting.var, c(est, interval))
}
```

Toy Example Code

```
## Plotting the results
temp <- seq(1000, length(y), 1000)
plot(temp, plotting.var[,1], type="l", ylim=c(min(plotting.var),
     max(plotting.var)), main="Estimates of the Mean", xlab="Iterations",
     ylab="Estimate")
points(temp, plotting.var[,2], type="l", col="red")
points(temp, plotting.var[,3], type="l", col="red")
legend("topright", legend=c("CI", "Estimate"), lty=c(1,1), col=c(2,1))
```

Toy Example

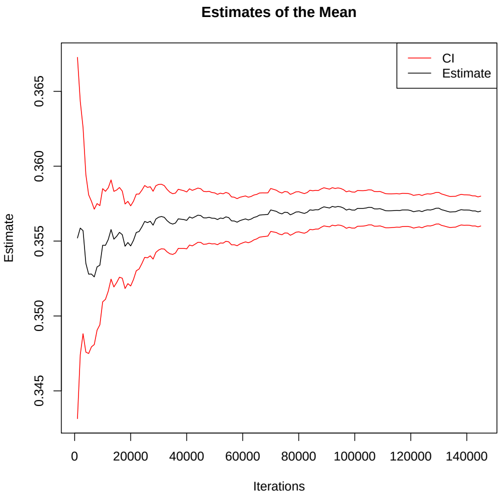


Figure: Results from one simulation using a cut-off of $\epsilon = 0.002$.

High-Dimensional Examples

- ▸ Link FiveThirtyEight's NBA Predictions
- ▸ Link Vanguard's Retirement Nest Egg Calculator
- ▸ Link Minitab's Monte Carlo Simulation Software for Manufacturing Engineers
- ▸ Link Fisher's Exact Test in R

Bayesian Statistics

Bayesian Statistics

- Suppose X has a distribution parameterized by θ.
- Let $f(\theta)$ be a density assigned to θ before observing any data. This density is call the prior distribution.
- Bayesian inference is driven by the likelihood, $L(\theta|x)$.

Bayesian Statistics

Starting with our prior, after observing data we can update our beliefs to form a posterior distribution (via Bayes Rule), i.e.

$$f(\theta|x) = cf(\theta)L(\theta|x)$$

where

$$c = \frac{1}{\int f(\theta)L(\theta|x)d\theta} \quad \text{(which is often difficult to compute).}$$

The posterior, $f(\theta|x)$ is used for Bayesian inference on θ.

Example: Suppose $X_1, \ldots, X_n$ are i.i.d. $N(\theta, \sigma^2)$ where σ^2 is known. Suppose further we have a prior $\theta \sim N(\mu, \tau^2)$. Then the posterior can be obtained as follows,

$$\begin{aligned} f(\theta|x) &\propto f(\theta) \prod_{i=1}^{n} f(x_i|\theta) \\ &\propto \exp\left\{-\frac{1}{2}\left(\frac{(\theta-\mu)^2}{\tau^2} + \frac{\sum_{i=1}^{n}(x_i-\theta)^2}{\sigma^2}\right)\right\} \\ &\propto \exp\left\{-\frac{1}{2}\frac{\left(\theta - \frac{\mu/\tau^2 + n\bar{x}/\sigma^2}{1/\tau^2 + n/\sigma^2}\right)^2}{\frac{1}{1/\tau^2 + n/\sigma^2}}\right\}. \end{aligned}$$

Or $f(\theta|x) \sim N(\mu_n, \tau_n^2)$ where

$$\mu_n = \left(\frac{\mu}{\tau^2} + \frac{n\bar{x}}{\sigma^2}\right)\tau_n^2 \quad \text{and} \quad \tau_n^2 = \frac{1}{1/\tau^2 + n/\sigma^2}.$$

Notice, this is a conjugate Bayes model. Also note a 95% credible region for θ is given by (this is also the HPD, highest posterior density)

$$(\mu_n - 1.96\tau_n, \mu_n + 1.96\tau_n).$$

For large n, the data will overwhelm the prior.

Bayesian Statistics

- If $f(\theta) \propto 1$, an improper prior, then a 95% credible region for θ is the same as a 95% confidence interval since $f(\theta|x) \sim N(\bar{x}, \sigma^2/n)$ (try to show this at home).
- Usually, we specify a prior and likelihood that result in an posterior that is intractable. That is, we can't work with it analytically or even calculate the appropriate normalizing constant c.
- However, it is often easy to simulate a Markov chain with $f(\theta|x)$ as its stationary distribution.

Markov Chain Basics

Consider discrete time, discrete state space Markov chains. If

$$P(X_{t+1} = j | X_0 = x_0, \ldots, X_t = i) = P(X_{t+1} = j | X_t = i) = p_{ij}$$

for all t, $x_0, \ldots, x_n \in S$, $i, j \in S$, then $\{X_t\}$ is a Markov chain (time homogeneous). This is governed by a Markov transition matrix

$$P = \begin{bmatrix} p_{00} & p_{01} & p_{02} & \cdots & p_{0n} \\ p_{10} & p_{11} & p_{12} & \cdots & p_{1n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{n0} & p_{n1} & p_{n3} & \cdots & p_{nn} \end{bmatrix} \quad \text{(rows sum to one).}$$

Markov Chain Basics

