Monte Carlo Statistical Methods

George Casella
Department of Statistics
University of Florida
casella@ufl.edu

Based on

- Monte Carlo Statistical Methods,
  Christian Robert and George Casella,
  2004, Springer-Verlag
- Programming in R (available as a free download from
  http://www.r-project.org
- Also WinBugs, available free from
  http://www.mrc-bsu.cam.ac.uk/bugs/
- R programs for the course available at
  http://www.stat.ufl.edu/∼casella/mcsm/

Introduction

- Statistical Models
- Likelihood Models
- Bayesian Models
- Deterministic Numerical Models
- Simulation vs. Numerical Methods

1.1 Statistical Models

- In a typical statistical model we observe

$$Y_1, Y_2, \ldots, Y_n \sim f(y|\theta)$$

- The distribution of the sample is given by the product, the likelihood function

$$\prod_{i=1}^{n} f(y_i|\theta).$$

- Inference about θ is based on this likelihood.
- In many situations the likelihood can be complicated

Example 1.1: Censored Random Variables

- If
$$X_1 \sim N(\theta, \sigma^2), \quad X_2 \sim N(\mu, \rho^2),$$
- the distribution of $Y = \min\{X_1, X_2\}$ is
$$\left[1 - \Phi\left(\frac{y-\theta}{\sigma}\right)\right] \times \rho^{-1}\phi\left(\frac{y-\mu}{\rho}\right) + \left[1 - \Phi\left(\frac{y-\mu}{\rho}\right)\right] \times \sigma^{-1}\phi\left(\frac{y-\theta}{\sigma}\right),$$
where Φ and ϕ are the cdf and pdf of the normal distribution.
- This results in a complex likelihood.

Example 1.2: Mixture Models

- Models of *mixtures of distributions*:

$$X \sim f_j \text{ with probability } p_j,$$

for $j = 1, 2, \ldots, k$, with overall density

$$X \sim p_1 f_1(x) + \cdots + p_k f_k(x) \ .$$

For a sample of independent rvs $(X_1, \cdots, X_n)$, sample density

$$\prod_{i=1}^{n} \{p_1 f_1(x_i) + \cdots + p_k f_k(x_i)\} \ .$$

- Expanding this product involves k^n elementary terms: prohibitive to compute in large samples.

Example 1.2 : Normal Mixtures

- For a mixture of two normal distributions,

$$p\mathcal{N}(\mu, \tau^2) + (1-p)\mathcal{N}(\theta, \sigma^2) \ ,$$

- The likelihood proportional to

$$\prod_{i=1}^{n} \left[p\tau^{-1}\varphi\left(\frac{x_i - \mu}{\tau}\right) + (1-p)\,\sigma^{-1}\,\varphi\left(\frac{x_i - \theta}{\sigma}\right) \right]$$

  containing 2^n terms.

- Standard maximization techniques often fail to find the global maximum because of multimodality of the likelihood function.

- **R program → normal-mixture1**

1.2: Likelihood Methods

- Maximum Likelihood Methods
  - For an iid sample $X_1, \ldots, X_n$ from a population with density $f(x|\theta_1, \ldots, \theta_k$ the *likelihood function* is

$$\begin{aligned} L(\boldsymbol{\theta}|\mathbf{x}) &= L(\theta_1, \ldots, \theta_k|x_1, \ldots, x_n) \\ &= \prod_{i=1}^{n} f(x_i|\theta_1, \ldots, \theta_k). \end{aligned}$$

  - Global justifications from asymptotics

Example 1.9: Student's t distribution

- Reasonable alternative to normal errors is Student's t distribution, denoted by

  $$\mathcal{T}(p, \theta, \sigma)$$

  more "robust" against possible modelling errors
- Density of $\mathcal{T}(p, \theta, \sigma)$ proportional to

  $$\sigma^{-1} \left(1 + \frac{(x-\theta)^2}{p\sigma^2}\right)^{-(p+1)/2} ,$$

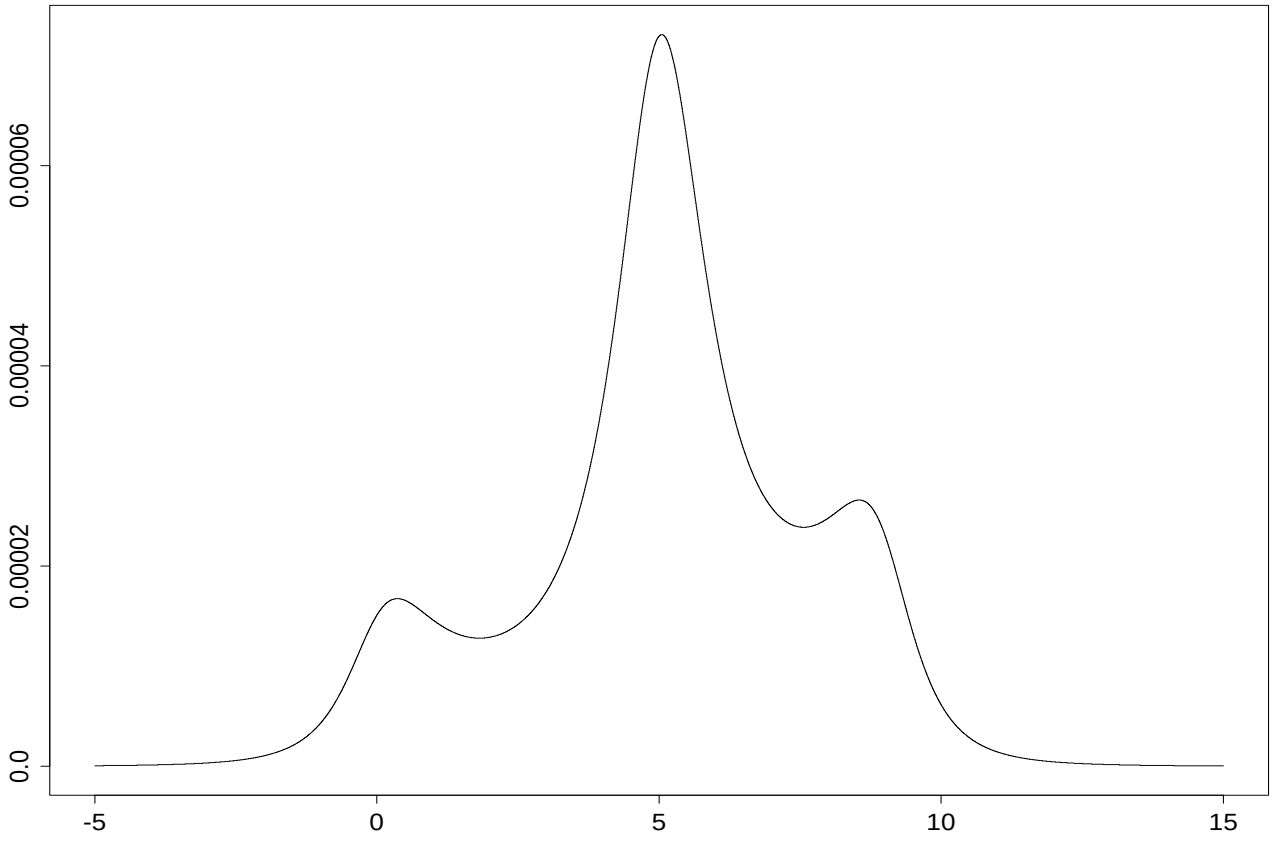
Example 1.9: Student's t distribution

- When p known and θ and σ both unknown, the likelihood

$$\sigma^{n\frac{p+1}{2}} \prod_{i=1}^{n} \left(1 + \frac{(x_i - \theta)^2}{p\sigma^2}\right) .$$

  may have n local minima.

- Each of which needs to be calculated to determine the global maximum.



- Illustration of the multiplicity of modes of the likelihood from a Cauchy distribution $\mathcal{C}(\theta, 1)$ $(p = 1)$ when $n = 3$ and $X_1 = 0$, $X_2 = 5$, and $X_3 = 9$.

Section 1.3 Bayesian Methods

- In the Bayesian paradigm, information brought by
  - the data x, realization of
$$X \sim f(x|\theta),$$
  - combined with prior information specified by *prior distribution* with density $\pi(\theta)$

Bayesian Methods

- Summary in a probability distribution, $\pi(\theta|x)$, called the **posterior distribution**
- Derived from the *joint* distribution $f(x|\theta)\pi(\theta)$, according to

$$\pi(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\int f(x|\theta)\pi(\theta)d\theta},$$

[Bayes Theorem]

- where

$$m(x) = \int f(x|\theta)\pi(\theta)d\theta$$

is the *marginal density* of X

Example 1.11: Binomial Bayes Estimator

- For an observation X
  from the binomial distribution Binomial(n, p) the (so-called) conjugate prior
  is the family of beta distributions Beta(a, b)
- The classical Bayes estimator δ^π is the posterior mean

$$\begin{aligned}\delta^\pi &= \frac{\Gamma(a+b+n)}{\Gamma(a+x)\Gamma(n-x+b)} \int_0^1 p\, p^{x+a-1}(1-p)^{n-x+b-1} dp \\ &= \frac{n}{a+b+n}\left(\frac{x}{n}\right) + \frac{a+b}{a+b+n}\left(\frac{a}{a+b}\right).\end{aligned}$$

- A Biased estimator of p

The Variance/Bias Trade-off

- Bayes Estimators are biased
- Mean Squared Error (MSE) = Variance + Bias^2
  - MSE = $\mathrm{E}(\delta^\pi - p)^2$
  - Measures average closeness to parameter
- Small Bias $\uparrow$ can yield large Variance $\downarrow$.

$$\delta^\pi = \frac{n}{a+b+n}\left(\frac{x}{n}\right) + \frac{a+b}{a+b+n}\left(\frac{a}{a+b}\right)$$

$$\mathrm{Var}\delta^\pi = \left(\frac{n}{a+b+n}\right)^2 \mathrm{Var}\left(\frac{x}{n}\right)$$

Conjugate Priors

- A prior is conjugate if

$$\pi(\theta)\text{(the prior) and } \pi(\theta|x)\text{(the posterior)}$$

  are in the same family of distributions.

- Examples
  - $\pi(\theta)$ normal , $\pi(\theta|x)$ normal
  - $\pi(\theta)$ beta , $\pi(\theta|x)$ beta
- Restricts the choice of prior
- Typically non-robust
- Originally used for computational ease

Example 1.13: Logistic Regression

- Standard regression model for binary $(0-1)$ responses: the *logit model* where distribution of Y modelled by

$$P(Y = 1) = p = \frac{\exp(x^t\beta)}{1 + \exp(x^t\beta)}.$$

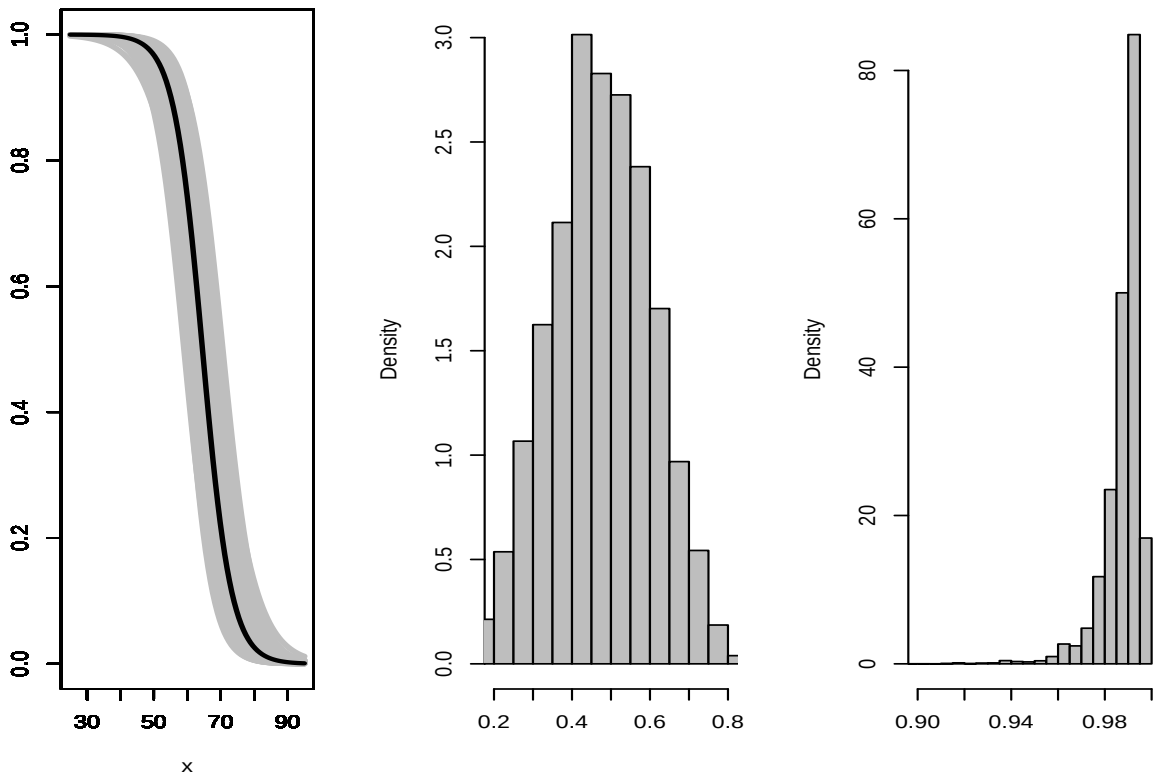
- Equivalently, the *logit* transform of p, $\text{logit}(p) = \log[p/(1-p)]$, satisfies $\text{logit}(p) = x^t\beta$.
- Computation of a confidence region on β quite delicate when $\pi(\beta|x)$ not explicit.
- In particular, when the confidence region involves only one component of a vector parameter, calculation of $\pi(\beta|x)$ requires the integration of the joint distribution over all the other parameters.

Challenger Data

- In 1986, the space shuttle Challenger exploded during take off, killing the seven astronauts aboard.
- The explosion was the result of an *O-ring* failure.

Flight No.	14	9	23	10	1	5	13	15	4	3	8	17
Failure	1	1	1	1	0	0	0	0	0	0	0	0
Temp.	53	57	58	63	66	67	67	67	68	69	70	70
Flight No.	2	11	6	7	16	21	19	22	12	20	18	
Failure	1	1	0	0	0	1	0	0	0	0	0	
Temp.	70	70	72	73	75	75	76	76	78	79	81	

- It is reasonable to fit a logistic regression, with p = probability of an O-ring failure and x = temperature.



- The left panel shows the average logistic function and variation
- The middle panel shows predictions of failure probabilities at 65^o Fahrenheit
- The right panel shows predictions of failure probabilities at 45^o Fahrenheit.

Section 1.4: Deterministic Numerical Methods

- To solve an equation of the form

$$f(x) = 0,$$

the *Newton–Raphson* algorithm produces a sequence x_n:

$$x_{n+1} = x_n - \left(\frac{\partial f}{\partial x}\bigg|_{x=x_n} \right)^{-1} f(x_n)$$

that converges to a solution of $f(x) = 0$.

- Note that $\frac{\partial f}{\partial x}$ is a matrix in multidimensional settings.

Example 1.17: Newton-Raphson

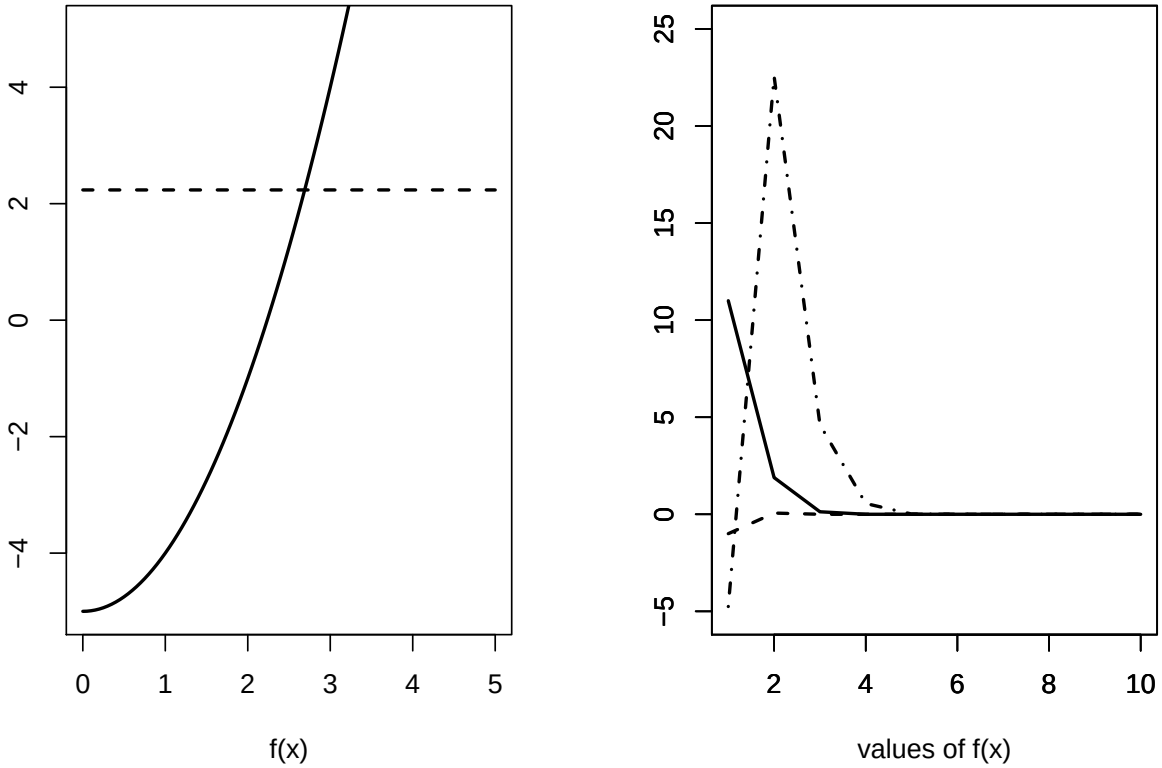
- Newton-Raphson algorithm can be used to find the square root of a number.
- If we are interested in the square root of b, this is equivalent to solving the equation

$$f(x) = x^2 - b = 0.$$

- This results in the iterations

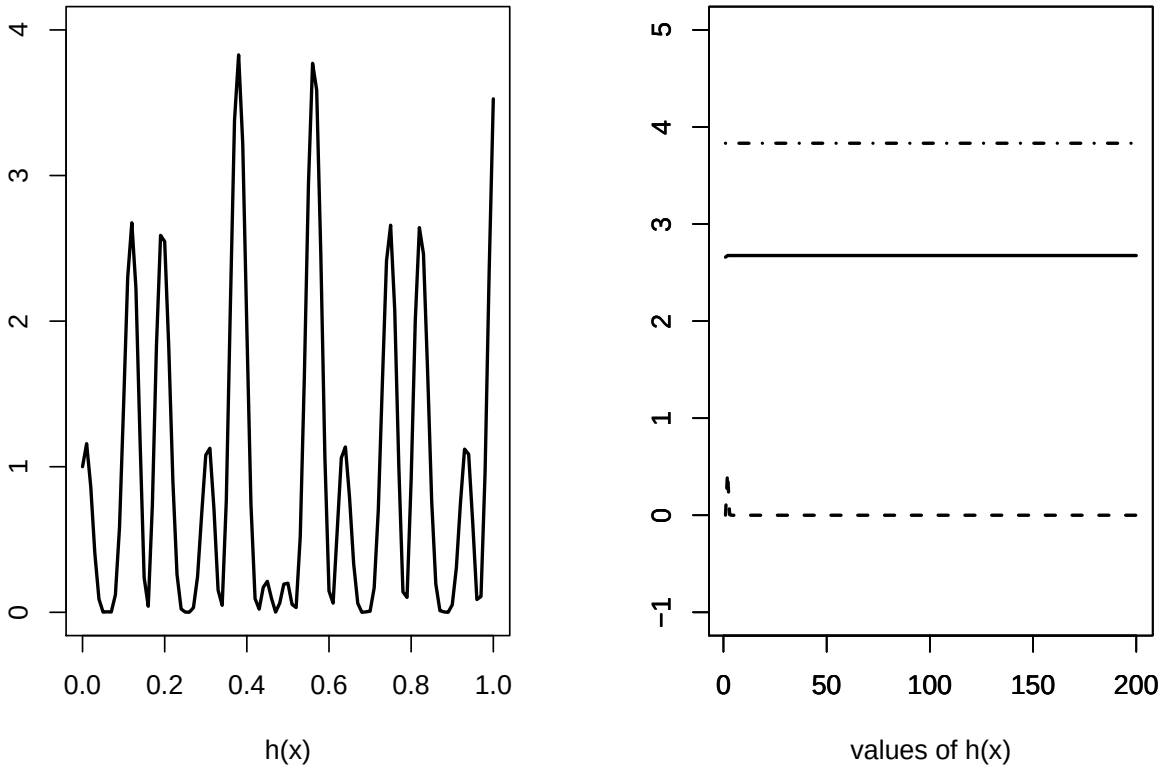
$$x^{(j+1)} = x^{(j)} - \frac{f(x^{(j)})}{f'(x^{(j)})} = x^{(j)} - \frac{x^{(j)2} - b}{2x^{(j)}} = \frac{1}{2}(x^{(j)} + \frac{b}{x^{(j)}}).$$

Example 1.17: Newton-Raphson -2



- Left: x^2; Right: $f(x) = x^2 - 2$
- Rapid convergence from different starting points
- Three runs are shown, starting at $x = .5, 2.4$.

Example 1.17: Newton-Raphson -3



- Problems with the function $h(x) = [\cos(50x) + \sin(20x)]^2$.
- "Greediness" of the Newton-Raphson algorithm pushes it to the nearest mode.

Variants of Newton-Raphson

- The *steepest descent* method, where each iteration results in a unidimensional optimizing problem for $F(x_n + td_n)$ $(t \in \Re)$, d_n being an acceptable direction, namely such that

$$\left.\frac{d^2F}{dt^2}(x_n + td_n)\right|_{t=0}$$

  is of the proper sign.

- The direction d_n is often chosen as ∇F or as

$$\left[\nabla\nabla^t F(x_n) + \lambda I\right]^{-1} \nabla F(x_n),$$

  in the *Levenberg–Marquardt version*.

Section 1.4.2: Integration

- The numerical computation of an integral

$$I = \int_a^b h(x)dx$$

  can be done by simple *Riemann integration*.

- By improved techniques such as the *trapezoidal rule*

$$\hat{I} = \frac{1}{2}\sum_{i=1}^{n-1}(x_{i+1} - x_i)(h(x_i) + h(x_{i+1}))\ ,$$

  where the x_i's constitute an ordered partition of $[a, b]$.

Section 1.4.2: Integration

- By *Simpson's rule*, whose formula is

$$\tilde{I} = \frac{\delta}{3}\left\{ f(a) + 4\sum_{i=1}^{n} h(x_{2i-1}) + 2\sum_{i=1}^{n} h(x_{2i}) + f(b) \right\}$$

  in the case of equally spaced samples with $(x_{i+1} - x_i) = \delta$.
  - Other approaches involve orthogonal polynomials (Gram–Charlier, Legendre, etc.)
  - Splines
- However, these methods may not work well in high dimensions

Comparison

- Advantages of Simulation
  - Integration may focus on areas of low probability
  - Simulation can avoid these
  - Local modes are a problem for deterministic methods
- Advantages of Deterministic Methods
  - Simulation does not consider the form of the function
  - Deterministic Methods can be much faster for smooth functions.
  - In low dimensions Riemann Sums or Quadrature are very fast

Comparison

- When the statistician
  - needs to study the details of a likelihood surface or posterior distribution
  - needs to simultaneously estimate several features of these functions
  - when the distributions are highly multimodal
- it is preferable to use a simulation-based approach.
- fruitless to advocate the superiority of one method over the other
- More reasonable to justify the use of simulation-based methods by the statistician in terms of *expertise*.
- The intuition acquired by a statistician in his or her every-day processing of random models can be directly exploited in the implementation of simulation techniques

Chapter 2: RV Generation

- Rely on the possibility of producing (with a computer) a supposedly endless flow of random variables (usually iid) for well-known distributions.
- Although we are not directly concerned with the *mechanics* of producing uniform rvs, we are concerned with the *statistics* of producing uniform and other rvs.
- We look at some basic methodology that can, starting from these simulated uniform rvs, produce rvs from both standard and nonstandard distributions.

Uniform Random Numbers

- *A uniform pseudo-random number generator* is an algorithm which, starting from an initial value u_0 and a transformation D, produces a sequence $(u_i) = (D^i(u_0))$ of values in $[0, 1]$.
- or all n, the values $(u_1, \ldots, u_n)$ reproduce the behavior of an iid sample $(V_1, \ldots, V_n)$ of uniform rvs when compared through a usual set of tests.

Uniform Random Numbers

- This definition is clearly restricted to *testable* aspects of the random variable generation, which are connected through the deterministic transformation $u_i = D(u_{i-1})$.
- The validity of the algorithm consists in the verification that the sequence $U_1, \ldots, U_n$ leads to acceptance of the hypothesis

$$\mathrm{H}_0 : U_1, \ldots, U_n \quad \text{are iid} \quad \mathcal{U}_{[0,1]}.$$

- The set of tests used is generally of some consequence.
  - Kolmogorov–Smirnov
  - Nonparametric
  - Time Series
  - Die Hard (Marsaglia)
- Our definition is *functional*: An algorithm that generates uniform numbers is acceptable if it is not rejected by a set of tests.

KISS Algorithm

- A preferred algorithm
  - A congruential generator $D(x) = ax + b(modM + 1)$
  - Register Shifts to break patterns
- Period of order 2^{95}
- Successfully tested on Die Hard

The Inverse Transform

- Lemma 2.4: If X has the cdf $F(x)$, then the rv $F(X)$ has the $\mathcal{U}_{[0,1]}$ distribution.

- Thus, formally, in order to generate a rv $X \sim F$, it suffices to generate U according to $\mathcal{U}_{[0,1]}$ and then make the transformation $x = F^{-}(u)$.

- In other words, simulate $U \sim \mathcal{U}_{[0,1]}$ then solve for X in

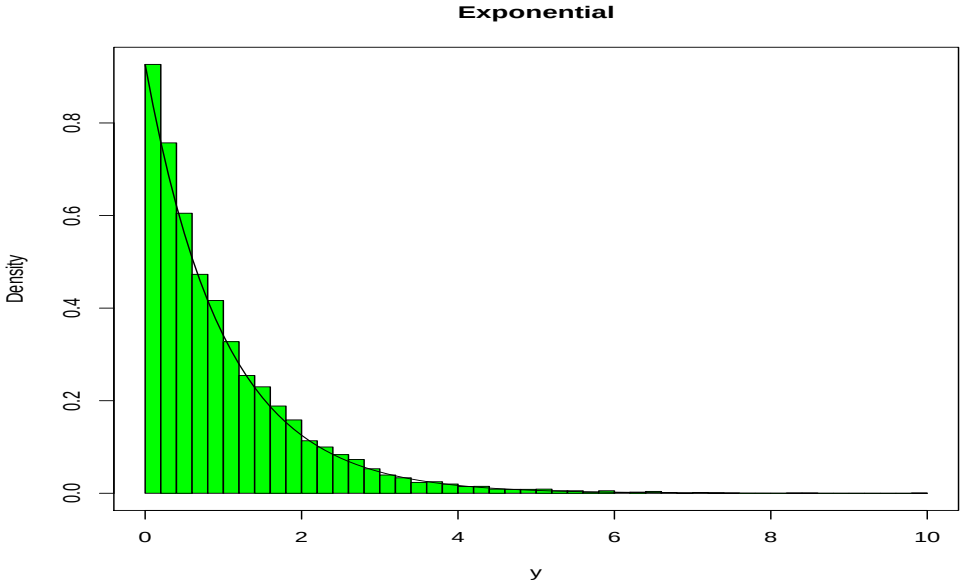
$$U = F(X) = \int_{-\infty}^{X} f(x)dx$$

Example 2.5: Exponential variable generation

- If $X \sim \mathcal{E}xp(1)$, so $F(x) = 1 - e^{-x}$, then solving for x in $u = 1 - e^{-x}$ gives $x = -\log(1 - u)$.
- Therefore, if $U \sim \mathcal{U}_{[0,1]}$, the random variable $X = -\log U$ has the exponential distribution
- R program

Exponentials from Uniforms

```
#This generates exponentials from uniforms#
nsim<-10000;u<-runif(nsim);
y<--log(u);
hist(y,main="Exponential",freq=F,col="green",breaks=50)
par(new=T)
plot(function(x)dexp(x), 0,10,xlab="",ylab="",xaxt="n",yaxt="n")
```



Example 2.7; Building on exponential random variables

- Some of the random variables that can be generated starting from an exponential distribution.
- If the X_i's are iid $\mathcal{E}xp(1)$ random variables,

$$Y = 2\sum_{j=1}^{\nu} X_j \sim \chi^2_{2\nu}\,, \qquad \nu \in \{1, 2, \ldots$$

$$Y = \beta\sum_{j=1}^{a} X_j \sim \mathcal{G}a(a, \beta)\,, \qquad a \in \{1, 2, \ldots$$

$$Y = \frac{\sum_{j=1}^{a} X_j}{\sum_{j=1}^{a+b} X_j} \sim \mathcal{B}e(a, b)\,, \qquad a, b \in \{1, 2, \ldots$$

Limitations

- These transformations are quite simple to use and, hence, will often be a favorite
- There are limits to their usefulness
  - In scope of variables that can be generated
  - Efficiency of generation
  - There are more efficient algorithms for gamma and beta random variables.
- We cannot use exponentials to generate gamma random variables with a non-integer shape parameter
- We cannot get a χ^2_1 variable, which would, in turn, get us a $\mathcal{N}(0,1)$ variable.

Example 2.8: Box-Muller

- If r and θ are the polar coordinates of (X_1, X_2), then,

$$\begin{aligned} r^2 &= X_1^2 + X_2^2 \sim \chi_2^2 = \mathcal{E}xp(1/2)\,, \\ \theta &\sim \mathcal{U}_{[0,2\pi]}\,. \end{aligned}$$

- If U_1 and U_2 are iid $\mathcal{U}_{[0,1]}$, the variables X_1 and X_2 defined by

$$X_1 = \sqrt{-2\log(U_1)}\;\cos(2\pi U_2)\;, \qquad X_2 = \sqrt{-2\log(U_1)}\;\sin(2\pi U_2)\;,$$

are then iid $\mathcal{N}(0,1)$.

Box-Muller Algorithm

1. Generate U_1, U_2 iid $\mathcal{U}_{[0,1]}$;
2. Define
$$\begin{cases} x_1 = \sqrt{-2\log(u_1)}\cos(2\pi u_2) \ , \\ x_2 = \sqrt{-2\log(u_1)}\sin(2\pi u_2) \ ; \end{cases}$$
3. Take x_1 and x_2 as two independent draws from $\mathcal{N}(0,1)$.

Note on Box-Muller

- In comparison with algorithms based on the Central Limit Theorem, this algorithm is $exact$
- It produces two normal random variables from two uniform random variables
- The only drawback (in speed) being the necessity of calculating functions such as $\log$, $\cos$, and $\sin$.
- Devroye(1985) gives faster alternatives that avoid the use of these functions

Poisson Random Variables

- Discrete Random Variables can always be generated using the Probability Integral Transform.
- For Example, to generate $X \sim \text{Poisson}(\theta)$ calculate

$$p_0 = P_\theta(X \leq 0), \quad p_1 = P_\theta(X \leq 1), \quad p_2 = P_\theta(X \leq 2), \quad \ldots$$

- Then generate $U \sim \text{Uniform}[0, 1]$ and take

$$X = k \text{ if } p_{k-1} < U < p_k.$$

- There are more efficient algorithms, but this is OK

- `R` Program "DiscreteX"

Discrete Random Variables

```
p<-c(.1,.2,.3,.3,.1)    #P(X=0), P(X=1), etc
sum(p)           #check
cp<-c(0,cumsum(p))
nsim<-5000
X<-array(0,c(nsim,1))

for(i in 1:nsim)
{
    u<-runif(1)
    X[i]<-sum(cp<u)-1
}
hist(X)
```

- See also "Logarithmic"

Negative Binomial Random Variables

- A Poisson generator can be used to get Negative Binomial random variables since

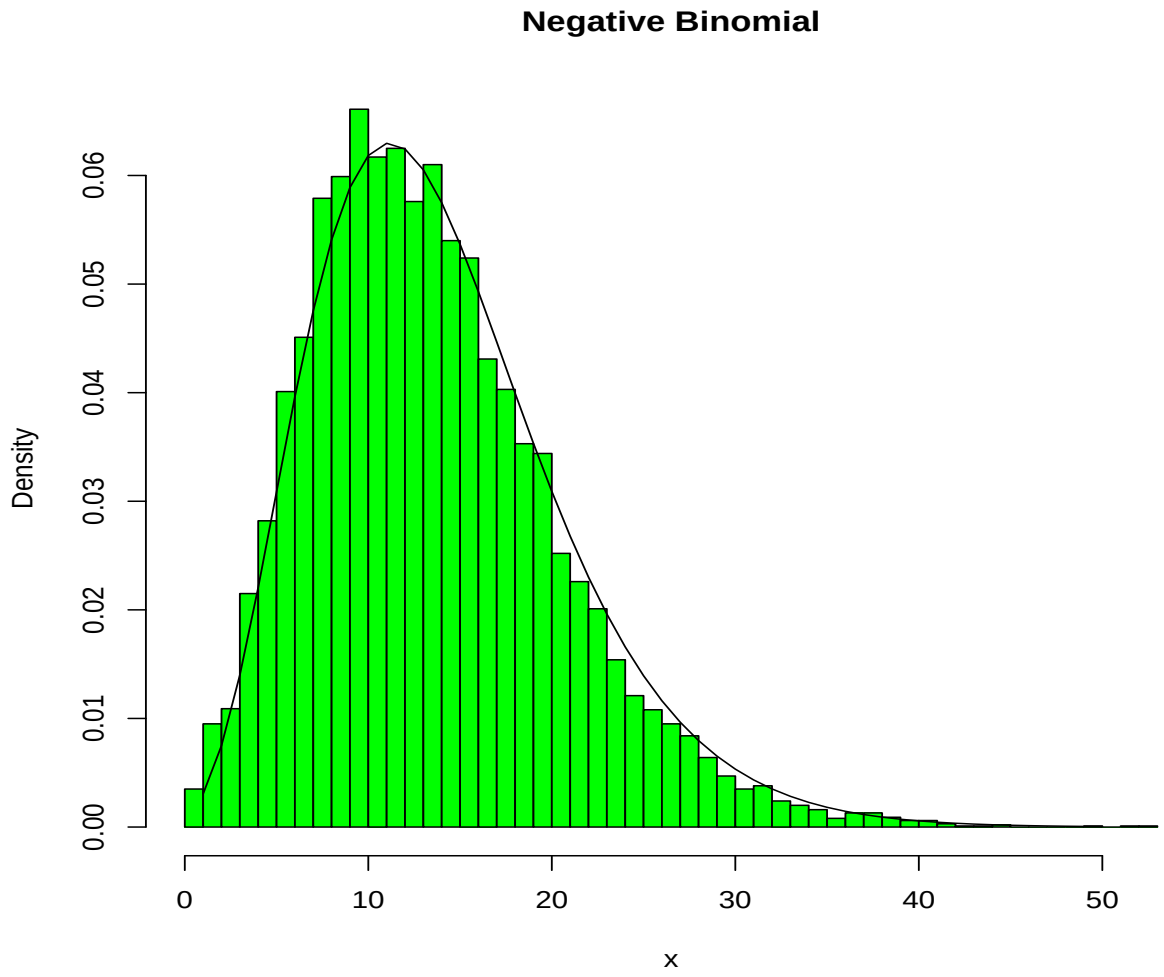
$$Y \sim \text{Gamma}(n, (1-p)/) \text{ and } X|y \sim \text{Poisson}(y)$$

implies

$$X \sim \text{Negative Binomial}(n, p)$$

Negative Binomial

```
nsim<-10000;n<-6;p<-.3;
y<-rgamma(nsim,n,p/(1-p));x<-rpois(nsim,y);
hist(x,main="Negative Binomial",freq=F,col="green",breaks=40)
par(new=T)
lines(1:50,dnbinom(1:50,n,p))
```


Negative Binomial
Density
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0
10
20
30
40
50
x

Mixture Representation

- The representation of the Negative Binomial is a particular case of a *mixture distribution*
- A mixture represents a density as the marginal of another distribution:

$$f(x) = \sum_i p_i f_i(x)$$

- To generate from $f(x)$
  - Choose f_i with probability p_i
  - Generate an observation from f_i

Section 2.3: Accept-Reject Methods

- There are many distributions from which it is difficult, or even impossible, to directly simulate by an inverse transform.
- Moreover, in some cases, we are not even able to represent the distribution in a usable form, such as a transformation or a mixture.
- We thus turn to another class of methods that only requires us to know the functional form of the density f of interest up to a multiplicative constant
- The key to this method is to use a simpler (simulationwise) density g from which the simulation is actually done. For a given density g—called the *instrumental or candidate density*— there are thus many densities f—called the *target densities*—which can be simulated this way.

The Accept-Reject Algorithm

1. Generate $X \sim g$, $U \sim \mathcal{U}_{[0,1]}$;
2. Accept $Y = X$ if $U \leq \frac{1}{M}\frac{f(X)}{g(X)}$;
3. Return to 1. otherwise.

Accept-Reject: Produces $Y \sim f$ exactly.

- Generate $X \sim g$, $U \sim$ Uniform$[0, 1]$.
- Accept $Y = X$ if $U \leq f(X)/Mg(X)$

$$\begin{aligned} P(Y \leq y | U \leq \frac{f(X)}{Mg(X)}) &= \frac{P(X \leq y, U \leq \frac{f(X)}{Mg(X)})}{P(U \leq \frac{f(X)}{Mg(X)})} \\ &= \frac{\int_{-\infty}^{y} \int_{0}^{f(x)/Mg(x)} du\ g(x)dx}{\int_{-\infty}^{\infty} \int_{0}^{f(x)/Mg(x)} du\ g(x)dx} \\ &= \frac{\int_{-\infty}^{y} \frac{f(x)}{Mg(x)}\ g(x)dx}{\int_{-\infty}^{\infty} \frac{f(x)}{Mg(x)} du\ g(x)dx} \\ &= P(Y \leq y) \end{aligned}$$

Two Interesting Properties of AR

- We can simulate from any density know up to a *multiplicative constant*
  - This is important in Bayesian calculations
  - The posterior distribution

    $$\pi(\theta|x) \propto f(x|\theta)\pi(\theta)$$

    is only specified up to a normalizing constant
- The probability of acceptance is $1/M$, and the expected number of trials until acceptance is M

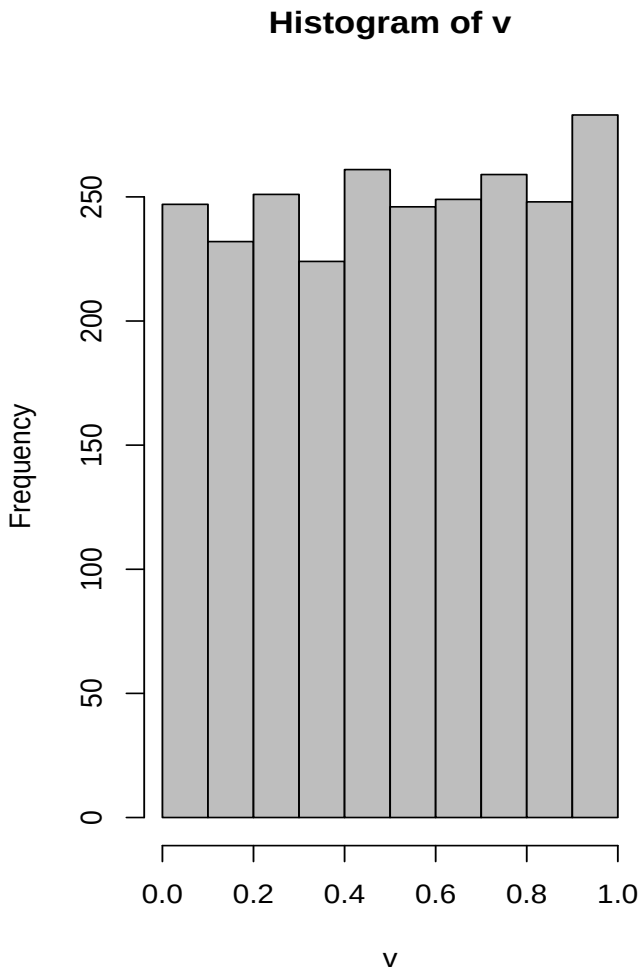
Example: Beta Accept-Reject

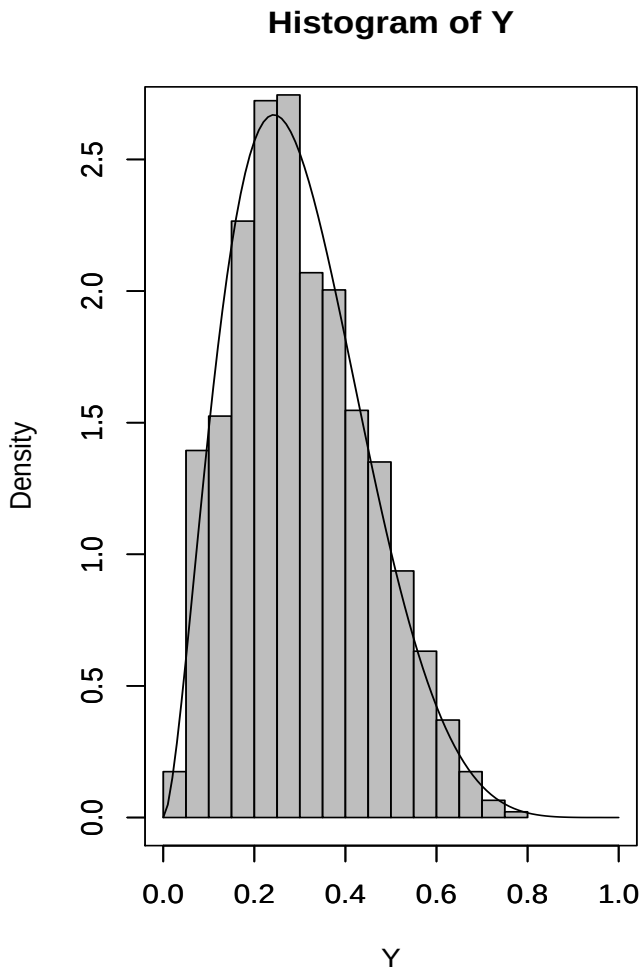
- Generate $Y \sim \text{beta}(a, b)$.
- No direct method if a and b are not integers.
- Use a uniform candidate
- For $a = 2.7$ and $b = 6.3$
  - Put the beta density $f_Y(y)$ inside a box
  - Box has sides 1 and c, where $c \geq \max_y f_Y(y)$.
- If (U, V) are independent uniform$(0, 1)$ random variables