Limit theory of Markov chains is important.

- A state the chain returns to w.p.1 is recurrent.
  - If the expected time to return is finite, then non null.
  - If the expected time to return is infinite, then null recurrent.
- A chain is irreducible if for all i, j pairs there exists $m > 0$ such that $P(X_{m+n} = i | X_n = j) > 0$.
- A chain is periodic if it can only visit portions of the state space at regular intervals, d is the smallest divisor of the times.
- A chain is aperiodic if $d = 1$.

Markov Chain Basics

- A Markov chain is ergodic if it is irreducible, aperiodic, and all its states are non null and recurrent.
- Suppose π is such that $\pi P = \pi$, then π is the stationary (or invariant) distribution for P.
- If $\{X_t\}$ is irreducible and aperiodic, then π is unique and

$$\lim_{n\to\infty} P(X_{t+n} = j | X_t = i) = \pi_j.$$

- And for any function h

$$\frac{1}{n}\sum_{i=1}^{n} h(x_i) \stackrel{a.s.}{\to} E_\pi\left[h(X)\right].$$

This is the ergodic theorem, a generalization of the SLLN.

Markov Chain Monte Carlo

Markov Chain Monte Carlo

MCMC methods are used most often in Bayesian inference where f or π is a posterior distribution. Challenge lies in construction of a suitable Markov chain with f as its stationary distribution. A key problem is we only get to observe t observations from $\{X_t\}$, which are serially dependent.

Questions to Consider:

How good are my MCMC estimators?

How long to run my Markov chain simulation?

How to compare MCMC samplers?

What to do in high-dimensional settings?

...

Metropolis-Hastings Algorithm

Setting $X_0 = x_0$ (somehow), the Metropolis-Hastings algorithm generates X_{t+1} given $X_t = x_t$ as follows:

1. Sample a candidate value $X^* \sim g(\cdot|x_t)$ where g is the proposal distribution.
2. Compute the MH ratio $R(x_t, X^*)$, where

$$R(x_t, X^*) = \frac{f(x^*)g(x_t|x^*)}{f(x_t)g(x^*|x_t)}.$$

3. Set

$$X_{t+1} = \begin{cases} x^* \text{ w.p. } \min\{R(x_t, X^*), 1\} \\ x_t \text{ otherwise.} \end{cases}$$

Metropolis-Hastings Algorithm

- Irreducibility and aperiodicity depend on the choice of g, these must be checked.
- Performance (finite sample) depends on the choice of g also, be careful.

Independence MH Chains

Suppose $g(x^*|x_t) = g(x^*)$, this yields an independence chain since the proposal does not depend on the current state. In this case, the MH ratio is

$$R(x_t, X^*) = \frac{f(x^*)g(x_t)}{f(x_t)g(x^*)},$$

and the resulting Markov chain will be irreducible and aperiodic if $g > 0$ where $f > 0$.

A good envelope function g should resemble f, but should cover f in the tails.

Random Walk MH Chains

Generate X^* such that $\epsilon \sim h(\cdot)$ and set $X^* = X_t + \epsilon$. Then $g(x^*|x_t) = h(x^* - x_t)$. Common choices of $h(\cdot)$ are symmetric zero mean random variables with a scale parameter, e.g. a Uniform$(-a, a)$, Normal$(0, \sigma^2)$, $c * T_\nu, \ldots$

For symmetric zero mean random variables, the MH ratio is

$$R(x_t, X^*) = \frac{f(x^*)}{f(x_t)}.$$

If the support of f is connected and h is positive in a neighborhood of 0, then the chain is irreducible and aperiodic.

Exercise: Suppose $f \sim Exp(1)$.

1. Write an independence MH sampler with $g \sim Exp(\theta)$.
2. Show $R(x_t, X^*) = \exp\{(x_t - x^*)(1-\theta)\}$.
3. Generate 1000 draws from f with $\theta \in \{1/2, 1, 2\}$.
4. Write a random walk MH sampler with $h \sim N(0, \sigma^2)$.
5. Show $R(x_t, X^*) = \exp\{x_t - x^*\} I(x^* > 0)$.
6. Generate 1000 draws from f with $\sigma \in \{.2, 1, 5\}$.
7. In general, do you prefer an independence chain or a random walk MH sampler? Why?

Metropolis Hastings Code

```
#########################################
## Introduction to MH Samplers
#########################################

## Independence Metropolis sampler with Exp(theta) proposal.

ind.chain <- function(x, n, theta = 1) {
  ## if theta = 1, then this is an iid sampler
  m <- length(x)
  x <- append(x, double(n))
  for(i in (m+1):length(x)){
    x.prime <- rexp(1, rate=theta)
    u <- exp((x[(i-1)]-x.prime)*(1-theta))
    if(runif(1) < u)
      x[i] <- x.prime
    else
      x[i] <- x[(i-1)]
  }
  return(x)
}
```

Metropolis Hastings Code

```
## Random Walk Metropolis sampler with N(0,sigma) proposal.

rw.chain <- function(x, n, sigma = 1) {
  m <- length(x)
  x <- append(x, double(n))
  for(i in (m+1):length(x)){
    x.prime <- x[(i-1)] + rnorm(1, sd = sigma)
    u <- exp((x[(i-1)]-x.prime))
    u
    if(runif(1) < u && x.prime > 0)
      x[i] <- x.prime
    else
      x[i] <- x[(i-1)]
  }
  return(x)
}
```

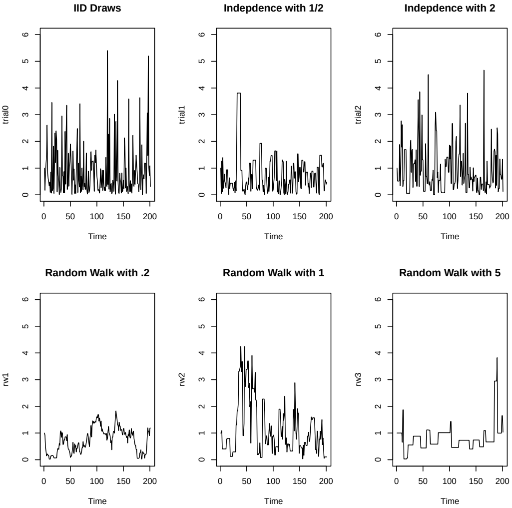
Metropolis Hastings Code

```
## Simulations

trial0 <- ind.chain(1, 200, 1)
trial1 <- ind.chain(1, 200, 2)
trial2 <- ind.chain(1, 200, 1/2)
rw1 <- rw.chain(1, 200, .2)
rw2 <- rw.chain(1, 200, 1)
rw3 <- rw.chain(1, 200, 5)

## Plotting

par(mfrow=c(2,3))
plot.ts(trial0, ylim=c(0,6), main="IID Draws")
plot.ts(trial1, ylim=c(0,6), main="Independence with 1/2")
plot.ts(trial2, ylim=c(0,6), main="Independence with 2")
plot.ts(rw1, ylim=c(0,6), main="Random Walk with .2")
plot.ts(rw2, ylim=c(0,6), main="Random Walk with 1")
plot.ts(rw3, ylim=c(0,6), main="Random Walk with 5")
par(mfrow=c(1,1))
```

Metropolis Hastings Code