$$P(V \leq y | U \leq \frac{1}{c} f_Y(V)) = P(Y \leq y)$$

Example: Beta Accept-Reject - Uniform Candidate

- Acceptance Rate $= 37\%$


Histogram of v
Frequency
0
50
100
150
200
250
0.0
0.2
0.4
0.6
0.8
1.0
v


Histogram of Y
Density
0.0
0.5
1.0
1.5
2.0
2.5
0.0
0.2
0.4
0.6
0.8
1.0
Y

Example: Beta Accept-Reject - Uniform Candidate

- R program: BetaAR-1

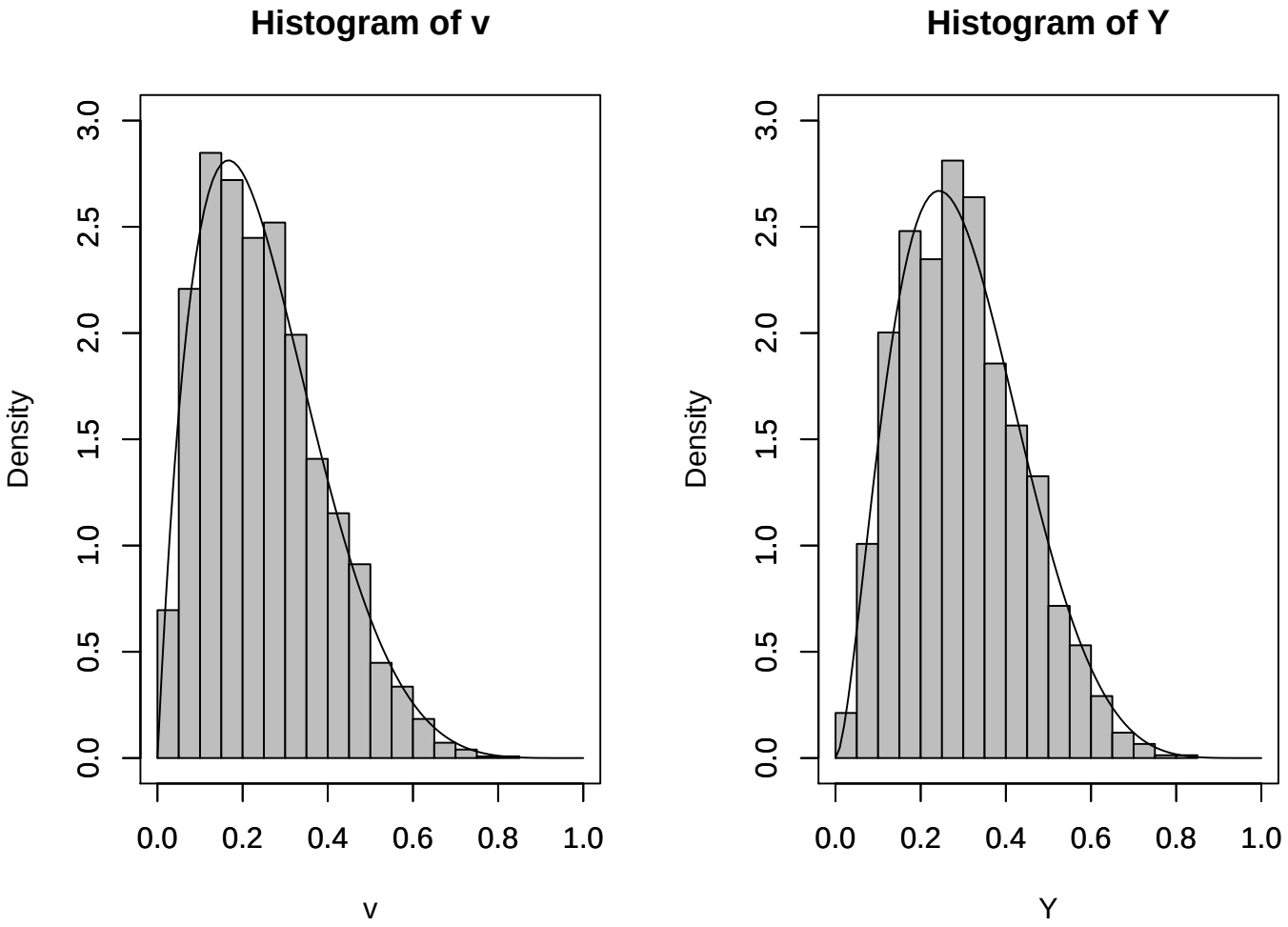
```
a<-2.7; b<-6.3; c<-2.669;nsim<-2500;
#Generate u and v#
u<-runif(nsim);v<-runif(nsim);
#---------Generate Y, the beta random variable-------------#
test<-dbeta(v, a, b)/c;          #density ratio
Y<-v*(u<test)                    #accepted values
Y<-Y[Y!=0]                       #eliminate zeros
length(Y)/nsim                   #percent accepted
#----------Plot---------------------------------------------#
par(mfrow=c(1,2))
hist(v)
hist(Y)
par(new=T)
plot(function(x)(dbeta(x, a, b)));
#-----------------------------------------------------------
```

Properties

- For c=2.669 the acceptance probability is $1/2.669 = .37$, so we accept 37%
- If we simulate from a beta(2,6), the bound is 1.67, so we accept 60%

Example: Beta Accept-Reject - Beta Candidate

- Acceptance Rate $\uparrow$ with better candidate
- Direct generation of Beta$(2, 6)$
- Acceptance Rate $= 60\%$


Histogram of v
Density
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0.0
0.2
0.4
0.6
0.8
1.0
v
Histogram of Y
Density
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0.0
0.2
0.4
0.6
0.8
1.0
Y

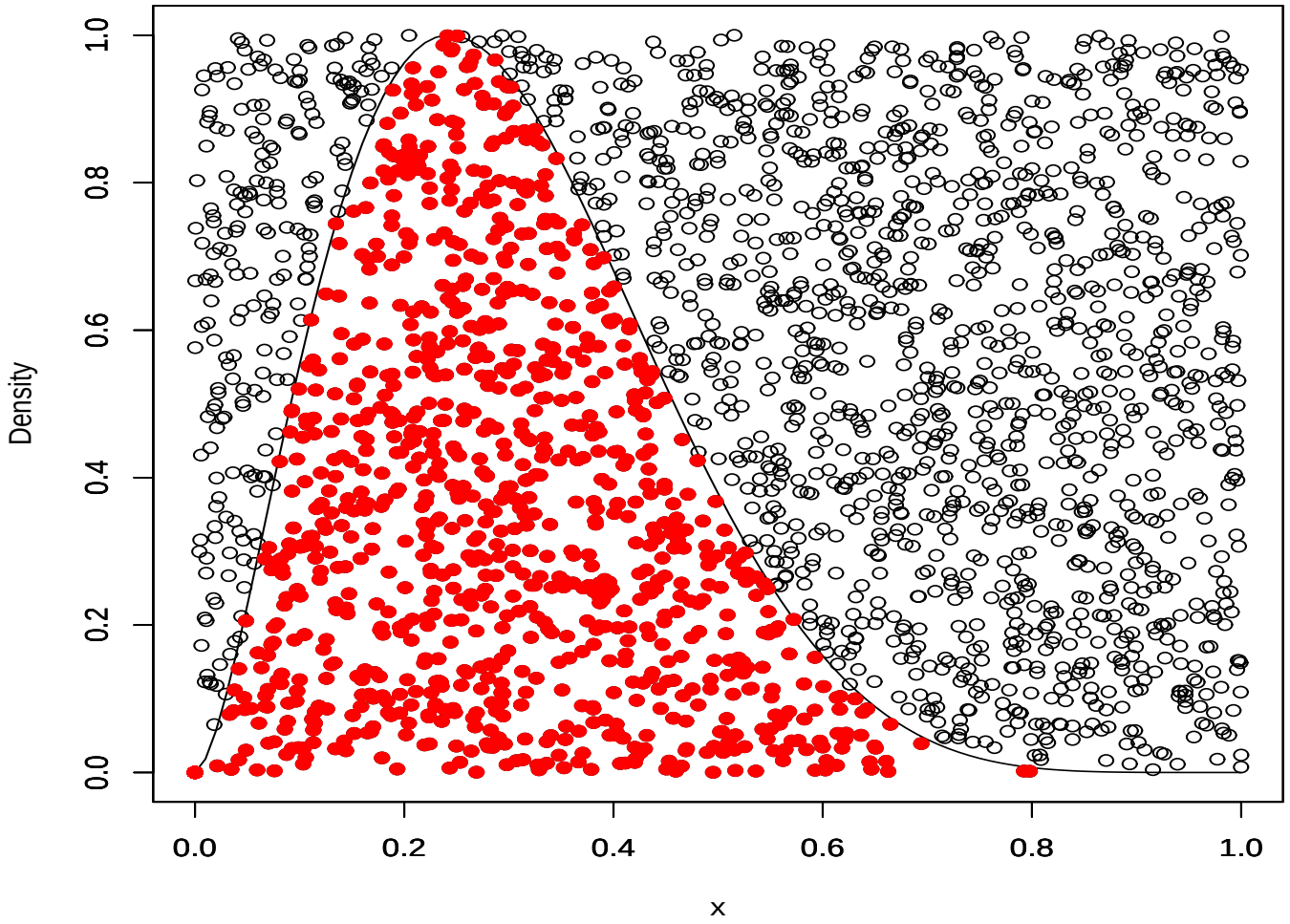
Example: Beta Accept-Reject - Beta Candidate

- R program: BetaAR-2

```
a<-2.7; b<-6.3; c<-1.67;nsim<-2500;
#Generate u and v#
u<-runif(nsim);
v<-rbeta(nsim,2,6)                  #beta candidate
#---------Generate Y, the beta random variable-------------#
test<-dbeta(v, a, b)/(c*dbeta(v, 2, 6));    #density ratio
Y<-v*(u<test)                               #accepted values
Y<-Y[Y!=0]                                  #eliminate zeros
length(Y)/nsim                              #percent accepted
#----------Plot---------------------------------------------#
par(mfrow=c(1,2))
hist(v)
par(new=T)
plot(function(x)(dbeta(x, 2, 6)))
hist(Y)
par(new=T)
plot(function(x)(dbeta(x, a, b)));
```

Beta AR Generation - Some Intuition

- Uniform Candidate
- Accepted Values are Under Density


Density
0.0
0.2
0.4
0.6
0.8
1.0
x

Example: Normal from Cauchy

- Normal: $f(x) = \frac{1}{\sqrt{2\pi}} \exp(-x^2/2)$
- Cauchy: $g(x) = \frac{1}{\pi} \frac{1}{1+x^2}$
- $f/g = \sqrt{\frac{\pi}{2}}(1 + x^2) \exp(-x^2/2) \leq \sqrt{\frac{2\pi}{e}} = 1.52$ attained at $x = \pm 1$.
- Prob. of acceptance $= 1/1.52 = .66$
- Mean number of trials to success $= 1.52$

Example 2.18: Normals from Double Exponential

- Generate Normal$(0, 1)$ from a Double Exponential with density

$$g(x|\alpha) = (\alpha/2)\exp(-\alpha x)$$

  - Minimum bound at $\alpha = 1$
  - Acceptance probability = .76

Example 2.19: Gamma Random Variables - Non Integer Shape

- Illustrates power of AR
- Gamma = sum of exponentials only if α an integer - no Chi Squared.
- Generate $f(x) = \frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x}$,

  $\beta = 1$ without loss of generality.
- Candidate density $g(x) = \frac{1}{\Gamma(a) b^a} x^{\alpha-1} e^{-x/b}$
- Then if $\alpha > a$ and $b > 1$

$$\frac{f(x)}{g(x)} \propto \frac{x^{\alpha-a}}{b^a} e^{(1/b-1)x} < \infty.$$

- Take $a = [\alpha]$. Then $b = [\alpha]/a$ minimizes M

Example 2.20: Truncated Normal distributions

- Truncated normal distributions are very useful (censoring).
- For the constraint $x > a$, the density $f_a(x)$ is proportional to

$$f_a(x) \propto e^{-\frac{1}{2\sigma^2}(x-\mu)^2} I(x > a)$$

- Naive method: generate $Y \sim N(\mu, \sigma^2)$ until $Y > a$
- Can sometimes work, but requires, on the average, $1/\Phi((\mu - a)/\sigma)$ simulations to get one random variable.
- For $a = \mu + 2\sigma$, need 44 simulations for each acceptance.

Truncated Normal distributions

- Better: Use a translated exponential distribution

$$g(x) \propto \alpha e^{-\alpha(x-a)} I(x > a)$$

- For $a = \mu + 2\sigma$, need less than 12 simulations for each acceptance.

Truncated Normal - Some Details

- The Accept-Reject ratio is

$$\frac{f(x)}{g(x)} = \frac{e^{-\frac{1}{2\sigma^2}(x-\mu)^2} I(x > a)}{\alpha e^{-\alpha(x-a)} I(x > a)}$$

  - These are unnormalized densities
  - We don't need to worry about the constants

- If $\alpha > a$

$$M = \max_{x>a} \frac{f(x)}{g(x)} = \frac{1}{\alpha}\, e^{\frac{1}{2}(\alpha^2 - 2\alpha a)},$$

  attained at $x = \alpha$.

- Can further optimize by minimizing in α

Truncated Normal - Some Details

- For simplicity, we will take $\alpha = a$ so that

$$M = \frac{1}{a}\, e^{-\frac{1}{2}a^2}$$

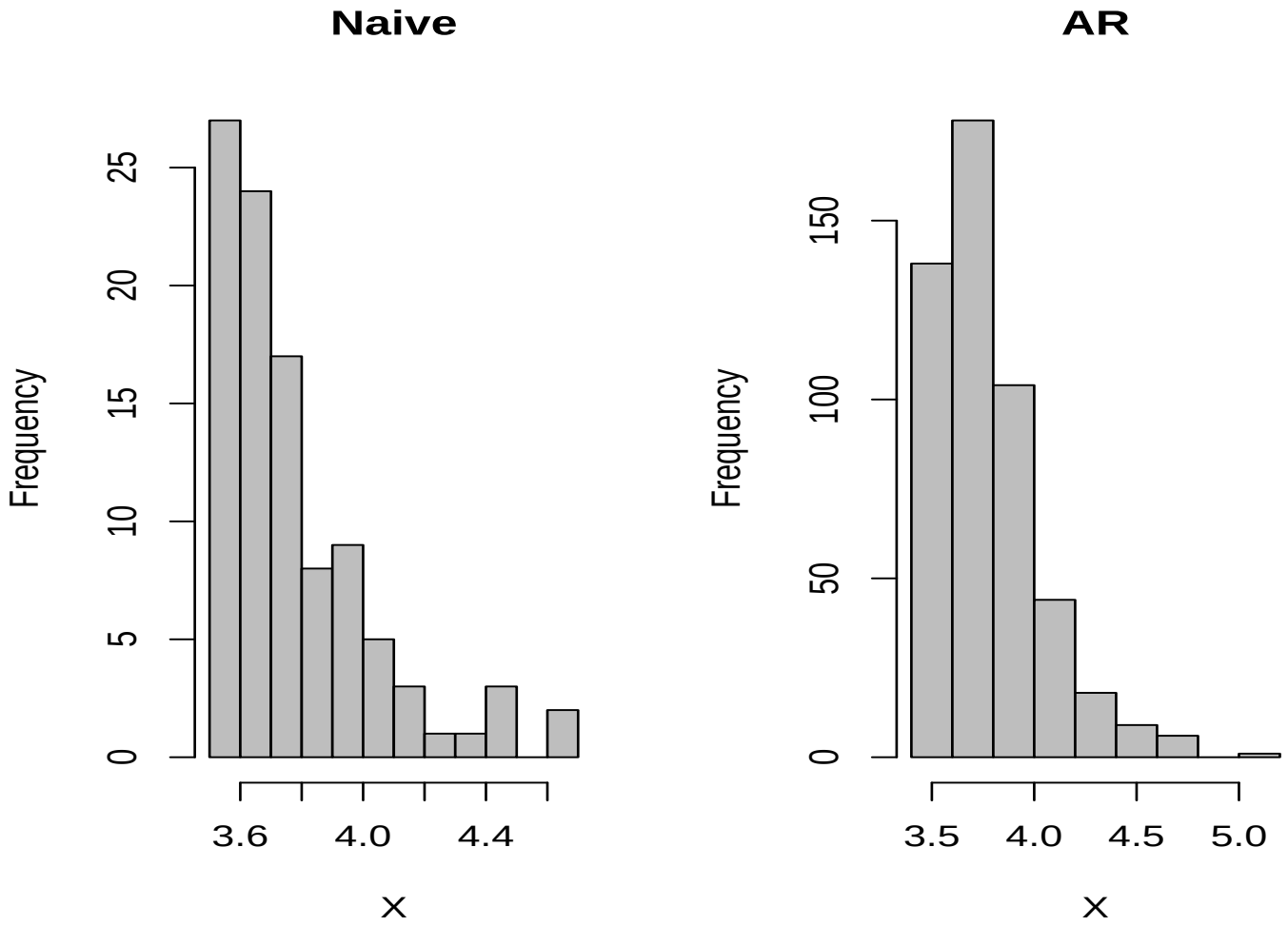
- and

$$\frac{f(x)}{Mg(x)} = a\, e^{-\frac{1}{2}(x-a)^2}$$

- Now lets compare AR to "naive" simulation
  - Generate 100 random variables
  - Take $a = 1$ and $a = 3.5$

Example: Truncated Normal

- Samples generated Naively and with AR
- Acceptance Rate very high for AR



- R Program "Truncated"

Chapter 3: Monte Carlo Integration

- Two major classes of numerical problems that arise in statistical inference
  - *optimization* problems
  - *integration* problems
- Although optimization is generally associated with the likelihood approach, and integration with the Bayesian approach, these are not strict classifications

Example 3.1 Bayes Estimator

- In general, the Bayes estimate under the loss function $L(\theta, \delta)$ and the prior π is the solution of the minimization program

$$\min_{\delta} \int_{\Theta} L(\theta, \delta)\, \pi(\theta)\, f(x|\theta)\, d\theta\ .$$

- Only when the loss function is the quadratic function $\|\theta - \delta\|^2$ will the Bayes estimator be a posterior expectation.
- For $L(\theta, \delta) = |\theta - \delta|$, the Bayes estimator associated with π is the posterior median of $\pi(\theta|x)$, $\delta^{\pi}(x)$, which is the solution to the equation

$$\int_{\theta \leq \delta^{\pi}(x)} \pi(\theta)\, f(x|\theta)\, d\theta = \int_{\theta \geq \delta^{\pi}(x)} \pi(\theta)\, f(x|\theta)\, d\theta\ .$$

Section 3.2: Classical Monte Carlo Integration

- Generic problem of evaluating the integral

$$\mathrm{E}_f[h(X)] = \int_{\mathcal{X}} h(x)\, f(x)\, dx\ .$$

- Based on previous developments, it is natural to propose using a sample $(X_1, \ldots, X_m)$ generated from the density f
- Approximate the integral by the empirical average
- This approach is often referred to as the *Monte Carlo method*

Strong Law

- For a sample $(X_1, \ldots, X_m)$, the empirical average

$$\overline{h}_m = \frac{1}{m} \sum_{j=1}^{m} h(x_j) \ ,$$

  converges almost surely to

$$\mathrm{E}_f[h(X)]$$

- This is the Strong Law of Large Numbers

Central Limit Theorem

- Estimate the variance with

$$\mathrm{var}(\overline{h}_m) = \frac{1}{m} \int_{\mathcal{X}} \left(h(x) - \mathrm{E}_f[h(X)]\right)^2 f(x)dx$$

- For m large,

$$\frac{\overline{h}_m - \mathrm{E}_f[h(X)]}{\sqrt{v_m}}$$

  is therefore approximately distributed as a $\mathcal{N}(0,1)$ variable

- This leads to the construction of a convergence test and of confidence bounds on the approximation of $\mathrm{E}_f[h(X)]$.

Example 3.4: Monte Carlo Integration

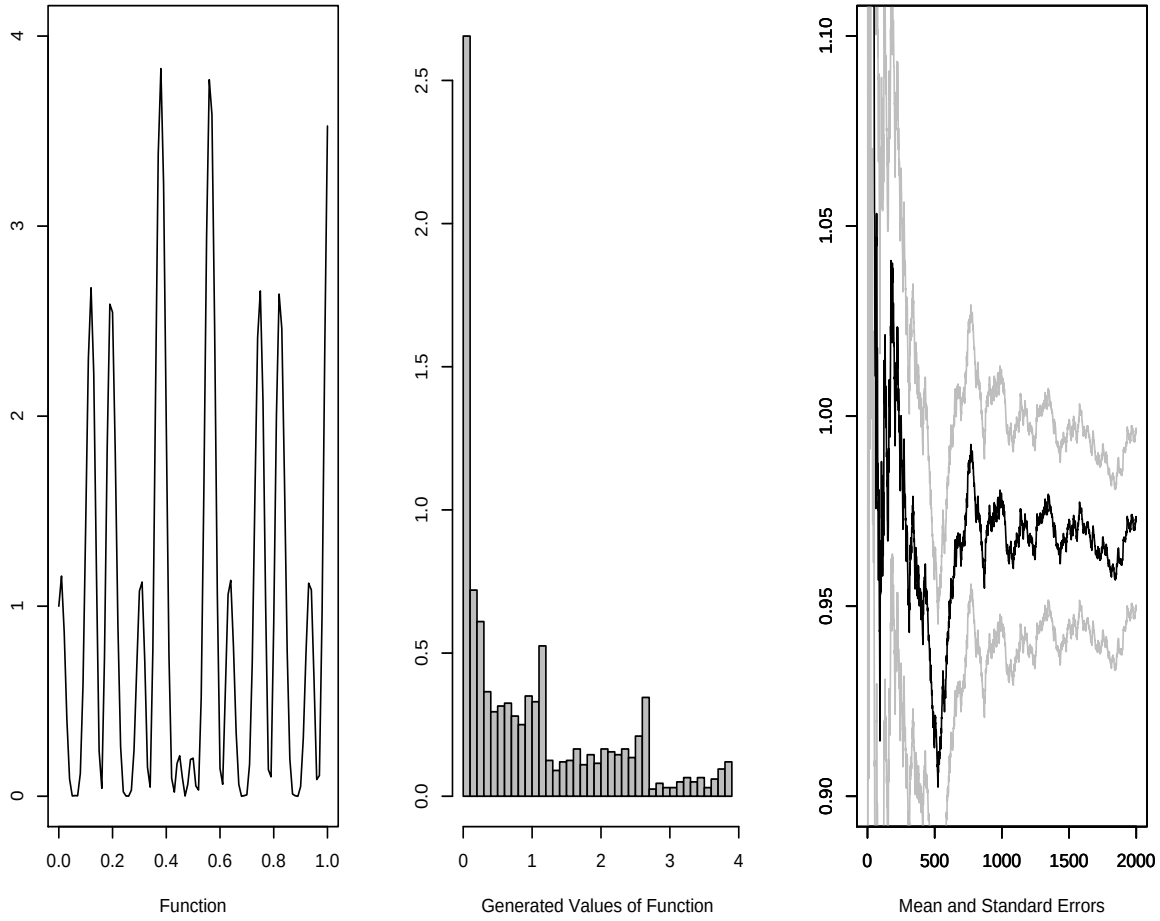
- Recall the function that we saw in the Newton-Raphson example:

$$h(x) = [\cos(50x) + \sin(20x)]^2$$

.