```
## Writing out a plot
pdf("MHPlot.pdf")
par(mfrow=c(2,3))
plot.ts(trial0, ylim=c(0,6), main="IID Draws")
plot.ts(trial1, ylim=c(0,6), main="Indepdence with 1/2")
plot.ts(trial2, ylim=c(0,6), main="Indepdence with 2")
plot.ts(rw1, ylim=c(0,6), main="Random Walk with .2")
plot.ts(rw2, ylim=c(0,6), main="Random Walk with 1")
plot.ts(rw3, ylim=c(0,6), main="Random Walk with 5")
par(mfrow=c(1,1))
dev.off()
```

Sampler Comparison



Gibbs Sampling

Works with a univariate (or blocks) conditional distribution, which are often available in closed form. Consider the following notation

$$X = \left(X^{(1)}, \ldots, X^{(p)}\right)^T \text{ and}$$
$$X^{(-i)} = \left(X^{(1)}, \ldots, X^{(i-1)}, X^{(i+1)}, \ldots, X^{(p)}\right)^T .$$

If $f(x^{(i)}|x^{(-i)})$ is available in closed form, then the Gibbs sampler is as follows.

Gibbs Sampling

1. Select starting values x_0 and set $t = 0$.
2. Generate in turn (deterministic scan Gibbs sampler)

$$
\begin{aligned}
&x_{t+1}^{(1)} \sim f(x^{(1)} | x_t^{(-1)}) \\
&x_{t+1}^{(2)} \sim f(x^{(2)} | x_{t+1}^{(1)}, x_t^{(3)}, \ldots, x_t^{(p)}) \\
&x_{t+1}^{(3)} \sim f(x^{(3)} | x_{t+1}^{(1)}, x_{t+1}^{(2)}, x_t^{(4)}, \ldots, x_t^{(p)}) \\
&\ldots \\
&x_{t+1}^{(p)} \sim f(x^{(p)} | x_{t+1}^{(-p)}) \ .
\end{aligned}
$$

3. Increment t and go to Step 2.

Gibbs Sampling

- Common to have one or more components not available in closed form. Then one can just use a MH sampler for those components known as a Metropolis within Gibbs or Hybrid Gibbs sampling.
- Common to "block" groups of random variables.

Capture-recapture Study

Exercise: Data from a fur seal pup capture-recapture study. Goal is to estimate the number of pups in a fur seal colony using a capture-recapture study.

		1	2	3	4	5	6	7
Captured	c_i	30	22	29	26	31	32	35
Newly Caught	m_i	30	8	17	7	9	8	5

Table: Count of fur seal pup capture-recapture study for $i = 7$ census attempts.

Capture-recapture Study

Let N be the population size, I be the number of census attempts where c_i were captured ($I = 7$ in our case), and r be the total number captured ($r = \sum_{i=1}^{I} m_i = 84$).

We consider a separate unknown capture probability for each census $(\alpha_1, \ldots, \alpha_I)$ where the animals are equally "catchable". Then

$$L(N, \alpha | c, r) \propto \frac{N!}{(N-r)!} \prod_{i=1}^{I} \alpha_i^{c_i} (1-\alpha_i)^{N-c_i}.$$

Capture-recapture Study

Assume N and α are apriori independent with

$$f(N) \propto 1 \text{ and } f(\alpha_i|\theta_1, \theta_2) \overset{i.i.d.}{\sim} \text{Beta}(\theta_1, \theta_2).$$

We use $\theta_1 = \theta_2 = 1/2$, which is the Jeffrey's Prior. The resulting posterior is proper when $I > 2$ and recommended when $I > 5$.

Capture-recapture Study

Then it is easy to show the posterior is

$$f(N, \alpha | c, r) \propto \frac{N!}{(N-r)!} \prod_{i=1}^{I} \alpha_i^{c_i} (1-\alpha_i)^{N-c_i} \prod_{i=1}^{I} \alpha_i^{-1/2} (1-\alpha_i)^{-1/2}.$$

Further, one can show

$$N - 84 | \alpha \sim \text{NB}\left(85, 1 - \prod_{i=1}^{I} (1-\alpha_i)\right) \text{ and}$$

$$\alpha_i | N \sim \text{Beta}\left(c_i + 1/2, N - c_i + 1/2\right) \text{ for all } i.$$

Capture-recapture Study

Then we can consider the chain

$$(N, \alpha) \to (N', \alpha) \to (N', \alpha')$$

or

$$(N, \alpha) \to (N, \alpha') \to (N', \alpha')\,,$$

where both involve a "block" update of α.

The following R code implements the Gibbs sampler above along with some measures of uncertainty for the resulting estimators.

Capture-recapture Code

```
set.seed(1)

########################################
## Capture-recapture Data
########################################

captured <- c(30, 22, 29, 26, 31, 32, 35)
new.captures <- c(30, 8, 17, 7, 9, 8, 5)
total.r <- sum(new.captures)
```

Capture-recapture Code

```
########################################
## Gibbs Sampler
########################################

gibbs.chain <- function(n, N.start = 94, alpha.start = rep(.5,7)) {
output <- matrix(0, nrow=n, ncol=8)
for(i in 1:n){
neg.binom.prob <- 1 - prod(1-alpha.start)
N.new <- rnbinom(1, 85, neg.binom.prob) + total.r
beta1 <- captured + .5
beta2 <- N.new - captured + .5
alpha.new <- rbeta(7, beta1, beta2)
output[i,] <- c(N.new, alpha.new)
N.start <- N.new
alpha.start <- alpha.new
}
return(output)
}
```

Capture-recapture Code

```
########################################
## Preliminary Simulations
########################################

trial <- gibbs.chain(1000)
plot.ts(trial[,1], main = "Trace plot for N")
for(i in 1:7){
plot.ts(trial[,(i+1)], main = paste("Trace plot for Alpha", i))
readline("Press <return to continue")
}

acf(trial[,1], main = "Lag Correlation plot for N")
for(i in 1:7){
acf(trial[,(i+1)], main = paste("Lag Correlation plot for Alpha", i))
readline("Press <return to continue")
}
```

Capture-recapture Code

```
##########################################
## Simulations
##########################################

sim <- gibbs.chain(10000)
N <- sim[,1]
alpha1 <- sim[,2]
hist(N, freq=F, main="Estimated Marginal Posterior for N")
hist(alpha1, freq=F, main ="Estimating Marginal Posterior for Alpha 1")

library(mcmcse)

ess(N)
ess(alpha1)

estvssamp(N)
estvssamp(alpha1)
```

Capture-recapture Code

```
mcse(N)
mcse.q(N, .05)
mcse.q(N, .95)

mcse(alpha1)
mcse.q(alpha1, .05)
mcse.q(alpha1, .95)
```

Capture-recapture Code

```
current <- sim[10000,] # start from here is you need more simulations
sim <- rbind(sim, gibbs.chain(10000, N.start = current[1],
       alpha.start = current[2:8]))
N.big <- sim[,1]
hist(N.big, freq=F, main="Estimated Marginal Posterior for N")

ess(N)
ess(N.big)

estvssamp(N)
estvssamp(N.big)

mcse(N)
mcse(N.big)