- To calculate the integral, we generate $U_1, U_2, \ldots, U_n$ iid $\mathcal{U}(0, 1)$ random variables, and approximate $\int h(x)dx$ with $\sum h(U_i)/n$.
- It is clear that the Monte Carlo average is converging, with value of 0.963 after $10,000$ iterations.

```
nsim<-10000;u<-runif(nsim);
#The function to be integrated
mci.ex <- function(x){(cos(50*x)+sin(20*x))^2}
plot(function(x)mci.ex(x), xlim=c(0,1),ylim=c(0,4))
#The monte carlo sum
sum(mci.ex(u))/nsim
```


0
1
2
3
4
0.0
0.2
0.4
0.6
0.8
1.0
Function
0.0
0.5
1.0
1.5
2.0
2.5
0
1
2
3
4
Generated Values of Function
0.90
0.95
1.00
1.05
1.10
0
500
1000
1500
2000
Mean and Standard Errors

Example 3.5: Normal CDF

- The approximation of

$$\Phi(t) = \int_{-\infty}^{t} \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy$$

  by the Monte Carlo method is

$$\hat{\Phi}(t) = \frac{1}{n} \sum_{i=1}^{n} I_{x_i \leq t},$$

- With (exact) variance $\Phi(t)(1 - \Phi(t))/n$
- The variables $I_{x_i \leq t}$ are independent Bernoulli with success probability $\Phi(t)$.
- Method breaks down for tail probabilities

Section 3.3 Importance Sampling

- Simulation from the true density f is not necessarily optimal
- The method of *importance sampling* is an evaluation of $\mathrm{E}_f[h(X)]$ based based on the alternative representation

$$\mathrm{E}_f[h(X)] = \int_{\mathcal{X}} h(x)\, f(x)\, dx = \int_{\mathcal{X}} h(x)\, \frac{f(x)}{g(x)}\, g(x)\, dx\ ,$$

- We generate a sample $X_1, \ldots, X_n$ from a given distribution g and approximating

$$\mathrm{E}_f[h(X)] \approx \frac{1}{m} \sum_{j=1}^{m} \frac{f(X_j)}{g(X_j)}\, h(X_j)\ .$$

- The Strong Law guarantees

$$\frac{1}{m} \sum_{j=1}^{m} \frac{f(X_j)}{g(X_j)}\, h(X_j) \to \mathrm{E}_f[h(X)]$$

Example - Normal Tail Probabilities

- For $a = 3.5, 4.5, 5.5$, calculate $P(Z > a) = \int_a^\infty \phi(x)dx$
- "Naive" approach: $X_i \sim N(0,1)$,

$$\int_a^\infty \phi(x)dx = \mathrm{E}I(X > a)$$

so

$$\frac{1}{n}\sum_{i=1}^{n} I(X_i > a) \to \int_a^\infty \phi(x)dx$$

Example - Normal Tail Probabilities - 2

- Importance sampling: $X_i \sim g(x) = e^{-(x-a)}, \quad x > a,$

$$\int_a^\infty \phi(x)dx = \int_a^\infty \left[\frac{\phi(x)}{g(x)}\right] g(x)\, dx$$

so

$$\frac{1}{n}\sum_{i=1}^{n} \frac{\phi(X_i)}{e^{-(X_i-a)}} \to \int_a^\infty \phi(x)dx$$

- And one more....

Example - Normal Tail Probabilities - 3

- Transform to Uniform

$$\int_a^\infty \phi(x)dx = \int_0^{1/a} \frac{\phi(1/y)}{y^2}\, dy, \qquad y = 1/x$$

- For $U_i \sim \text{Uniform}(0, 1/a)$ with density $g(x) = a$

$$\frac{1}{n}\sum_{i=1}^{n} \frac{\phi(1/U_i)}{aU_i^2} \to \int_a^\infty \phi(x)dx$$

- Can monitor convergence with standard deviation
- R Program TruncatedIS
- **Also - Multivariate Normal Tails**
  - R Program MultivariateTruncatedIS

Importance Sampling Facts

- Candidate g needs to have heavier tails than target f
- The same sample g can be used for many targets f
  - This cuts down error in Monte Carlo comparisons
- Alternative form

$$\sum_{j=1}^{m} \left(\frac{\frac{f(X_j)}{g(X_j)}}{\sum_j \frac{f(X_j)}{g(X_j)}} \right) h(X_j)$$

  - Biased, but with smaller variance
  - Often beats unbiased estimator in MSE
  - Strong Law applies

Example 3.13: Student's t

- $X \sim \mathcal{T}(\nu, \theta, \sigma^2)$, with density

$$f(x) = \frac{\Gamma((\nu+1)/2)}{\sigma\sqrt{\nu\pi}\,\Gamma(\nu/2)} \left(1 + \frac{(x-\theta)^2}{\nu\sigma^2}\right)^{-(\nu+1)/2} .$$

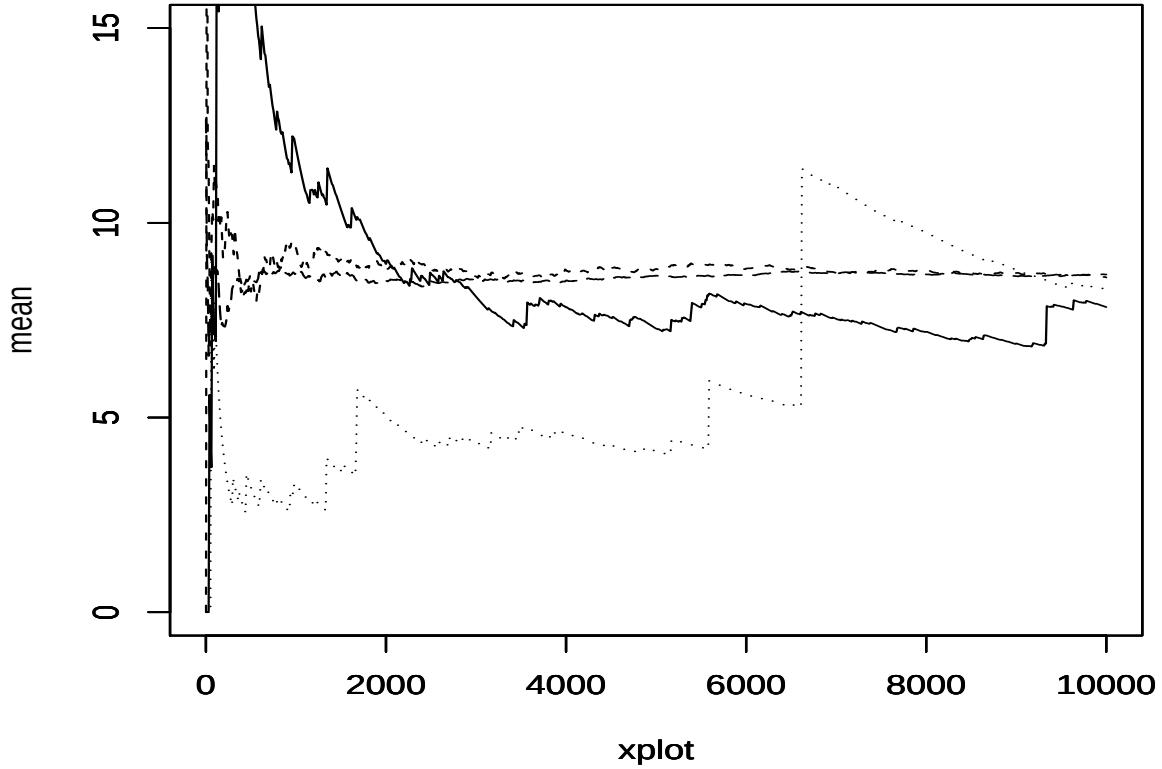
- Take $\theta = 0$ and $\sigma = 1$.
- Estimate

$$\int_{2.1}^{\infty} x^5 f(x) dx$$

- Candidates
  - f itself
  - Cauchy
  - Normal
  - Uniform$(0, 1/2.1)$

Importance Sampling Comparisons

- f (solid), Cauchy (short dash), Normal (dots), Uniform(long dash)
- Uniform candidate the best



- R program "Students-t-moment"

Chapter 5: Monte Carlo Optimization

- Differences between the numerical approach and the simulation approach to the problem

$$\max_{\theta \in \Theta} h(\theta)$$

  lie in the treatment of the function h.

- In an optimization problem using deterministic numerical methods
  - The analytical properties of the target function (convexity, boundedness, smoothness) are often paramount.

- For the simulation approach
  - We are concerned with h from a probabilistic (rather than analytical) point of view.

Monte Carlo Optimization

- The problem

$$\max_{\theta \in \Theta} h(\theta)$$

- Deterministic numerical methods $\longrightarrow$ analytical properties
- Simulation approach $\longrightarrow$ probabilistic view.
  - This dichotomy is somewhat artificial
  - Some simulation approaches have no probabilistic interpretation
- Nonetheless, the use of the analytical properties of h plays a lesser role in the simulation approach.

Two Simulation Approaches

- Exploratory Approach
  - Goal: To optimize h by describing its entire range
  - Actual properties of h play a lesser role
- Probabilistic Approach
  - Monte Carlo exploits probabilistic properties of h
  - This approach tied to missing data methods

Section 5.2: Stochastic Exploration

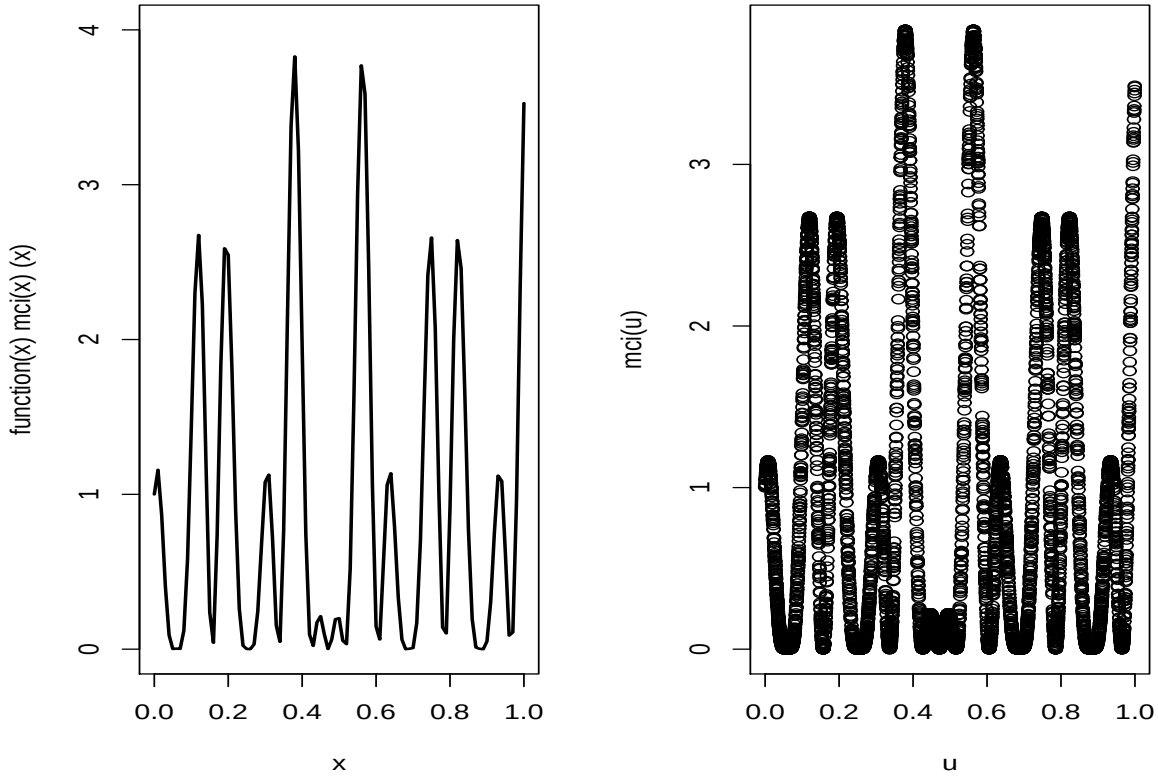
- A first approach is to simulate from a uniform distribution on Θ, $u_1, \ldots, u_m \sim \mathcal{U}_\Theta$,
- Use the approximation

$$h_m^* = \max(h(u_1), \ldots, h(u_m)).$$

- This method converges (as m goes to ∞), but it may be very slow since it does not take into account any specific feature of h.
- Distributions other than the uniform, which can possibly be related to h, may then do better.
- In particular, in setups where the likelihood function is extremely costly to compute the number of evaluations of the function h is best kept to a minimum.

Example 5.2: A first Monte Carlo maximization

- Recall the function $h(x) = [\cos(50x) + \sin(20x)]^2$.
- we try our naïve strategy and simulate $u_1, \ldots, u_m \sim \mathcal{U}(0, 1)$, and use the approximation $h_m^* = \max(h(u_1), \ldots, h(u_m))$



A Probabilistic Approach

- If h is positive with $\int h < \infty$
  - Finding $\max h$ is the same as
  - Finding the modes of h
- $h \longrightarrow \exp(h)$ makes h positive

Properties

- Many local minima
- Standard methods may not find global minimum
- We can simulate from $\exp(-h(x, y))$
- Get minimum from $\min_i h(x_i, y_i)$
- Can use other methods...

Deterministic Gradient Methods

- The *gradient method* is a deterministic numerical approach to the problem

$$\max_{\theta\in\Theta} h(\theta).$$

- It produces a sequence (θ_j) that converges to the maximum when
- ◦ the domain $\Theta \subset \Re^d$
  ◦ the function $(-h)$

  are both convex.
- The sequence (θ_j) is constructed in a recursive manner through

$$\theta_{j+1} = \theta_j + \alpha_j \nabla h(\theta_j)\ , \qquad \alpha_j > 0\ ,$$

Here

◦ ∇h is the gradient of h

◦ α_j is chosen to aid convergence

Stochastic Variant

- There are stochastic variants of the gradient method
- They do not always go along the steepest slope
- This is an advantage, as it can avoid local maxima and saddlepoints
- The best, and simple version is Simulated Annealing/Metropolis Algorithm

Simulated Annealing

- This name is borrowed from Metallurgy:
- A metal manufactured by a slow decrease of temperature ($annealing$) is stronger than a metal manufactured by a fast decrease of temperature.
- The fundamental idea of simulated annealing methods is that a change of scale, called $temperature$, allows for faster moves on the surface of the function h to maximize.
  - Rescaling partially avoids the trapping attraction of local maxima.
  - As T decreases toward 0, the values simulated from this distribution become concentrated in a narrower and narrower neighborhood of the local maxima of h

Metropolis Algorithm/Simulated Annealing

- Simulation method proposed by Metropolis *et al.* (1953)
- Starting from θ_0, ζ is generated from

$$\zeta \sim \text{Uniform in a neighborhood of } \theta_0.$$

- The new value of θ is generated as

$$\theta_1 = \begin{cases} \zeta & \text{with probability } \rho = \exp(\Delta h/T) \wedge 1 \\ \theta_0 & \text{with probability } 1-\rho, \end{cases}$$

  - $\Delta h = h(\zeta) - h(\theta_0)$
  - If $h(\zeta) \geq h(\theta_0)$, ζ is accepted
  - If $h(\zeta) < h(\theta_0)$, ζ may still be accepted
  - This allows escape from local maxima

Metropolis/Simulated Annealing Algorithm

- In its most usual implementation, the simulated annealing algorithm modifies the temperature T at each iteration
- It has the form

1. Simulate ζ from an instrumental distribution with density $g(|\zeta - \theta_i|)$;
2. Accept $\theta_{i+1} = \zeta$ with probability

$$\rho_i = \exp\{\Delta h_i / T_i\} \wedge 1;$$

   take $\theta_{i+1} = \theta_i$ otherwise.
3. Update T_i to T_{i+1}.

Metropolis/Simulated Annealing Algorithm - Comments

1. Simulate ζ from an instrumental distribution with density $g(|\zeta - \theta_i|)$;
2. Accept $\theta_{i+1} = \zeta$ with probability

$$\rho_i = \exp\{\Delta h_i / T_i\} \wedge 1;$$

   take $\theta_{i+1} = \theta_i$ otherwise.
3. Update T_i to T_{i+1}.

- All positive moves accepted
- As $T \downarrow 0$
  - Harder to accept downward moves
  - No big downward moves
- Not a Markov Chain - difficult to analyze

Simple Example Revisited

- Recall the function $h(x) = [\cos(50x) + \sin(20x)]^2$
- The specific algorithm we use is

  Starting at iteration t, the iteration is at $(x^{(t)}, h(x^{(t)}))$:

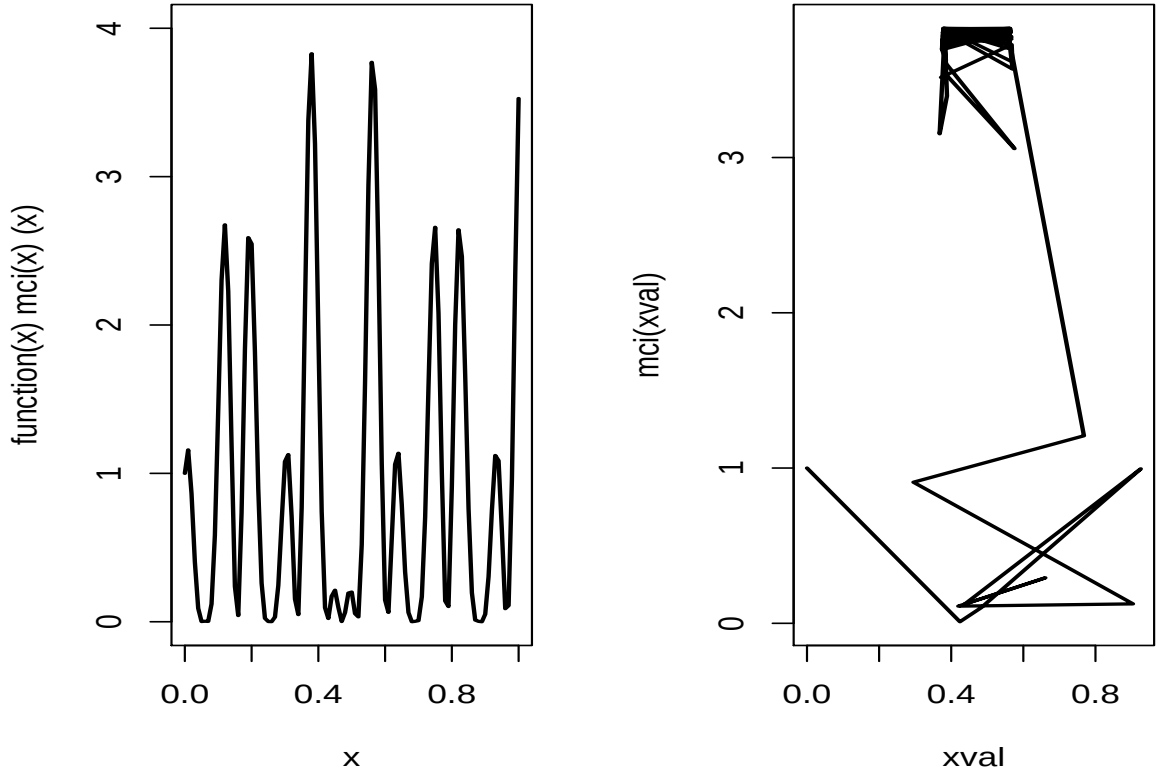
  1. Simulate $u \sim \mathcal{U}(a_t, b_t)$ where $a_t = \max(x^{(t)} - r, 0)$ and $b_t = \min(x^{(t)} + r, 1)$
  2. Accept $x^{(t+1)} = u$ with probability

$$\rho^{(t)} = \min\left\{\exp\left(\frac{h(u) - h(x^{(t)})}{T_t}\right), 1\right\},$$

     take $x^{(t+1)} = x^{(t)}$ otherwise.
  3. Update T_t to T_{t+1}.

- The value of r controls the size of the interval around the current point (staying in $(0, 1)$)
- The value of T_t controls the cooling.