mcse.q(N, .05)
mcse.q(N.big, .05)
mcse.q(N, .95)
mcse.q(N.big, .95)
```

MCMC Output Analysis

- Let π be a probability distribution having support $\mathcal{X} \subseteq \mathbb{R}^d$, $d \geq 1$ we want to explore.
- When i.i.d. observations are unavailable, a Markov chain with stationary distribution π can be utilized.
- Summarize π with expectations, quantiles, density plots ...

Target Features

- Consider estimating an expectation with respect to π denoted

$$\theta = \mathbb{E}_\pi g = \int_{\mathcal{X}} g(x)\pi(dx),$$

  where $g : \mathcal{X} \to \mathbb{R}$.
- However, this expectation is often intractable.
- θ is an unknown quantity I would like to estimate using simulated data.
- Let $X = \{X^{(0)}, X^{(1)}, \dots\}$ be a Markov chain.
- Usually, $X^{(j)} \sim F_j \neq \pi$ and $\mathrm{Cov}(g(X^{(j)}), g(X^{(j+1)})) > 0$.

Monte Carlo Error

- We can often find a consistent estimator of θ, say

$$\theta_n = \bar{g}(n) := \frac{1}{n} \sum_{j=0}^{n-1} g\left(X^{(j)}\right).$$

- Want $\theta_n - \theta$, the Monte Carlo error, to be small.
- Under regularity conditions, a Markov chain CLT holds,

$$\sqrt{n}\,(\theta_n - \theta) \xrightarrow{d} \mathrm{N}\left(0, \sigma^2\right) \text{ where}$$

$$\sigma^2 = \mathrm{Var}_\pi\left[g\right] + 2\sum_{k=1}^{\infty} \mathrm{Cov}_\pi\left[g(X^{(0)}), g(X^{(0+k)})\right].$$

Monte Carlo Error

- Let $\hat{\sigma}(n)$ denote an estimator of σ. Then the CLT allows construction of a $100(1-\delta)\%$ confidence interval with width

$$w_n = 2z_{\delta/2}\frac{\hat{\sigma}(n)}{\sqrt{n}}.$$

- Suppose $\epsilon > 0$, then a **fixed-width stopping rule** terminates the simulation the first time $w_n < \epsilon$.

AR(1) Model

Consider the Markov chain such that

$$X_i = \rho X_{i-1} + \epsilon_i$$

where $\epsilon_i \stackrel{iid}{\sim} N(0, 1)$.

- Consider $X_1 = 0$, $\rho = .95$, and estimating $E_\pi X = 0$.
- Run until
  $$w_n = 2z_{.975} \frac{\hat{\sigma}(n)}{\sqrt{n}} \leq 0.2$$
  where $\hat{\sigma}(n)$ is calculated using batch means.

AR(1) Code

```
# The following will provide an observation from the MC 1 step ahead

ar1 <- function(m, rho, tau) {
rho*m + rnorm(1, 0, tau)
}

# Next, we will add to this program so that we can give it a Markov
# chain and the result will be p observations from the Markov chain.

ar1.gen <- function(mc, p, rho, tau, q=1) {
loc <- length(mc)
junk <- double(p)
mc <- append(mc, junk)

for(i in 1:p){
j <- i+loc-1
mc[(j+1)] <- ar1(mc[j], rho, tau)
}
return(mc)
}
```

AR(1) Code

```
set.seed(20)
library(mcmcse)

tau <- 1
rho <- .95
out <- 0
eps <- 0.1
start <- 1000
r <- 1000
```

AR(1) Code

```
out <- ar1.gen(out, start, rho, tau)
MCSE <- mcse(out)$se
N <- length(out)
t <- qt(.975, (floor(sqrt(N) - 1)))
muhat <- mean(out)
check <- MCSE * t

while(eps < check) {
out <- ar1.gen(out, r, rho, tau)
MCSE <- append(MCSE, mcse(out)$se)
N <- length(out)
t <- qt(.975, (floor(sqrt(N) - 1)))
muhat <- append(muhat, mean(out))
check <- MCSE[length(MCSE)] * t
}
```

AR(1) Code

```
N <- seq(start, length(out), r)
t <- qt(.975, (floor(sqrt(N) - 1)))
half <- MCSE * t
sigmahat <- MCSE*sqrt(N)
N <- seq(start, length(out), r) / 1000

plot(N, muhat, main="Estimates of the Mean",
         xlab="Iterations (in 1000's)")
points(N, muhat, type="l", col="red")
abline(h=0, lwd=3)
legend("bottomright", legend=c("Observed", "Actual"),
        lty=c(1,1), col=c(2,1), lwd=c(1,3))
```

AR(1) Code