The Trajectory



- Left Panel is the function
- Right Panel is the Simulated Annealing Trajectory

R Program

```
par(mfrow=c(1,2))
#The function to be optimized
mci <- function(x){(cos(50*x)+sin(20*x))^2}
plot(function(x)mci(x), xlim=c(0,1),ylim=c(0,4),lwd=2)
#optimize(mci, c(0, 1), tol = 0.0001, maximum=TRUE)
#The monte carlo maximum
nsim<-2500
u<-runif(nsim)
#Simulated annealing
xval<-array(0,c(nsim,1));r<-.5
for(i in 2:nsim){
test<-runif(1, min=max(xval[i-1]-r,0),max=min(xval[i-1]+r,1));
delta<-mci(test)-mci(xval[i-1]);
rho<-min(exp(delta*log(i)/1),1);
xval[i]<-test*(u[i]<rho)+xval[i-1]*(u[i]>rho)
}
mci(xval[nsim])
plot(xval,mci(xval),type="l",lwd=2)
```

Simulated Annealing Property

- Theorem 5.7: Under mild assumptions, the Simulated Annealing algorithm is guaranteed to find the global maximum

Return to the difficult maximization

- Apply simulated Annealing
- Different choices of T_i
  - Results dependent on choice of T_i
  - $T_i \propto 1/\log(i+1)$ preferred

Simulated Annealing Runs

- $g \sim$ Uniform$(-.1, .1)$
- Starting point $(0.5, 0.4)$

Case	T_i	θ_T	$h(\theta_T)$	$\min_t h(\theta_t)$	Accept. rate
1	$1/10i$	$(-1.94, -0.480)$	0.198	$4.02\,10^{-7}$	0.9998
2	$1/\log(1+i)$	$(-1.99, -0.133)$	3.408	3.823×10^{-7}	0.96
3	$100/\log(1+i)$	$(-0.575, 0.430)$	0.0017	4.708×10^{-9}	0.6888
4	$1/10\log(1+i)$	$(0.121, -0.150)$	0.0359	2.382×10^{-7}	0.71

- Case 3 explores "valley" near the minimum
- Recommended $T_i \approx \Gamma/\log(i+1)$ for large Γ

Section 5.3: Missing Data

- Methods that work directly with the objective function are less concerned with fast exploration of the space
- Need to be concerned wheh approximating an objective function - we may introduce an additional level of error
- Many of these methods work well in Missing data models, where the likelihood $g(x|\theta)$ can be expressed as

$$g(x|\theta) = \int_{\mathcal{Z}} f(x, z|\theta)\, dz$$

- More generally, the function $h(x)$ to be optimized can be expressed as the expectation

$$h(x) = \mathrm{E}[H(x, Z)]$$

Example 5.14: Censored data likelihood

- Observe $Y_1, \ldots, Y_n$, iid, from $f(y-\theta)$
- Order the observations so that $\mathbf{y} = (y_1, \cdots, y_m)$ are uncensored and $(y_{m+1}, \ldots, y_n)$ are censored (and equal to a).

- The observed likelihood function is

$$L(\theta|\mathbf{y}) = \prod_{i=1}^{m} [1 - F(a-\theta)]^{n-m} f(y_i - \theta) \ ,$$

  where F is the cdf associated with f.

- If we had observed the last $n-m$ values, say $\mathbf{z} = (z_{m+1}, \ldots, z_n)$, with $z_i > a$ $(i = m+1, \ldots, n)$, we could have constructed the complete data likelihood

$$L^c(\theta|\mathbf{y}, \mathbf{z}) = \prod_{i=1}^{m} f(y_i - \theta) \prod_{i=m+1}^{n} f(z_i - \theta),$$

  with which it often is easier to work.

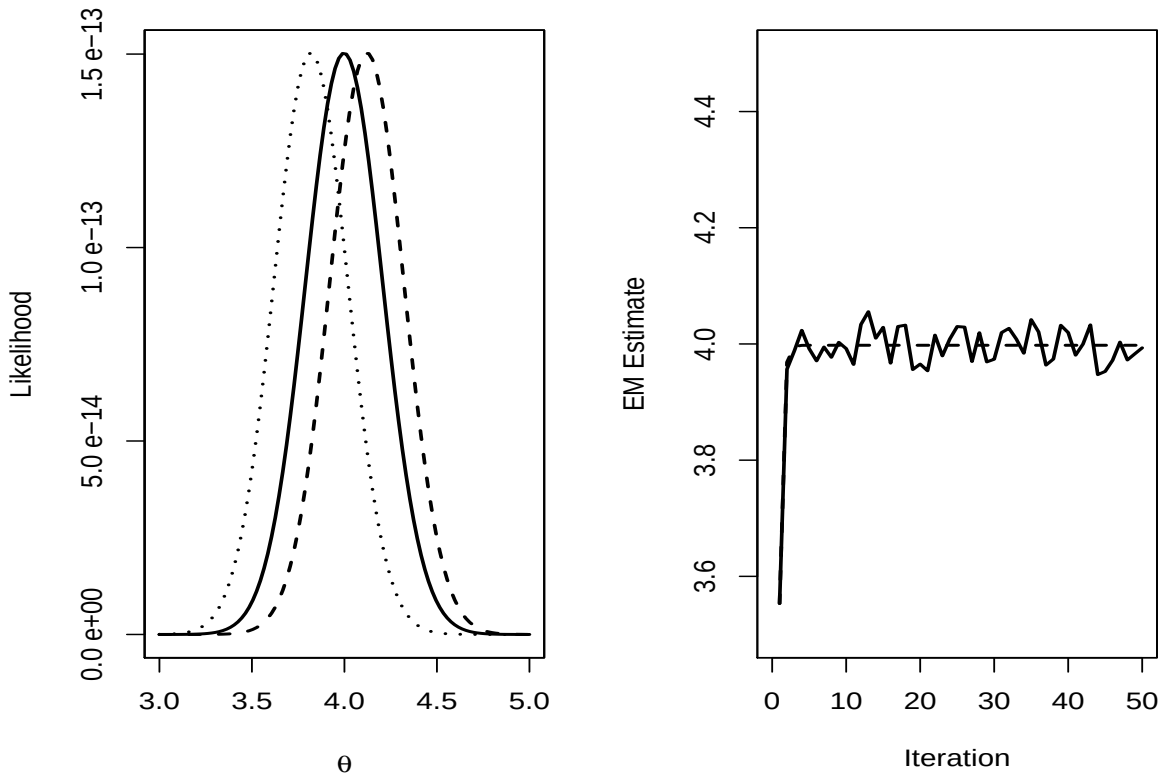
- Note that

$$L(\theta|\mathbf{y}) = \mathrm{E}[L^c(\theta|\mathbf{y}, \mathbf{Z})] = \int_{\mathcal{Z}} L^c(\theta|\mathbf{y}, \mathbf{z}) f(\mathbf{z}|\mathbf{y}, \theta) \, d\mathbf{z},$$

  where $f(\mathbf{z}|\mathbf{y}, \theta)$ is the density of the missing data conditional on the observed data.

Three Likelihoods

- For $f(y-\theta) = \mathcal{N}(\theta, 1)$ three likelihoods are shown
  - leftmost (dotted):values greater than 4.5 are replaced by the value 4.5
  - center (solid): observed data likelihood
  - rightmost (dashed): the actual data.
- Right panel: EM/MCEM algorithms


Likelihood
0.0 e+00
5.0 e−14
1.0 e−13
1.5 e−13
3.0
3.5
4.0
4.5
5.0
θ
EM Estimate
3.6
3.8
4.0
4.2
4.4
0
10
20
30
40
50
Iteration

Section 5.3.2: The EM Algorithm

- Dempster, Laird and Rubin (1977)
- Takes advantage of the representation

$$g(x|\theta) = \int_{\mathcal{Z}} f(x, z|\theta)\, dz$$

  - Solves a sequence of easier maximization problems
  - Limit is the answer to the original problem

EM Details

- Observe $X_1, \ldots, X_n$, iid from $g(x|\theta)$ and want to compute

$$\hat{\theta} = \arg\max L(\theta|\mathbf{x}) = \prod_{i=1}^{n} g(x_i|\theta)$$

- We augment the data with $\mathbf{z}$, where $\mathbf{X}, \mathbf{Z} \sim f(\mathbf{x}, \mathbf{z}|\theta)$

  - Note the basic EM Identity

$$k(\mathbf{z}|\theta, \mathbf{x}) = \frac{f(\mathbf{x}, \mathbf{z}|\theta)}{g(\mathbf{x}|\theta)},$$

    where $k(\mathbf{z}|\theta, \mathbf{x})$ is the conditional distribution of the missing data $\mathbf{Z}$ given the observed data $\mathbf{x}$.

EM Details - continued

- The identity leads to the following relationship between the
  - complete-data likelihood $L^c(\theta|\mathbf{x},\mathbf{z})$
  - observed data likelihood $L(\theta|\mathbf{x})$.

  For any value θ_0,

$$\log L(\theta|\mathbf{x}) = \mathrm{E}_{\theta_0}[\log L^c(\theta|\mathbf{x},\mathbf{z})] - \mathrm{E}_{\theta_0}[\log k(\mathbf{z}|\theta,\mathbf{x})],$$

  where the expectation is with respect to $k(\mathbf{z}|\theta_0,\mathbf{x})$.

- To maximize $\log L(\theta|\mathbf{x})$, we only have to deal with the first term on the right side, as the other term can be ignored.

EM Details - continued

- Note that

$$\mathrm{E}_{\theta_0}[\log L^c(\theta|\mathbf{x},\mathbf{z})] = \int \log L^c(\theta|\mathbf{x},\mathbf{z}) k(\mathbf{z}|\theta_0,\mathbf{x}) d\mathbf{z}$$

- Given θ_0,
  - we then maximize $\mathrm{E}_{\theta_0}[\log L^c(\theta|\mathbf{x},\mathbf{z})]$ in θ

- A sequence of estimators $\hat{\theta}_{(j)}$, $j = 1, 2, \ldots$, is obtained iteratively

$$\mathrm{E}_{\hat{\theta}_{(j-1)}}[\log L^c(\hat{\theta}_{(j)}|\mathbf{x},\mathbf{z})] = \max_{\theta} \mathrm{E}_{\hat{\theta}_{(j-1)}}[\log L^c(\theta|\mathbf{x},\mathbf{z})].$$

EM Details - continued

- The iteration contains both an expectation step and a maximization step, giving the algorithm its name.

  1. Compute
  $$\mathrm{E}_{\hat{\theta}_{(m)}}[\log L^c(\theta|\mathbf{x},\mathbf{z})]\,,$$
  where the expectation is with respect to $k(\mathbf{z}|\hat{\theta}_m,\mathbf{x})$ $(the\ E\text{-}step)$.

  2. Maximize $\mathrm{E}_{\hat{\theta}_{(m)}}[\log L^c(\theta|\mathbf{x},\mathbf{z})]$ in θ and take $(the\ M\text{-}step)$
  $$\theta_{(m+1)} = \arg\max_{\theta}\ \mathrm{E}_{\hat{\theta}_{(m)}}[\log L^c(\theta|\mathbf{x},\mathbf{z})].$$

- The iterations are conducted until a fixed point is obtained.

EM Theorem

- Theoretical core of the EM Algorithm
  - by maximizing $\mathrm{E}_{\hat{\theta}_{(m)}}[\log L^c(\theta|\mathbf{x},\mathbf{z})]$ at each step
  - the observed data likelihood on the left is increased at each step.

Theorem 5.15

The sequence $(\hat{\theta}_{(j)})$ satisfies

$$L(\hat{\theta}_{(j+1)}|\mathbf{x}) \geq L(\hat{\theta}_{(j)}|\mathbf{x}).$$

Genetic Linkage

- The classic missing data example
- 197 animals are distributed into four catagories

$$(x_1, x_2, x_3, x_4) = (125, 18, 20, 34)$$

and modeled with the multinomial distribution

$$\mathcal{M}\left(n; \frac{1}{2} + \frac{\theta}{4}, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{\theta}{4}\right).$$

- Estimation is easier if the x_1 cell is split into two cells, so we create the augmented model

$$(z_1, z_2, x_2, x_3, x_4) \sim \mathcal{M}\left(n; \frac{1}{2}, \frac{\theta}{4}, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{\theta}{4}\right),$$

with $x_1 = z_1 + z_2$.

Genetic Linkage

- The observed likelihood function is proportional to

$$\left(\frac{1}{2}+\frac{\theta}{4}\right)^{x_1}\left(\frac{1}{4}(1-\theta)\right)^{x_2+x_3}\left(\frac{\theta}{4}\right)^{x_4} \propto (2+\theta)^{x_1}(1-\theta)^{x_2+x_3}\theta^{x_4},$$

- and the complete-data likelihood function is

$$\left(\frac{1}{2}\right)^{z_1}\left(\frac{\theta}{4}\right)^{z_2}\left(\frac{1}{4}(1-\theta)\right)^{x_2+x_3}\left(\frac{\theta}{4}\right)^{x_4} \propto \theta^{z_2+x_4}(1-\theta)^{x_2+x_3}.$$

- The missing data density is

$$\text{missing data density} = \frac{\text{complete-data likelihood function}}{\text{observed likelihood function}}.$$

Genetic Linkage

- The observed likelihood function $\propto (2+\theta)^{x_1}(1-\theta)^{x_2+x_3}\theta^{x_4}$,
- and the complete-data likelihood function $\propto \theta^{z_2+x_4}(1-\theta)^{x_2+x_3}$.
- The missing data density is

$$\frac{\theta^{z_2+x_4}(1-\theta)^{x_2+x_3}}{(2+\theta)^{x_1}(1-\theta)^{x_2+x_3}\theta^{x_4}} \propto \left(\frac{\theta}{2+\theta}\right)^{z_2}\left(\frac{2}{2+\theta}\right)^{x_1-z_2}$$

so $Z_2 \sim$ binomial$(x_1, \frac{\theta}{2+\theta})$.

Note that x_2, x_3, x_4 cancel.

Genetic Linkage

- For the EM algorithm, the expected complete log-likelihood function is

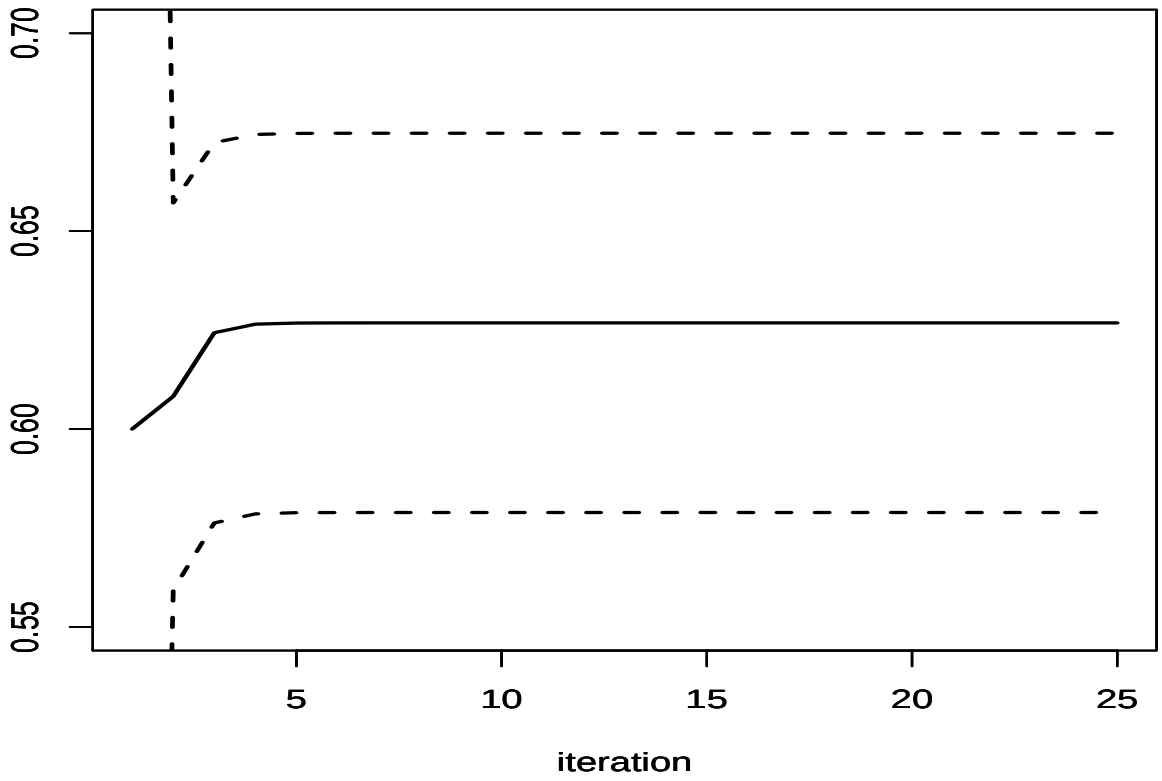
$$E_{\theta_0}[(Z_2 + x_4)\log\theta + (x_2 + x_3)\log(1-\theta)]$$
$$= \left(\frac{\theta_0}{2+\theta_0}x_1 + x_4\right)\log\theta + (x_2 + x_3)\log(1-\theta).$$

- and the EM iterates are

$$\theta_{j+1} = \text{argmax}_\theta \left[\left(\frac{\theta_j}{2+\theta_j}x_1 + x_4\right)\log\theta + (x_2 + x_3)\log(1-\theta)\right]$$
$$= \frac{\frac{\theta_j}{2+\theta_j}x_1 + x_4}{\frac{\theta_j}{2+\theta_j}x_1 + x_2 + x_3 + x_4}.$$

- R program "GeneticEM"

EM Sequence (and standard errors)



Example 5.17: EM for censored data

- For $Y_i \sim \mathcal{N}(\theta, 1)$, with censoring at a, the complete-data likelihood is

$$L^c(\theta|\mathbf{y}, \mathbf{z}) \propto \prod_{i=1}^{m} \exp\{-(y_i - \theta)^2/2)\} \prod_{i=m+1}^{n} \exp\{(z_i - \theta)^2/2\}.$$

- The density of of the missing data $\mathbf{z} = (z_{n-m+1}, \ldots, z_n)$ is a truncated normal

$$\mathbf{Z} \sim k(\mathbf{z}|\theta, \mathbf{y}) = \frac{1}{(2\pi)^{(n-m)/2}} \exp\left\{ \sum_{i=m+1}^{n} (z_i - \theta)^2/2 \right\},$$

Censored EM - continued

- Complete-data log likekihood

$$-\frac{1}{2}\sum_{i=1}^{m}(y_i-\theta)^2-\frac{1}{2}\sum_{i=n-m+1}^{n}\mathrm{E}_{\theta'}[(Z_i-\theta)^2].$$

- Differentiate and set equal to zero, solving for the EM estimate

$$\hat{\theta}=\frac{m\bar{y}+(n-m)\mathrm{E}_{\theta'}(Z_1)}{n}.$$

- Evaluate the expectation to get the EM sequence

$$\hat{\theta}^{(j+1)}=\frac{m\bar{y}+(n-m)\hat{\theta}^{(j)}+\frac{\phi(a-\hat{\theta}^{(j)})}{1-\Phi(a-\hat{\theta}^{(j)})}}{n},$$

  where ϕ and Φ are the normal pdf and cdf, respectively.

Section 5.3.3: Monte Carlo EM

- A difficulty with the implementation of the EM algorithm is that each "E-step" requires the computation of the expected log likelihood

$$\mathrm{E}_{\theta_0}(\log L^c(\theta|\mathbf{x},\mathbf{z})).$$

- To overcome this difficulty
  - simulate $Z_1,\ldots,Z_m \sim k(\mathbf{z}|\mathbf{x},\theta)$
  - maximize the approximate complete data log-likelihood

$$\hat{\mathrm{E}}_{\theta_0}(\log L^c(\theta|\mathbf{x},\mathbf{z})) = \frac{1}{m}\sum_{i=1}^{m} \log L^c(\theta|\mathbf{x},\mathbf{z}) \ .$$

Monte Carlo EM -2

- Maximize the approximate complete data log-likelihood

$$\hat{\mathrm{E}}_{\theta_0}(\log L^c(\theta|\mathbf{x},\mathbf{z})) = \frac{1}{m}\sum_{i=1}^{m} \log L^c(\theta|\mathbf{x},\mathbf{z}) \ .$$

  - When m goes to infinity, this quantity converges to $\mathrm{E}_{\theta_0}(\log L^c(\theta|\mathbf{x},\mathbf{z}))$
  - Thus, *Monte Carlo EM* $\rightarrow$ regular EM.

Genetic Linkage

- For the Monte Carlo EM algorithm, we average the complete-data log likelihood over z_2

$$\begin{aligned}
&\frac{1}{m}\sum_{i=1}^{m}\log\left[\theta^{z_{2i}+x_4}(1-\theta)^{x_2+x_3}\right]\\
=&\left(\frac{1}{m}\sum_{i=1}^{m}z_{2i}+x_4\right)\log(\theta)+(x_2+x_3)\log(1-\theta)\\
=&\ (\bar{z}_2+x_4)\log(\theta)+(x_2+x_3)\log(1-\theta),
\end{aligned}$$

where $\bar{z}_2=\frac{1}{m}\sum_{i=1}^{m}z_{2i}$, $z_{2i}\sim$ Binomial$(x_1,\theta_0/(2+\theta_0))$.

Genetic Linkage

- The Monte Carlo MLE in θ is then the Beta MLE

$$\hat{\theta} = \frac{\overline{z}_2 + x_4}{\overline{z}_2 + x_2 + x_3 + x_4}.$$

- For the EM sequence

$$\hat{\theta}^{(j+1)} = \frac{m\bar{y} + (n-m)\mathrm{E}_{\hat{\theta}^{(j)}}(Z_1)}{n},$$

- the MCEM solution replaces $\mathrm{E}_{\theta^{(j)}}(Z_1)$ with

$$\frac{1}{M}\sum_{i=1}^{M} Z_i, \quad Z_i \sim k(z|\hat{\theta}^{(j)}, \mathbf{y}).$$

Censored MCEM

- Complete-data log likekihood

$$-\frac{1}{2}\sum_{i=1}^{m}(y_i-\theta)^2-\frac{1}{2}\sum_{i=n-m+1}^{n}\mathrm{E}_{\theta'}[(Z_i-\theta)^2].$$

- Differentiate and set equal to zero, solving for the EM estimate

$$\hat{\theta}=\frac{m\bar{y}+(n-m)\mathrm{E}_{\theta'}(Z_1)}{n}.$$

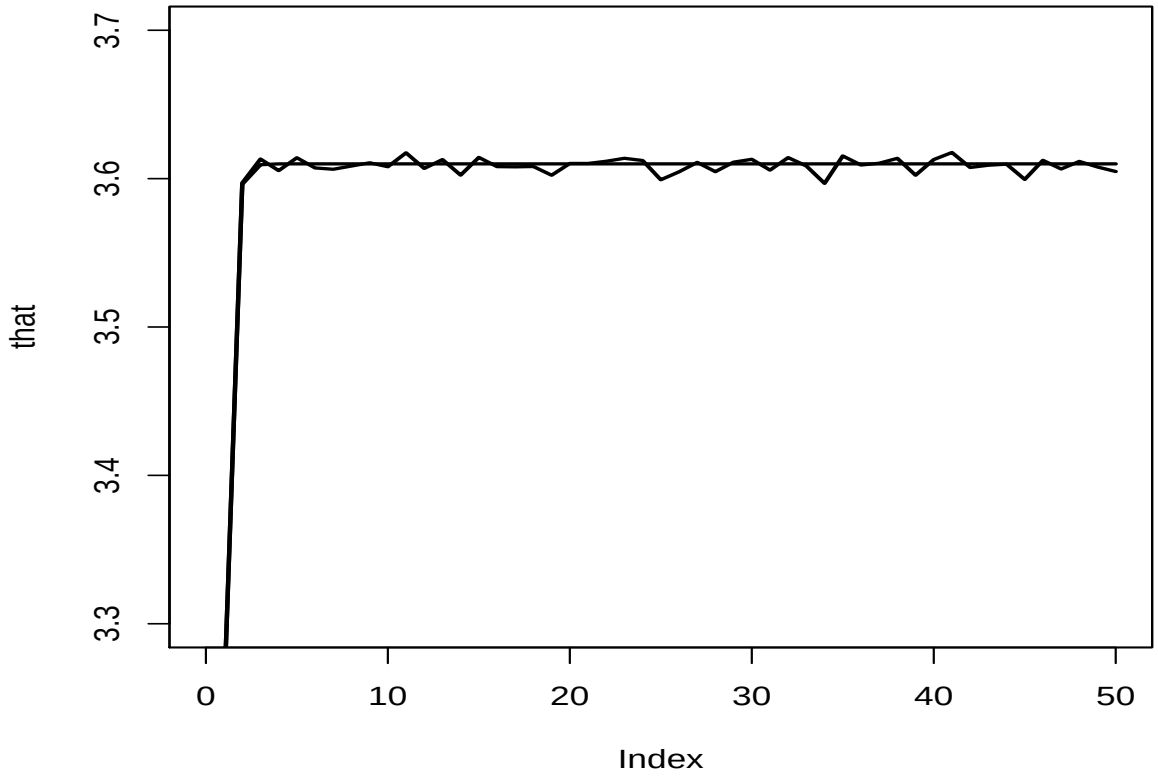
- Evaluate the expectation to get the MCEM sequence

$$\hat{\theta}^{(j+1)}=\frac{m\bar{y}+(n-m)+\bar{Z}}{n},$$

where $\bar{Z}$ is the mean of $(Z_1,\ldots,Z_M)$

$$(Z_1,\ldots,Z_M)\sim \text{ truncated normal with mean } \hat{\theta}^{(j)}$$

EM and MCEM Sequence for censored data



R program

```
xdata<-c(3.64, 2.78, 2.91,2.85,2.54,2.62,3.16,2.21,4.05,2.19,2.97,4.32,
3.56,3.39,3.59,4.13,4.21,1.68,3.88,4.33)
n<-25;m<-20;t0<-4;a<-4.5;nt<-50
xbar<-mean(xdata);that<-array(xbar,dim=c(nt,1));
for (j in 2:nt)  {
that[j] <-(m/n)*xbar+(1-m/n)*(that[j-1]+dnorm(a-that[j-1])
    /(1-pnorm(a-that[j-1])))}
#now do MCEM, z=missing data, nz=size of MC sample
tmc<-array(xbar,dim=c(nt,1));nz<-500;
for (j in 2:nt)  {
z<-array(a-1,dim=c(nz,1));
for (k in 1:nz)  {while(z[k] <a) z[k] <- rnorm(1,mean=tmc[j-1],sd=1)}
zbar<-mean(z)
tmc[j] <-(m/n)*xbar+(1-m/n)*zbar}
plot(that,type="l",xlim=c(0,nt),ylim=c(3.3,3.7),lwd=2)
par(new=T)
plot(tmc,type="l",xlim=c(0,nt),ylim=c(3.3,3.7),xlab="",
    ylab="",xaxt="n",yaxt="n",lwd=2)
```

EM Standard Errors

- Recall that the variance of the MLE, is approximated by

$$\text{Var}\,\hat{\theta} \approx \left[\frac{\partial^2}{\partial\theta^2}\text{E}\left(\log L(\theta|\mathbf{x})\right)\right]^{-1}$$

- We estimate this with

$$\text{Var}\,\hat{\theta} \approx \left[\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x})\bigg|_{\theta=\hat{\theta}}\right]^{-1}$$

- For Genetic Linkage, the observed likelihood function $\propto$

$$(2+\theta)^{x_1}(1-\theta)^{x_2+x_3}\theta^{x_4},$$

EM Standard Errors -2

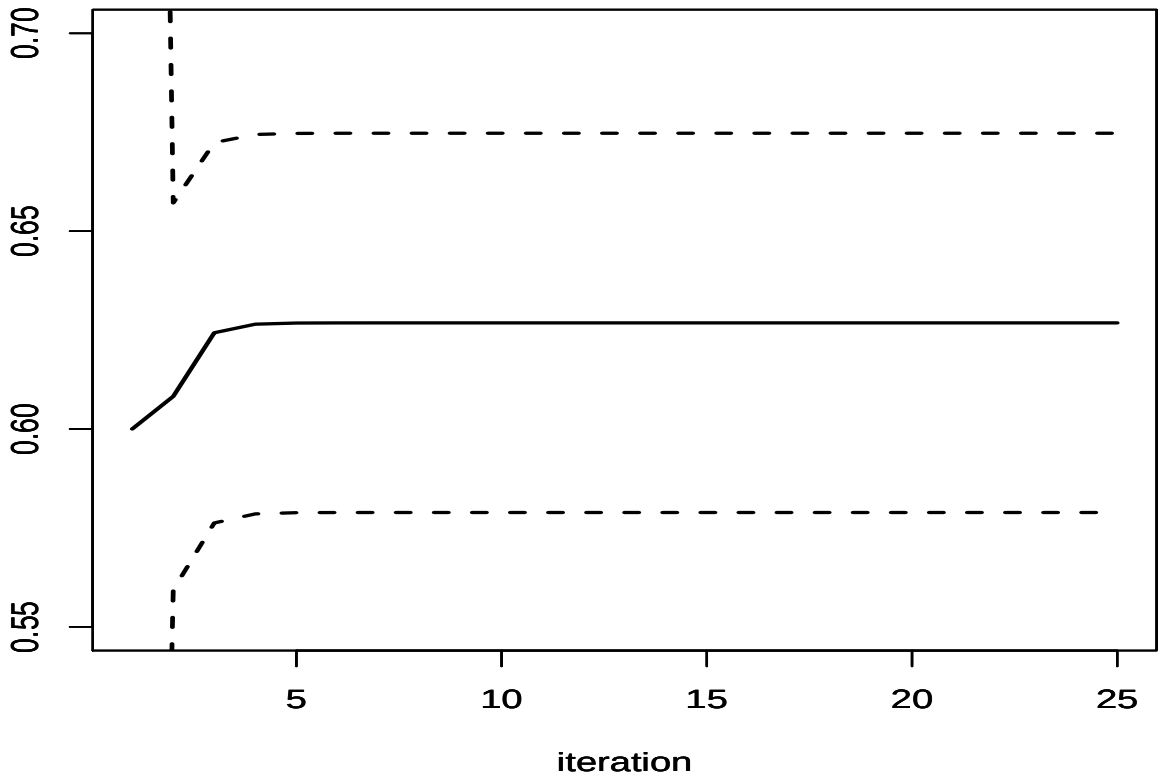
- For Genetic Linkage, the observed likelihood function $\propto$

$$(2+\theta)^{x_1}(1-\theta)^{x_2+x_3}\theta^{x_4},$$

- The variance is estimated with

$$\left[\frac{d^2}{d\theta^2}(2+\theta)^{x_1}(1-\theta)^{x_2+x_3}\theta^{x_4}\bigg|_{\theta=\hat{\theta}}\right]^{-1}$$

EM Sequence (and standard errors)



MCEM Standard Errors

- The variance of the MLE, is approximated with the observed data likelihood

$$\text{Var}\,\hat{\theta} \approx \left[\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x})\right]^{-1}$$

- Oakes (1999) expressed this wih only the complete-data likelihood

$$\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x}) = \left\{\frac{\partial^2}{\partial\theta'^2}E[\log L(\theta'|\mathbf{x},\mathbf{z})|\theta] + \frac{\partial^2}{\partial\theta'\partial\theta}E[\log L(\theta'|\mathbf{x},\mathbf{z})|\theta]\right\}\Bigg|_{\theta'=\theta}$$

  with expectation under the missing data distribution.

- This expression only involves the complete data likelihood!

- But, the expression is not good for simulation.
- With effort, we can write this as

$$\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x}) = E\left(\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x},\mathbf{z})\middle|\,\theta\right)+\text{var}\left(\frac{\partial}{\partial\theta}\log L(\theta|\mathbf{x},\mathbf{z})\middle|\,\theta\right).$$

- This allows the Monte Carlo evaluation

$$\begin{aligned}&\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x})\\ &\quad=\frac{1}{M}\sum_{j=1}^{M}\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x},\mathbf{z}^{(j)})\\ &\qquad+\frac{1}{M}\sum_{j=1}^{M}\left(\frac{\partial}{\partial\theta}\log L(\theta|\mathbf{x},\mathbf{z}^{(j)})-\frac{1}{M}\sum_{j'=1}^{M}\frac{\partial}{\partial\theta}\log L(\theta|\mathbf{x},\mathbf{z}^{(j')})\right)^2,\end{aligned}$$

where $(\mathbf{z}^{(j)})$, $j=1,\ldots,M$ are generated from the missing data distribution (and have already been generated to do MCEM).

Chapter 6: Markov Chains

- A Markov chain is a sequence of rvs that can be thought of as evolving over time
- The probability of a transition depends on the particular set the chain is in.
- We define a Markov chain by its transition kernel
  - When $\mathcal{X}$ is *discrete*, the transition kernel simply is a (transition matrix K with elements

$$P_{xy} = P(X_n = y | X_{n-1} = x) \,, \qquad x, y \in \mathcal{X}.$$

  - In the continuous case, the *kernel* also denotes the conditional density $K(x, x')$ $P(X \in A|x) = \int_A K(x, x')dx' = \int_A f(x'|x)dx'$.

Section 6.1: Essentials of MCMC

- In the setup of MCMC algorithms, Markov chains are constructed from a *transition kernel* K, a conditional probability density

$$X_{n+1} \sim K(X_n, X_{n+1}).$$

- An example is a random walk

$$X_{n+1} = X_n + \epsilon_n$$

  where ϵ_n is generated independently of $X_n, X_{n-1}, \ldots$.

- If ϵ_n is symmetric about zero, the sequence is called a *symmetric random walk*

Example 6.6: AR(1) Models

- AR(1) models provide a simple illustration of Markov chains on continuous state-space
- Here
$$X_n = \theta X_{n-1} + \varepsilon_n \ , \qquad \theta \in \Re,$$
with $\varepsilon_n \sim N(0, \sigma^2)$
- If the ε_n's are independent, X_n is independent from $X_{n-2}, X_{n-3}, \ldots$ conditionally on X_{n-1}.

Essentials of MCMC - continued

- The chains encountered in MCMC settings enjoy a very strong stability property
- The stationary distribution, or the marginal distribution always exists.
  - The stationary distribution π satisfies

$$X_n \sim \pi \Rightarrow X_{n+1} \sim \pi,$$

AR(1) Stationary Distribution

- The stationary distribution $\phi(x|\mu,\tau^2)$must satisfy

$$\int \phi(x_n|\theta x_{n-1},\sigma^2) \times \phi(x_{n-1}|\mu,\tau^2)dx_{n-1} = \phi(x_n|\mu,\tau^2)$$

- Evaluating the integral yields

$$\mathrm{E}X_n = \mu = \theta\mu \text{ and } \mathrm{Var}X_n = \tau^2 = \sigma^2 + \theta^2\tau^2$$

- Therefore

$$\mu = 0 \text{ and } \tau^2 = \frac{\sigma^2}{1-\theta^2}$$

  which requires $|\theta| < 1$.

Essentials - continued

- If the kernel allows for free moves over the entire state space, the chain is irreducible
- This also insures that the chains are positive recurrent, that is, they visit every set infinitely often.
- The stationary distribution is also a *limiting distribution* in the sense that the limiting distribution of X_{n+1} is π

Essentials - continued

- An irreducible, positive recurrent Markov chain is ergodic, that is, it converges.
- In a simulation setup, a consequence of this convergence property is that the average

$$\frac{1}{N}\sum_{n=1}^{N} h(X_n) \to \mathrm{E}_\pi[h(X)]$$

as.

- Under a slightly stronger assumption a Central Limit Theorem also holds for this average

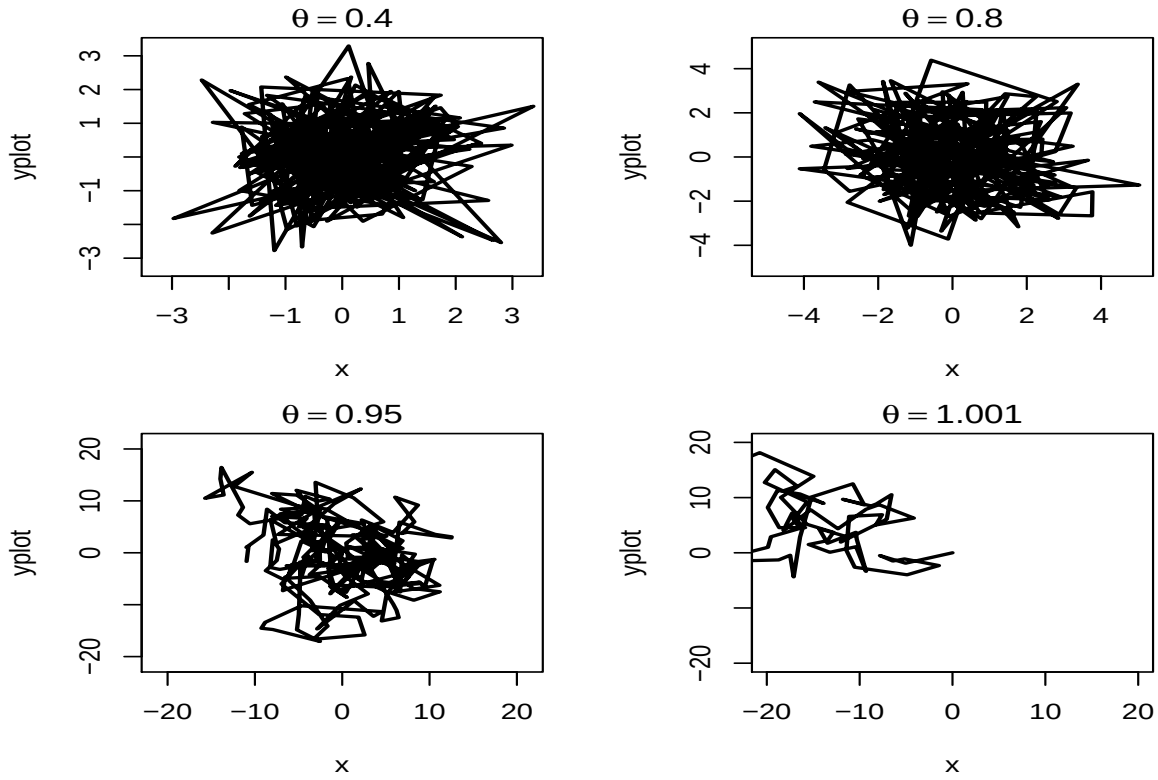
Essentials - continued

- As a final essential, we associate the probabilistic language of Markov chains with the statistical language of data analysis.

Statistics		Markov Chain
marginal distribution	$\Leftrightarrow$	invariant distribution
proper marginals	$\Leftrightarrow$	positive recurrent

- If the marginals are not proper, or if they do not exist, then the chain is not positive recurrent. It is either null recurrent or transient, and both are bad.

AR(1) Recurrent and Transient -Note the Scale



Chapter 7: The Metropolis-Hastings Algorithm

Section 7.1: The MCMC Principle

- It is not necessary to directly simulate from f to calculate

$$\int h(x)f(x)dx$$

- Now we obtain
  - $X_1, \ldots, X_n \sim$ approx f without simulating from f
  - Use an ergodic Markov Chain

Working Principle of MCMC Algorithms

- For an arbitrary starting value $x^{(0)}$, a chain $(X^{(t)})$ is generated using a transition kernel with stationary distribution f
- This ensures the convergence in distribution of $(X^{(t)})$ to a random variable from f
- Given that the chain is ergodic, the starting value $x^{(0)}$ is, in principle, unimportant.

Definition A *Markov chain Monte Carlo (MCMC) method* for the simulation of a distribution f is any method producing an ergodic Markov chain $(X^{(t)})$ whose stationary distribution is f.

Section 7.3: The Metropolis-Hastings Algorithm

- The algorithm starts with and target density f
- A candidate density $q(y|x)$
- The ratio

$$\frac{f(x)}{q(y|x)}$$

must be known up to a constant.

The Algorithm

- The Metropolis–Hastings algorithm associated with the objective (target) density f and the conditional density q produces a Markov chain $(X^{(t)})$ through the following transition:

1. Generate $Y_t \sim q(y|x^{(t)})$.
2. Take

$$X^{(t+1)} = \begin{cases} Y_t & \texttt{with probability } \rho(x^{(t)}, Y_t), \\ x^{(t)} & \texttt{with probability } 1 - \rho(x^{(t)}, Y_t), \end{cases}$$

where

$$\rho(x, y) = \min \left\{ \frac{f(y)}{f(x)} \, \frac{q(x|y)}{q(y|x)} \, , 1 \right\} \, .$$

MH Properties

- This algorithm always accepts values y_t such that the ratio $f(y_t)/q(y_t|x^{(t)})$ is increased
- It may accept values y_t such that the ratio is decreased, similar to stochastic optimization
- Like the Accept–Reject method, the Metropolis–Hastings algorithm only depends on the ratios

$$f(y_t)/f(x^{(t)}) \qquad \text{and} \qquad q(x^{(t)}|y_t)/q(y_t|x^{(t)})$$

and is, therefore, independent of normalizing constants

MH Properties - continued

- There are similarities between MH and the Accept–Reject methods
- A sample produced by MH differs from an iid sample.
  - For one thing, such a sample may involve repeated occurrences of the same value
  - Rejection of Y_t leads to repetition of $X^{(t)}$ at time $t+1$

MH Properties - continued

- It is necessary to impose minimal regularity conditions on both f and the conditional distribution q for f to be the limiting distribution of the chain $(X^{(t)})$
- ◦ The support of f should be connected
  ◦ It is better that $\sup_x f(x)/q(x|x') < \infty$

MH Convergence

- Under mild conditions, MH is a reversible, ergodic Markov Chain, hence it converges
- ◦ The empirical sums $\frac{1}{M}\sum h(X_i)$ converge
  ◦ The CLT is satisfied

Section 7.4: The Independent MH Algorithm

- the instrumental distribution q is independent of $X^{(t)}$ and is denoted g by analogy. Given $x^{(t)}$

  (a) Generate $Y_t \sim g(y)$.

  (b) Take

$$X^{(t+1)} = \begin{cases} Y_t & \texttt{with probability } \min\left\{\dfrac{f(Y_t)\, g(x^{(t)})}{f(x^{(t)})\, g(Y_t)}\, , 1\right\} \\ x^{(t)} & \texttt{otherwise.} \end{cases}$$

- Although the Y_t's are generated independently, the resulting sample is not iid, if only because the probability of acceptance of Y_t depends on $X^{(t)}$

Example 7.10: Generating Gamma Variables

- Generate $\mathcal{G}a(\alpha, \beta)$ using a Gamma $\mathcal{G}a([\alpha], b)$ candidate (where $[a]$ denotes the integer part of a).
- Take $\beta = 1$

1. Generate $Y_t \sim \mathcal{G}a([\alpha], [\alpha]/\alpha)$.
2. Take

$$X^{(t+1)} = \begin{cases} Y_t & \texttt{with probability } \varrho_t \\ x^{(t)} & \texttt{otherwise,} \end{cases}$$

where

$$\varrho_t = \min \left[\left(\frac{Y_t}{x^{(t)}} \exp \left\{ \frac{x^{(t)} - Y_t}{\alpha} \right\} \right)^{\alpha - [\alpha]}, 1 \right] .$$

Example 7.11: Logistic Regression

- Return to the Challenger Data
- We observe (x_i, y_i), $i = 1, \ldots, n$ according to the model

$$Y_i \sim \text{Bernoulli}(p(x_i)), \quad p(x) = \frac{\exp(\alpha + \beta x)}{1 + \exp(\alpha + \beta x)},$$

where $p(x)$ is the probability of an O-ring failure at temperature x.