```
plot(N, sigmahat, main="Estimates of Sigma", xlab="Iterations (in 1000's)")
points(N, sigmahat, type="l", col="red")
abline(h=20, lwd=3)
legend("bottomright", legend=c("Observed", "Actual"), lty=c(1,1),
        col=c(2,1), lwd=c(1,3))

plot(N, 2*half, main="Calculated Interval Widths", xlab="Iterations
       (in 1000's)", ylab="Width", ylim=c(0, 1.8))
points(N, 2*half, type="l", col="red")
abline(h=0.2, lwd=3)
legend("topright", legend=c("Observed", "Cut-off"), lty=c(1,1), col=c(2,1),
       lwd=c(1,3))
```

AR(1) Model

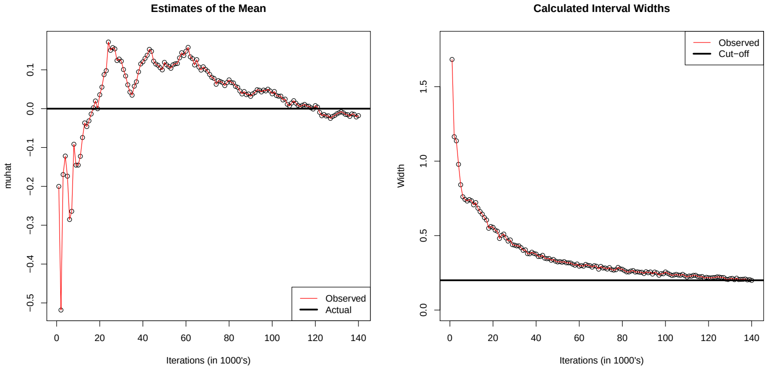


Figure: Results from one simulation using a cut-off of $\epsilon = 0.2$.

Asymptotically Valid Confidence Intervals

What requirements are necessary for asymptotically valid confidence intervals?

1. Need a Markov chain CLT to hold.
2. Need $\hat{\sigma}_g^2$ to be a strongly consistent estimator of σ_g^2.

Does this work in practice with finite samples? How does it compare to other methods?

Asymptotically Valid Confidence Intervals

Need $\hat{\sigma}_g^2$ to be a strongly consistent estimator of σ_g^2.

- Batch Means
- Overlapping Batch Means (Subsampling)
- Spectral Variance Estimators
- Regeneration

Batch Means

- Batch Means produces a strongly consistent estimator of σ_g^2.
- Let b_n be the batch size, $a_n = n/b_n$ be the number of batches, and define a batch mean as

$$\bar{Y}_k := \frac{1}{b_n} \sum_{i=1}^{b_n} g(X_{kb_n+i}) \qquad \text{for } k = 0, \ldots, a_n - 1 \; .$$

  Then

$$\hat{\sigma}^2_{BM} = \frac{b_n}{a_n - 1} \sum_{k=0}^{a_n-1} (\bar{Y}_k - \bar{g}_n)^2 \; .$$

- Requires $b_n \to \infty$ and $a_n \to \infty$ as $n \to \infty$.

Gelman and Rubin Diagnostic

Gelman and Rubin Diagnostic — another stopping criteria.

- Most popular method for stopping the simulation, one of many *convergence diagnostics*.
- Simulates m independent parallel Markov chains.
- Considers a ratio of two different estimates of $\text{Var}_\pi g$, not σ_g^2 from the CLT.
- Argue the simulation should continue until the diagnostic ($\widehat{R}_{0.975}$) is close to 1.

Toy Example

- Let $Y_1, \ldots, Y_m$ be i.i.d. $\mathrm{N}(\mu, \lambda)$ and let the prior for (μ, λ) be proportional to $1/\sqrt{\lambda}$. The posterior density is characterized by

$$\pi(\mu, \lambda | y) \propto \lambda^{-\frac{m+1}{2}} \exp\left\{ -\frac{1}{2\lambda} \sum_{j=1}^{m} (y_j - \mu)^2 \right\}$$

  which is proper as long as $m \geq 3$.