  - The likelihood is

$$L(\alpha, \beta|\mathbf{y}) \propto \prod_{i=1}^{n} \left(\frac{\exp(\alpha + \beta x_i)}{1 + \exp(\alpha + \beta x_i)} \right)^{y_i} \left(\frac{1}{1 + \exp(\alpha + \beta x_i)} \right)^{1-y_i}$$

  - and we take the prior to be

$$\pi_\alpha(\alpha|b)\pi_\beta(\beta) = \frac{1}{b} e^{\alpha} e^{-e^{\alpha}/b} d\alpha d\beta,$$

Logistic Regression - continued

- The prior

$$\pi_\alpha(\alpha|b)\pi_\beta(\beta) = \frac{1}{b}e^{\alpha}e^{-e^{\alpha}/b}d\alpha d\beta,$$

  - puts an exponential prior on $\log \alpha$
  - a flat prior on β
  - insures propriety of the posterior distribution

- Choose b so that $\mathrm{E}\alpha = \hat{\alpha}$, where $\hat{\alpha}$ is the MLE of α

Logistic Regression - continued

- The posterior distribution is proportional to $L(\alpha, \beta|\mathbf{y})\pi(\alpha, \beta)$
- To simulate from this distribution we take an independent candidate

$$g(\alpha, \beta) = \pi_\alpha(\alpha|\hat{b})\phi(\beta),$$

where $\phi(\beta)$ is a normal distribution with mean $\hat{\beta}$ and variance $\hat{\sigma}^2_\beta$, the MLEs.

  - Note that although basing the prior distribution on the data is somewhat in violation of the formal Bayesian paradigm, nothing is violated if the candidate depends on the data.
  - In fact, this will usually result in a more effective simulation, as the candidate is placed close to the target.

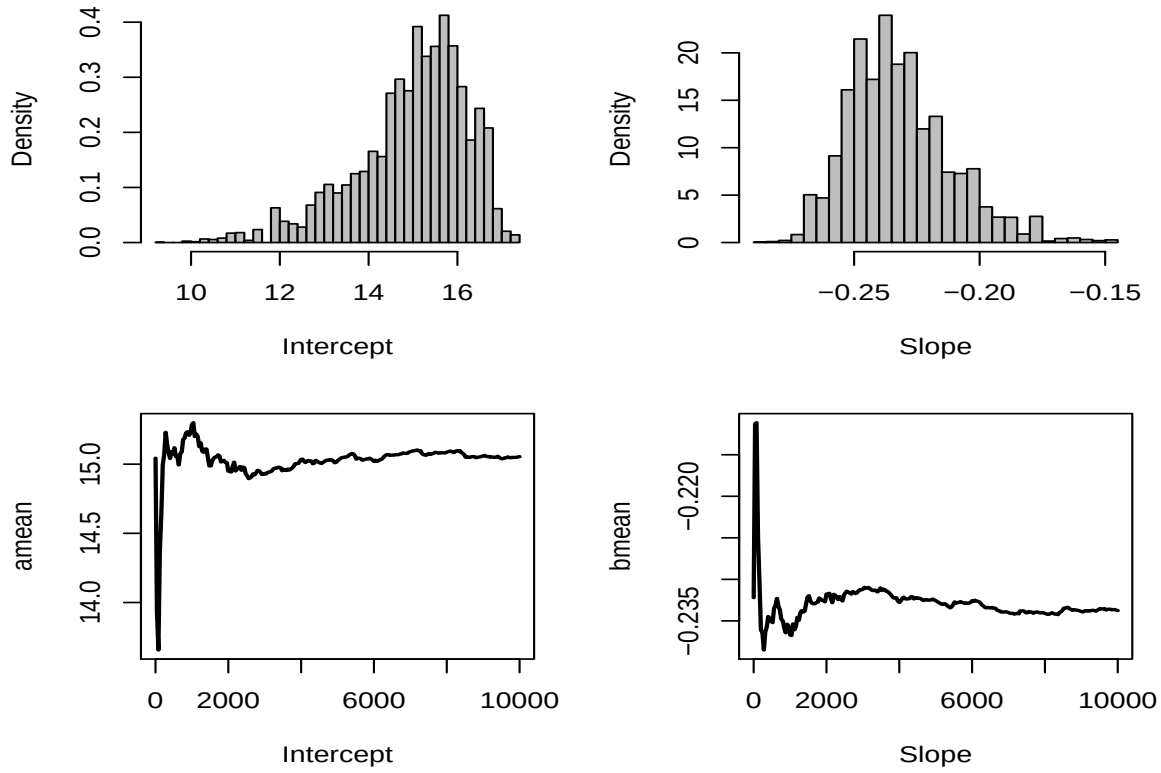
Logistic Regression - continued

- Generating a random variable from $g(\alpha, \beta)$ is straightforward
- If we are at the point (α_0, β_0) in the Markov chain, and we generate (α', β') from $g(\alpha, \beta)$, we accept the candidate with probability

$$\min \left\{ \frac{L(\alpha', \beta'|\mathbf{y})}{L(\alpha_0, \beta_0|\mathbf{y})} \frac{\phi(\beta_0)}{\phi(\beta')}, 1 \right\}.$$

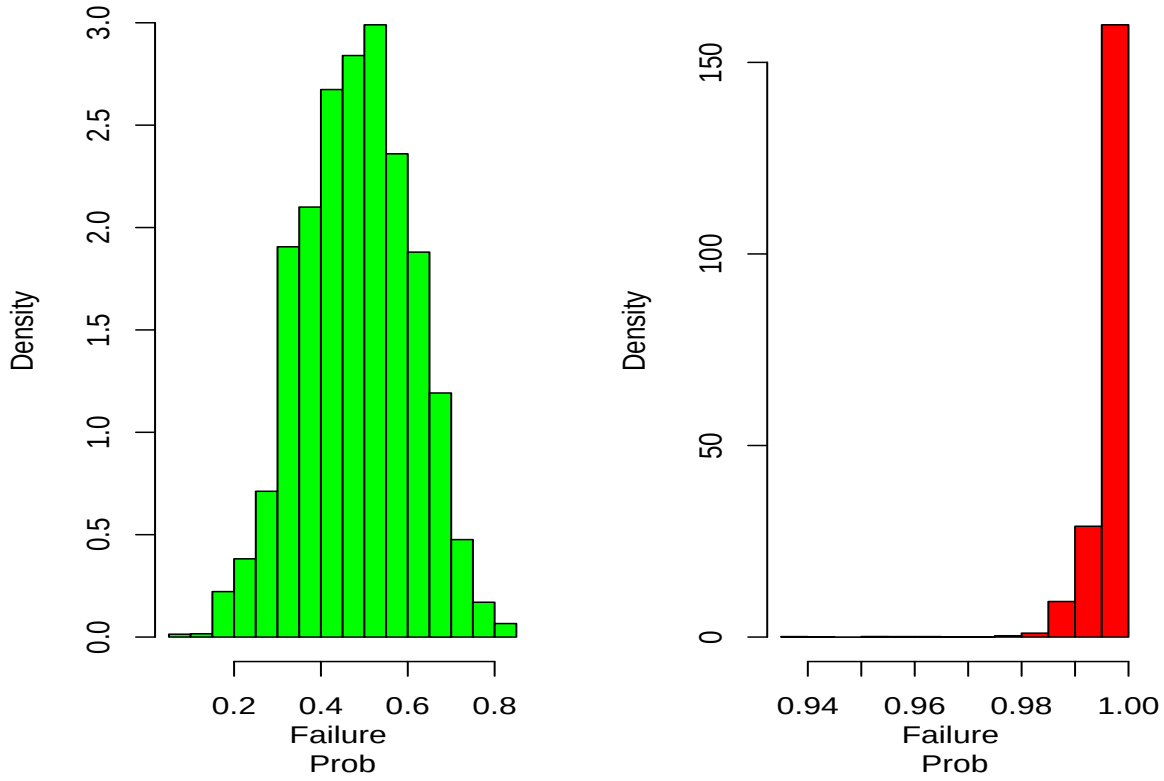
Logistic Regression - continued

- Estimation of the slope and intercept from the Challenger logistic regression. The top panels show histograms of the distribution of the coefficients, while the bottom panels show the convergence of the means.



Logistic Regression - continued

- Estimation of the failure probabilities from the Challenger logistic regression. The left panel is for 65^o Fahrenheit and the right panel is for 40^o.
- We can run the programs



Section 7.5 Random Walk Metropolis

- Take into account the value previously simulated to generate the following value
- This idea is already used in algorithms such as the simulated annealing
  - Since the candidate g in the MH algorithm is allowed to depend on the current state $X^{(t)}$, a first choice to consider is to simulate Y_t according to

    $$Y_t = X^{(t)} + \varepsilon_t,$$

    where ε_t is a random perturbation with distribution g, independent of $X^{(t)}$.
  - $q(y|x)$ is now of the form $g(y - x)$
  - The Markov chain associated with q is a random walk

Random Walk Metropolis - continued

- The choice of a *symmetric function* g (that is, such that $g(-t) = g(t)$), leads to the following random walk MH algorithm
  Given $x^{(t)}$,

  (a) Generate $Y_t \sim g(|y - x^{(t)}|)$.

  (b) Take

$$X^{(t+1)} = \begin{cases} Y_t & \text{with probability } \min\left\{1, \dfrac{f(Y_t)}{f(x^{(t)})}\right\} \\ x^{(t)} & \text{otherwise.} \end{cases}$$

Random Walk Metropolis - continued

- Hastings (1970) considers the generation of the normal distribution $\mathcal{N}(0,1)$ based on the uniform distribution on $[-\delta, \delta]$

- The algorithm: At time t

(a) Generate $Y = X_t + U$

(b)

$$\rho = \min\left\{e^{-.5(Y^2 - X_t^2)}, 1\right\}$$

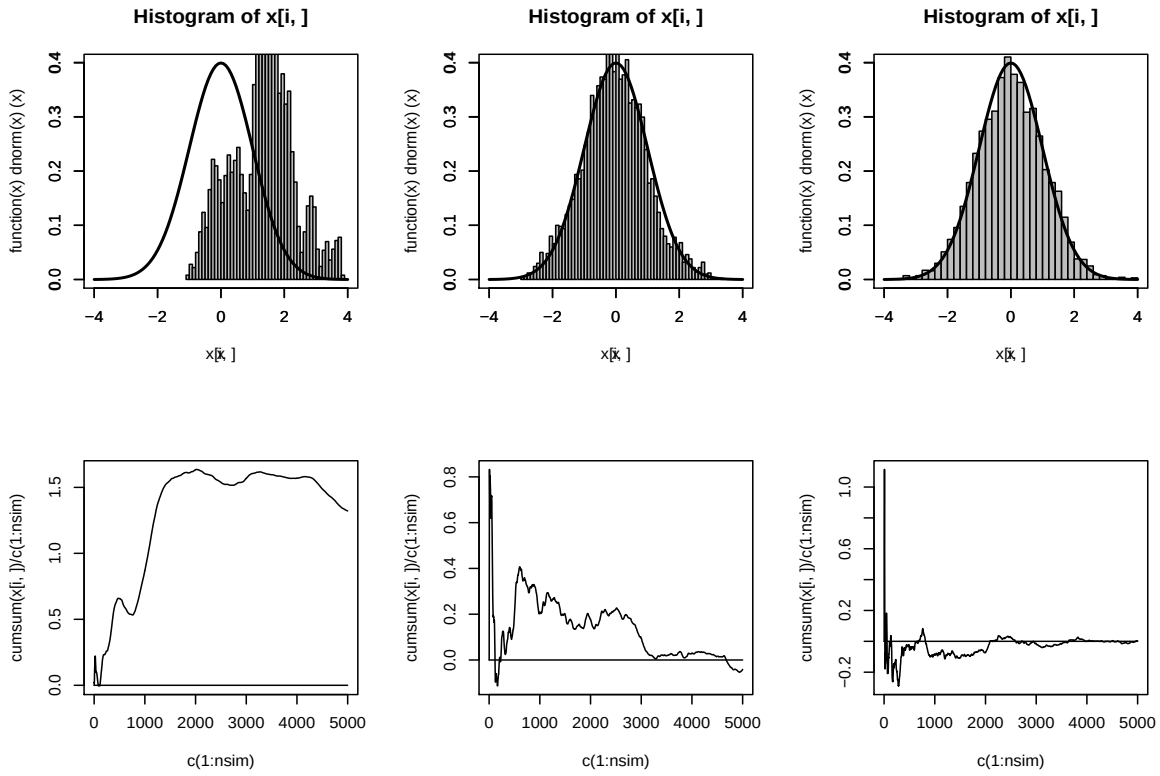
(c)

$$X_{t+1} = \begin{cases} Y & \text{with probability } \rho \\ X_t & \text{otherwise} \end{cases}$$

- Three samples of $20,000$ points produced by this method for $\delta = 0.1, 0.5$, and 1.

- R program "Hastings"

Random Walk Metropolis - continued



- Note the convergence for larger ranges
- R program "RandomWalkMet"

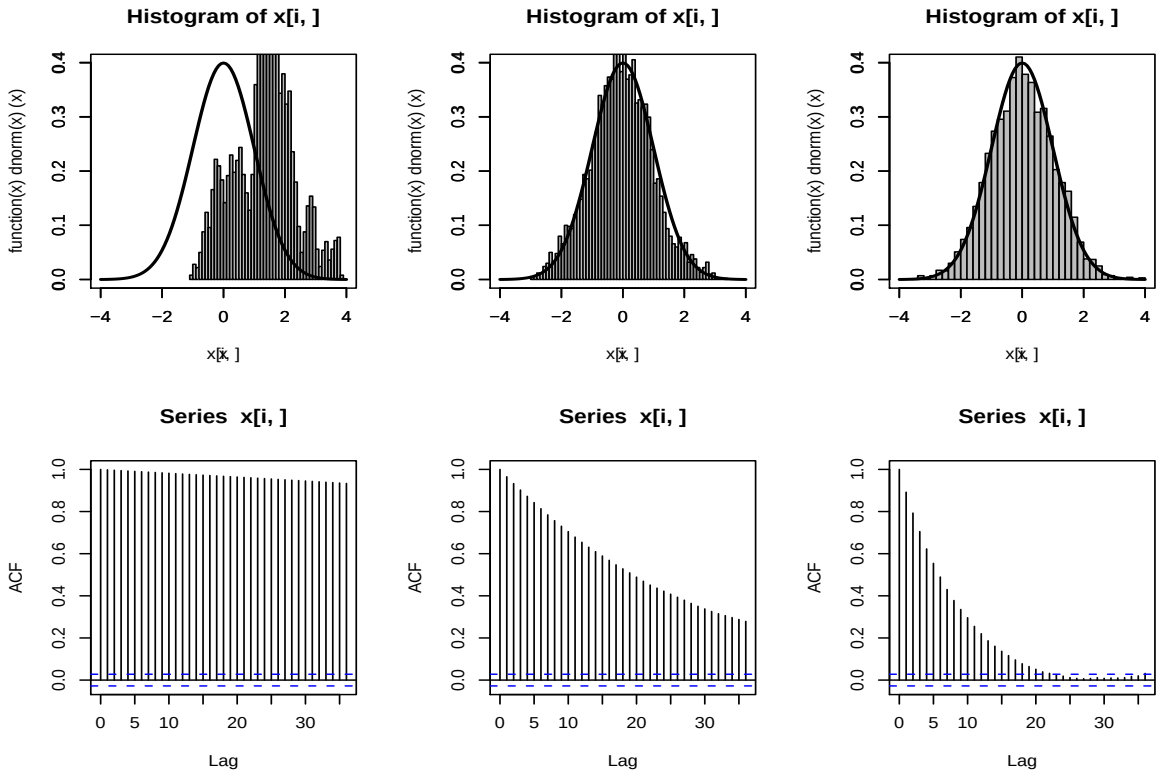
Random Walk Metropolis - 3

- Explaining the behavior
- The Random Walk

$$Y = X_t + U, \qquad U \sim \mathcal{U}(-\delta, \delta)$$

  has high autocorrelation for small δ
- High Autocorrelation $\longrightarrow$ Poor Mixing
- Look at Autocorrelation for $\delta = 0.1, 0.5$, and 1.
- `R` program "RandomWalkMetAC"

Random Walk Metropolis - 4



- Smaller Autocorrelation for larger ranges
- R program "RandomWalkMetAC"

Chapter 9: The Two Stage Gibbs Sampler

- The implementation of the two-stage Gibbs sampler is straightforward.
- Suppose that the rvs X and Y have joint density $f(x, y)$
- The two-stage Gibbs sampler generates a Markov chain (X_t, Y_t) according to the following steps:

  Take $X_0 = x_0$
  For $t = 1, 2, \ldots,$ generate

  1. $Y_t \sim f_{Y|X}(\cdot|x_{t-1})$;
  2. $X_t \sim f_{X|Y}(\cdot|y_t)$.

  where $f_{Y|X}$ and $f_{X|Y}$ are the conditional distributions associated with f

  - Then $(X_t, Y_t) \to (X, Y) \sim f(x, y)$
  - $X_t \to X \sim f(x)$
  - $Y_t \to Y \sim f(y)$

Example 9.1: Normal Bivariate Gibbs

- For the special case of the bivariate normal density,

$$(X, Y) \sim \mathcal{N}_2 \left(0, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right) ,$$

- The Gibbs sampler is

  Given y_t, generate

$$\begin{aligned} X_{t+1} \mid y_t &\sim \mathcal{N}(\rho y_t,\ 1-\rho^2) , \\ Y_{t+1} \mid x_{t+1} &\sim \mathcal{N}(\rho x_{t+1},\ 1-\rho^2). \end{aligned}$$

  - The Gibbs sampler is obviously not necessary in this particular case
  - The marginal Markov chain in X is defined by the AR(1) relation

$$X_{t+1} = \rho^2 X_t + \sigma\epsilon_t, \qquad \epsilon_t \sim \mathcal{N}(0,1) ,$$

    with $\sigma^2 = 1 - \rho^2 + \rho^2(1-\rho^2) = 1-\rho^4$.

  - The stationary distribution of this chain is $\mathcal{N}\left(0, \frac{1-\rho^4}{1-\rho^4}\right)$.

Gibbs Sampler: Missing Data

- Gibbs works well in missing data models
- We start with a marginal density $f_X(x)$ and construct (or *complete*) a joint density to aid in simulation
- Like the case of the EM algorithm
  - In missing data models we write

$$g(x|\theta) = \int_{\mathcal{Z}} f(x, z|\theta)\, dz$$

  - Which results in the Gibbs sampler

$$\theta \sim \frac{f(x, z|\theta)}{\int_\Theta f(x, z|\theta)}$$

$$Z \sim \frac{f(x, z|\theta)}{\int_{\mathcal{Z}} f(x, z|\theta)}$$

Example 9.7: Grouped counting data

- For 360 consecutive time units, consider recording the number of passages of individuals, per unit time, past some sensor.
  - the number of cars observed at a crossroad
  - number of leucocytes in a region of a blood sample
- Hypothetical results are

Number of passages	0	1	2	3	4 or more
Number of observations	139	128	55	25	13

Poisson Bayes completion

- Assume Poisson $\mathcal{P}(\lambda)$ model
- The observed data likelihood is

$$\ell(\lambda|x_1,\ldots,x_5) \propto e^{-347\lambda}\lambda^{128+55\times 2+25\times 3}\left(1-e^{-\lambda}\sum_{i=0}^{3}\frac{\lambda^i}{i!}\right)^{13},$$

  for $x_1 = 139, \ldots, x_5 = 13$.