- A Gibbs sampler requires the full conditionals:

$$\begin{aligned} \mu | \lambda, y &\sim \mathrm{N}(\bar{y}, \lambda/m) \ , \\ \lambda | \mu, y &\sim \mathrm{IG}\left(\frac{m-1}{2}, \frac{s^2 + m(\bar{y} - \mu)^2}{2} \right) \ , \end{aligned}$$

  where $\bar{y}$ is the sample mean and $s^2 = \sum (y_i - \bar{y})^2$.

Toy Example

$$\boxed{\pi(\mu, \lambda | y) \propto \lambda^{-\frac{m+1}{2}} \exp\left\{-\frac{1}{2\lambda} \sum (y_j - \mu)^2\right\}}$$

Consider the Gibbs sampler that updates λ then μ.

$$(\lambda', \mu') \to (\lambda, \mu') \to (\lambda, \mu)$$

Jones and Hobert showed this sampler is geometrically ergodic.

1. Suppose $m = 11$, $\bar{y} = 1$, and $s^2 = 14$.
   - Then $E(\mu|y) = 1$ and $E(\lambda|y) = 2$.
2. Consider estimating $E(\mu|y)$ and $E(\lambda|y)$ with $\bar{\mu}_n$ and $\bar{\lambda}_n$.
   - CLT holds!
   - Using $b = \lfloor n^{1/2} \rfloor$, BM Theorem holds!

Simulation Settings

Stopped the simulation when

$$BM: \quad t_{.975,(a-1)} \frac{\hat{\sigma}_{BM}}{\sqrt{n}} + I(n < 400) < 0.04$$

$$GRD: \qquad \hat{R}_{0.975} + I(n < 400) < 1.005$$

1. 1000 independent replications
   - Starting from $\bar{y}$ for BM.
   - Starting from draws from π for GRD.
2. Used 4 chains for GRD.

Simulation Results

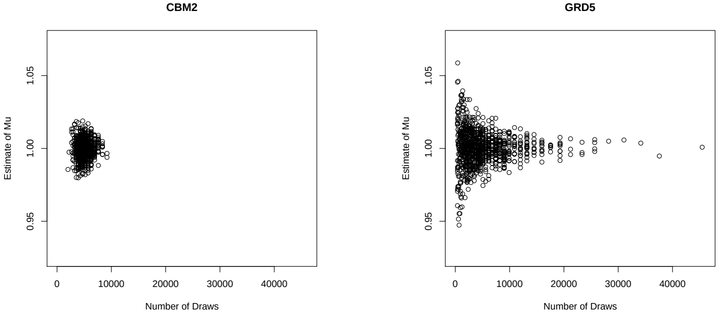


Figure: Plots of $\bar{\mu}_n$ vs. n for both stopping methods.

Simulation Results

	BM	GRD
MSE for $E(\mu \vert y)$	3.73e-05 (1.8e-06)	0.000134 (9.2e-06)
MSE for $E(\lambda \vert y)$	0.000393 (1.8e-05)	0.00165 (0.00012)

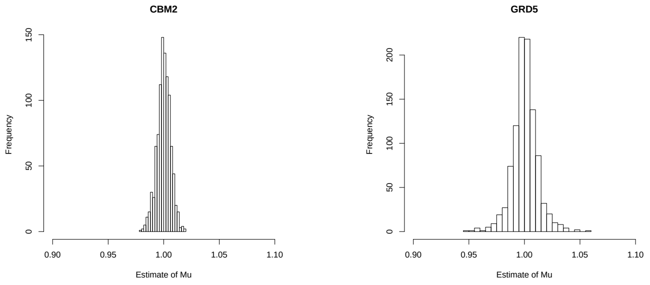


Figure: Histograms of $\bar{\mu}_n$ for both stopping methods.

Summary

- Monte Carlo and MCMC Simulations
  - Include uncertainty estimates, e.g. a MCSE
  - Useful for interpretation (`mcmcse` R package)
- Finding a good MCMC sampler is critical
  - `mcmc` R package is a good starting point, but there are others
  - Other software available; `OpenBUGS`, `Stan`, `JAGS`, packages within `Python` ...

Other Topics in MCMC

- Convergence diagnostics, ESS, trace plots, ACF plots, ...
- Estimating quantiles, or endpoints of credible regions
- Fixed-width stopping rules
    - Relative standard deviation fixed-width stopping rule equivalent to stopping when ESS is large enough
- Multivariate estimation and output analysis
- Slice sampling, reversible-jump Metropolis, adaptive random walk samplers, sequential Monte Carlo (particle filters), simulated annealing algorithms, ...