- Complete the data with

$$\mathbf{z} = (z_1,\ldots,z_{13})$$

Poisson Bayes completion

- Start with $\mathbf{x}$ = observed data, $\mathbf{z}$ = missing data, then

$$\mathbf{X}|\lambda \sim \prod_{i=1}^{347} \frac{e^{-\lambda}\lambda^{x_i}}{x_i!}$$

$$\mathbf{Z}|, \lambda, \mathbf{x} \sim \prod_{i=1}^{13} \frac{e^{-\lambda}\lambda^{z_i}}{z_i!} I(z_i \geq 4).$$

- The joint distribution is

$$\frac{e^{-360\lambda}\lambda^{\sum_i x_i + \sum_i z_i}}{\prod_i x_i! \prod_i z_i!} \prod_i I(z_i \geq 4)$$

Poisson Bayes completion

- For $\pi(\lambda) \propto 1/\lambda$, the full conditionals are

$$\begin{aligned} \mathbf{Z}|\lambda, \mathbf{x} &\sim \text{Truncated Poisson}(\lambda) \\ \lambda|\mathbf{z}, \mathbf{x} &\sim \text{Gamma}(\sum_i x_i + \sum_i z_i + 1, 1/360) \end{aligned}$$

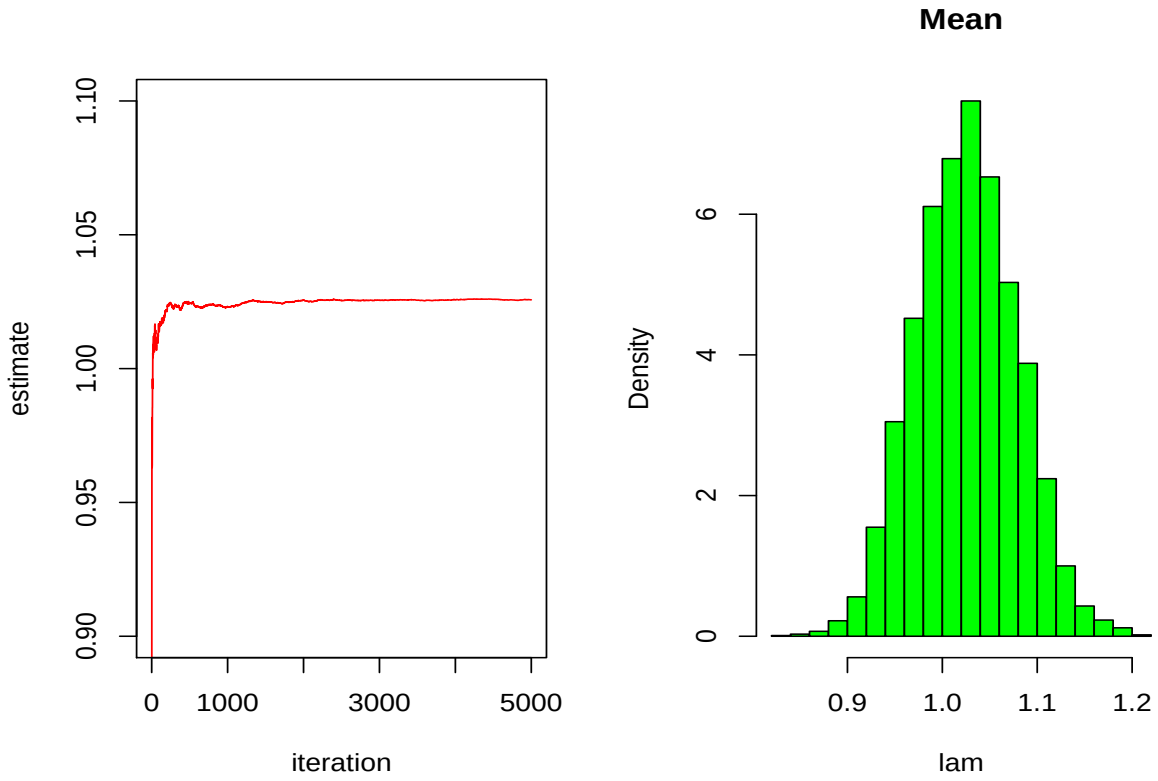
- A Gibbs sampler in λ and $\mathbf{z}$ can do the calculations

- Given $\lambda^{(t-1)}$,

1. `Simulate` $Y_i^{(t)} \sim \mathcal{P}(\lambda^{(t-1)})\ \mathrm{I}_{y \geq 4}\ (i = 1, \ldots, 13)$
2. `Simulate`

$$\lambda^{(t)} \sim \mathcal{G}a\left(313 + \sum_{i=1}^{13} y_i^{(t)},\ 360\right) .$$

Poisson Bayes Gibbs Sampler Output



Two Estimators of Lambda

- Output from the Gibbs Sampler

$$\begin{aligned} \mathbf{Z}|\lambda, \mathbf{x} &\sim \text{Truncated Poisson}(\lambda) \\ \lambda|\mathbf{z}, \mathbf{x} &\sim \text{Gamma}(\sum_i x_i + \sum_i z_i + 1, 1/360) \end{aligned}$$

- Estimate λ with the Empirical Average,

$$\frac{1}{M}\sum_{j=1}^{M} \lambda^{(j)}$$

- or the Conditional Expectation

$$\delta_{rb} = \frac{1}{M}\sum_{j=1}^{M} \mathrm{E}\left[\lambda\middle|\mathbf{x}, \mathbf{z}^{(j)}\right] = \frac{1}{360M}\sum_{j=1}^{M}\left(313 + \sum_{i=1}^{13} z_i^{(j)}\right),$$

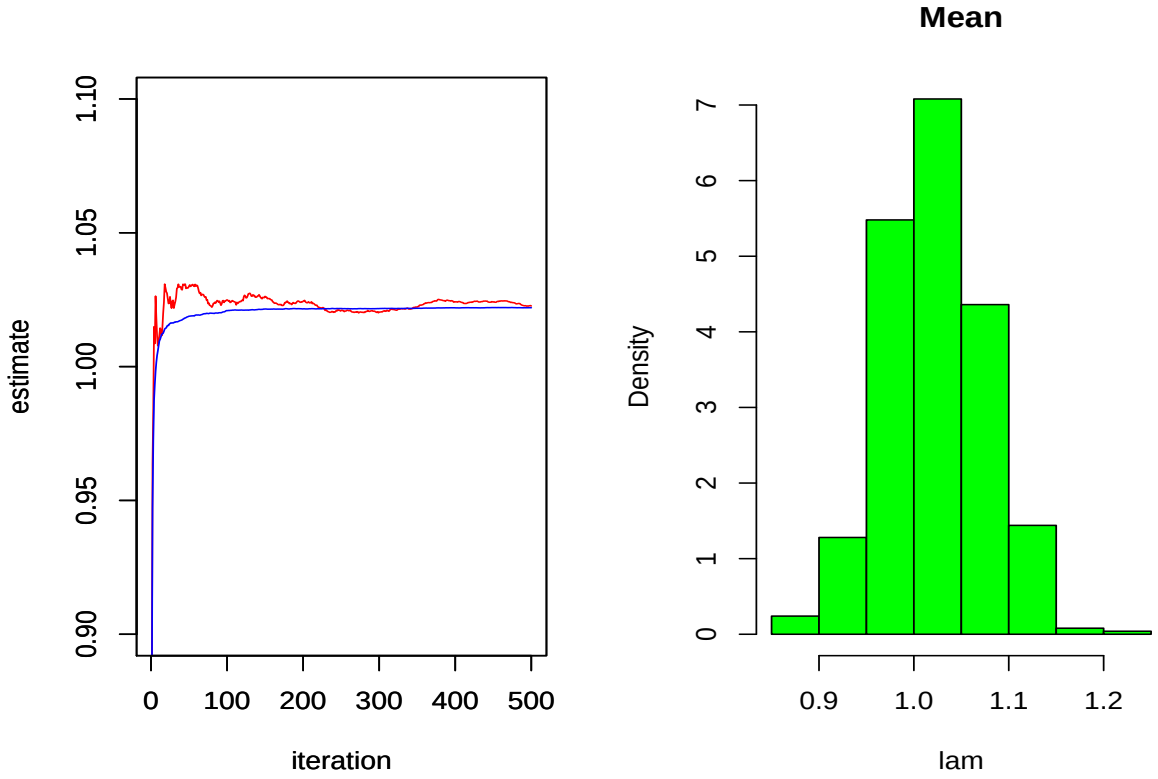
Two Estimators of Lambda -2

- or the Conditional Expectation

$$\delta_{rb} = \frac{1}{M}\sum_{j=1}^{M} \mathrm{E}\left[\lambda \middle| \mathbf{x}, \mathbf{z}^{(j)}\right] = \frac{1}{360M}\sum_{j=1}^{M}\left(313 + \sum_{i=1}^{13} z_i^{(j)}\right),$$

  - "Rao-Blackwellized"
  - Typically Smoother
- Convergence Diagnostic $\longrightarrow$ Both estimators converge
- `R` program "PoissonCompletion2"
- See `R` program "PoissonCompletion3" to eliminate "`while`"

Poisson Gibbs Sampler - Convergence of Estimators



Poisson EM Algorithm

- There is a corresponding EM algorithm: For the observed data likelihood

$$L(\lambda|x_1,\ldots,x_5) \propto e^{-347\lambda}\lambda^{313}\left(1-e^{-\lambda}\sum_{i=0}^{3}\frac{\lambda^i}{i!}\right)^{13},$$

- We have the complete data likelihood

$$L(\lambda|x_1,\ldots,x_5,\mathbf{z}) \propto e^{-347\lambda}\lambda^{313}\left(e^{-13\lambda}\prod_{i=1}^{13}\frac{\lambda^{z_i}}{z_i!}\right),$$

- With expected log likelihood

$$\log L \propto -360\lambda + (313 + E[\sum_i z_i])\log\lambda$$

Poisson EM Algorithm

- from the expected log likelihood

$$\log \ell \propto -360\lambda + (313 + E[\sum_i z_i]) \log \lambda$$

- We get the Monte Carlo EM iteration

$$\lambda^{(t+1)} = \frac{1}{360}(313 + 13E_{\lambda^{(t)}}[Z_i])$$

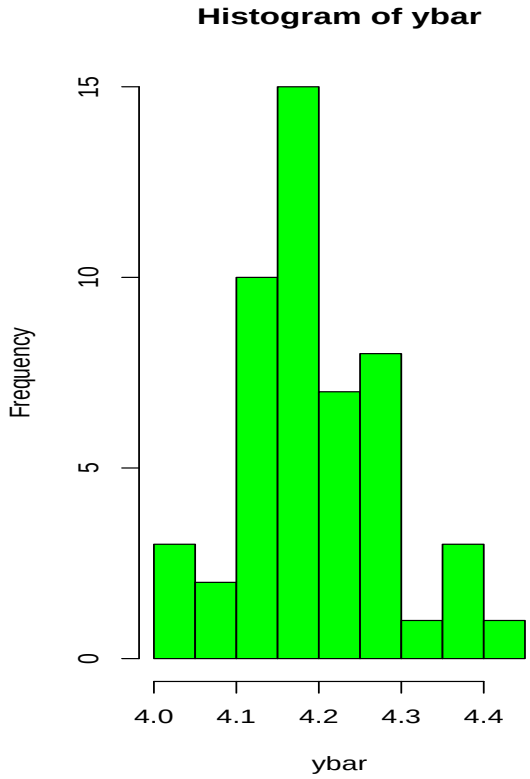
  - where

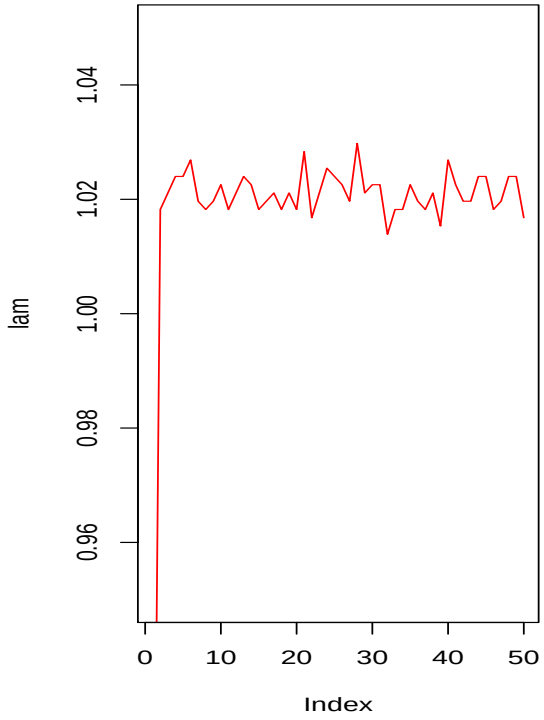
$$Z_i \sim \mathcal{P}(\lambda^{(t)}).$$

Poisson EM Algorithm R code

```
nsim<-50;lam<-array(313/360,dim=c(nsim,1));ybar<-array(4,dim=c(nsim,1))
#Use m values for the mean
m<-25;y<-array(0,dim=c(m,1));
for (j in 2:nsim)  {
for(i in 1:m){while(y[i] < 4) y[i] <- rpois(1,lam[j -1])};
ybar[j]<-mean(y);
lam[j]<-(313+13*ybar[j])/360;
y<-y*0;
}
par(mfrow=c(1,2))
hist(ybar,col="green",breaks=10)
plot(lam,col="red",type="l",ylim=c(.95,1.05))
```

Poisson EM Output





Section 9.4: The EM–Gibbs Connection

- The is a General EM/Gibbs relationship
- $\mathbf{X} \sim g(x|\theta)$ is the observed data
- $\mathbf{Z} \sim f(\mathbf{x}, \mathbf{z}|\theta)$ is the augmented data
- We have the complete-data and incomplete-data likelihoods

$$L^c(\theta|\mathbf{x}, \mathbf{z}) = f(\mathbf{x}, \mathbf{z}|\theta) \text{ and } L(\theta|\mathbf{x}) = g(\mathbf{x}|\theta)\,,$$

with the missing data density

$$k(\mathbf{z}|\mathbf{x}, \theta) = \frac{L^c(\theta|\mathbf{x}, \mathbf{z})}{L(\theta|\mathbf{x})}.$$

The EM–Gibbs Connection

- If we can normalize the complete-data likelihood in θ
- That is, if $\int L^c(\theta|\mathbf{x},\mathbf{z})d\theta < \infty$
- Define

$$L^*(\theta|\mathbf{x},\mathbf{z}) = \frac{L^c(\theta|\mathbf{x},\mathbf{z})}{\int L^c(\theta|\mathbf{x},\mathbf{z})d\theta}$$

  and create the two-stage Gibbs sampler:

$$\begin{aligned} 1.&\quad \mathbf{z}|\theta \sim k(\mathbf{z}|\mathbf{x},\theta) \\ 2.&\quad \theta|\mathbf{z} \sim L^*(\theta|\mathbf{x},\mathbf{z}) \end{aligned}$$

- Note the direct connection to an EM algorithm based on L^c and k.

Remember Genetic Linkage

- The observed likelihood function is proportional to

$$\left(\frac{1}{2}+\frac{\theta}{4}\right)^{x_1}\left(\frac{1}{4}(1-\theta)\right)^{x_2+x_3}\left(\frac{\theta}{4}\right)^{x_4} \propto (2+\theta)^{x_1}(1-\theta)^{x_2+x_3}\theta^{x_4},$$

- and the complete-data likelihood function is

$$\left(\frac{1}{2}\right)^{z_1}\left(\frac{\theta}{4}\right)^{z_2}\left(\frac{1}{4}(1-\theta)\right)^{x_2+x_3}\left(\frac{\theta}{4}\right)^{x_4} \propto \theta^{z_2+x_4}(1-\theta)^{x_2+x_3}.$$

- The missing data density is

$$\text{missing data density} = \frac{\text{complete-data likelihood function}}{\text{observed likelihood function}}.$$

Genetic Linkage

- To Gibbs sample this (with a uniform prior on θ) use

$$\theta|\mathbf{x}, z_2 \propto \theta^{z_2+x_4}(1-\theta)^{x_2+x_3} = \text{Beta}(z_2+x_4+1, x_2+x_3+1)$$

$$z_2|\mathbf{x}, \theta \propto \theta^{z_2+x_4}(1-\theta)^{x_2+x_3} = \text{Binomial}(x_1, \frac{\theta}{2+\theta})$$

Example 9.21: Censored Data Gibbs

- For the censored data example, the distribution of the missing data is

$$Z_i \sim \frac{\phi(z-\theta)}{1-\Phi(a-\theta)}$$

and the distribution of $\theta|x,z$ is

$$L(\theta|x,z) \propto \prod_{i=1}^{m} e^{-(x_i-\theta)^2/2} \prod_{i=m+1}^{n} e^{-(z_i-\theta)^2/2},$$

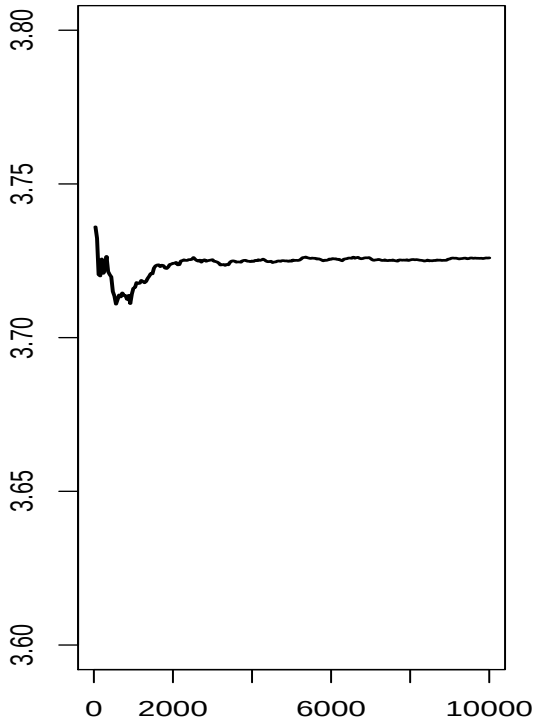
which corresponds to a

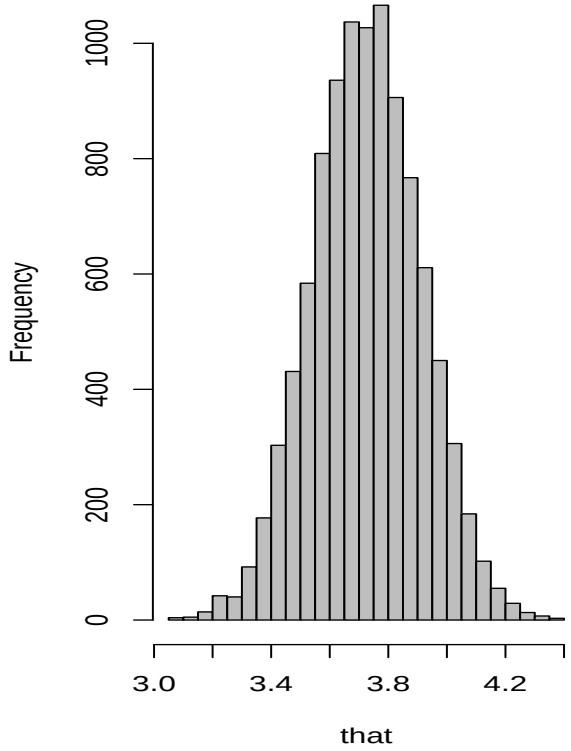
$$\mathcal{N}\left(\frac{m\bar{x}+(n-m)\bar{z}}{n}, \frac{1}{n}\right)$$

distribution and so we immediately have that L^* exists and that we can run a Gibbs sampler

- R program $\longrightarrow$ `censoredGibbs`
  - Generate Z with Accept-Reject

Example 9.21: Censored Data Gibbs





Examples 5.18 and 9.22: Cellular Phone Plans

- It is typical for cellular phone companies to offer "plans" of options, bundling together four or five options
- One cellular company had offered a four-option plan in some areas, and a five-option plan (which included the four, plus one more) in another area
- In each area, customers were ask to choose their favorite option, and the results were tabulated. In some areas they choose their favorite from four plans, and in some areas from five plans.
- The phone company is interested in knowing which are the popular plans, to help them set future prices.

Cellular Phone: The Data

- Cellular phone plan preferences in 37 areas: Data are number of customers who choose the particular plan as their favorite.

	Plan						Plan				
	1	2	3	4	5		1	2	3	4	5
1	26	63	20	0	–	20	56	18	29	5	–
2	31	14	16	51	–	21	27	53	10	0	–
3	41	28	34	10	–	22	47	29	4	11	–
4	27	29	19	25	–	23	43	66	6	1	–
5	26	48	41	0	–	24	14	30	23	23	6
6	30	45	12	14	–	25	4	24	24	32	7
7	53	39	12	11	–	26	11	30	22	23	8
⋮	⋮	⋮	⋮	⋮	⋮						

Cellular Phones: EM - 1

- We can model the complete data as follows. In area i, there are n_i customers, each of whom chooses their favorite plan from Plans $1-5$.
  - The observation for customer i is

$$Y_i = (Y_{i1}, \ldots, Y_{i5}), \text{ where } Y_i \sim \mathcal{M}(1, (p_1, p_2, \ldots, p_5)).$$

  - If we assume the customers are independent, in area i the data are

$$T_i = (T_{i1}, \ldots, T_{i5}) = \sum_{j=1}^{n_i} Y_i \sim \mathcal{M}(n_i, (p_1, p_2, \ldots, p_5))$$

Cellular Phones: EM - 2

- If the first m observations have the Y_{i5} missing, denote the missing data by z_i and then we have the complete data likelihood

$$L(\mathbf{p}|\mathbf{T},\mathbf{z}) = \prod_{i=1}^{m} \binom{n_i + z_i}{T_{i1},\ldots,T_{i4},z_i} p_1^{T_{i1}} \cdots p_4^{T_{i4}} p_5^{z_i} \times \prod_{i=m+1}^{n} \binom{n_i}{T_{i1},\ldots,T_{i5}} \prod_{j=1}^{5} p_j^{T_{ij}}$$

  where

  - $\mathbf{p} = (p_1, p_2, \ldots, p_5)$,
  - $\mathbf{T} = (T_1, T_2, \ldots, T_5)$,
  - $\mathbf{z} = (z_1, z_2, \ldots, z_m)$, and
  - $\binom{n}{n_1,n_2,\ldots,n_k}$ is the multinomial coefficient $\frac{n!}{n_1!n_2!\cdots n_k!}$.

Cellular Phones: EM - 3

- The observed data likelihood can be calculated as

$$L(\mathbf{p}|\mathbf{T}) = \sum_{\mathbf{z}} L(\mathbf{p}|\mathbf{T}, \mathbf{z})$$

leading to the missing data distribution

$$k(\mathbf{z}|\mathbf{T}, \mathbf{p}) = \prod_{i=1}^{m} \binom{n_i + z_i}{z_i} p_5^{z_i} (1 - p_5)^{n_i+1},$$

a product of negative binomial distributions.

Cellular Phones: EM - 4

- Define
  - $W_j = \sum_{i=1}^{n} T_{ij}$ for $j = 1, \ldots, 4$, and
  - $W_5 = \sum_{i=1}^{m} T_{i5}$ for $j = 5$.

- The expected complete data log likelihood is

$$\sum_{j=1}^{4} W_j \log p_j + [W_5 + \sum_{i=1}^{m} \mathrm{E}(Z_i|\mathbf{p}')] \log(1 - p_1 - p_2 - p_3 - p_4).$$

Cellular Phones: EM -5

- The expected complete data log likelihood is

$$\sum_{j=1}^{4} W_j \log p_j + [W_5 + \sum_{i=1}^{m} \mathrm{E}(Z_i|\mathbf{p}')] \log(1 - p_1 - p_2 - p_3 - p_4).$$

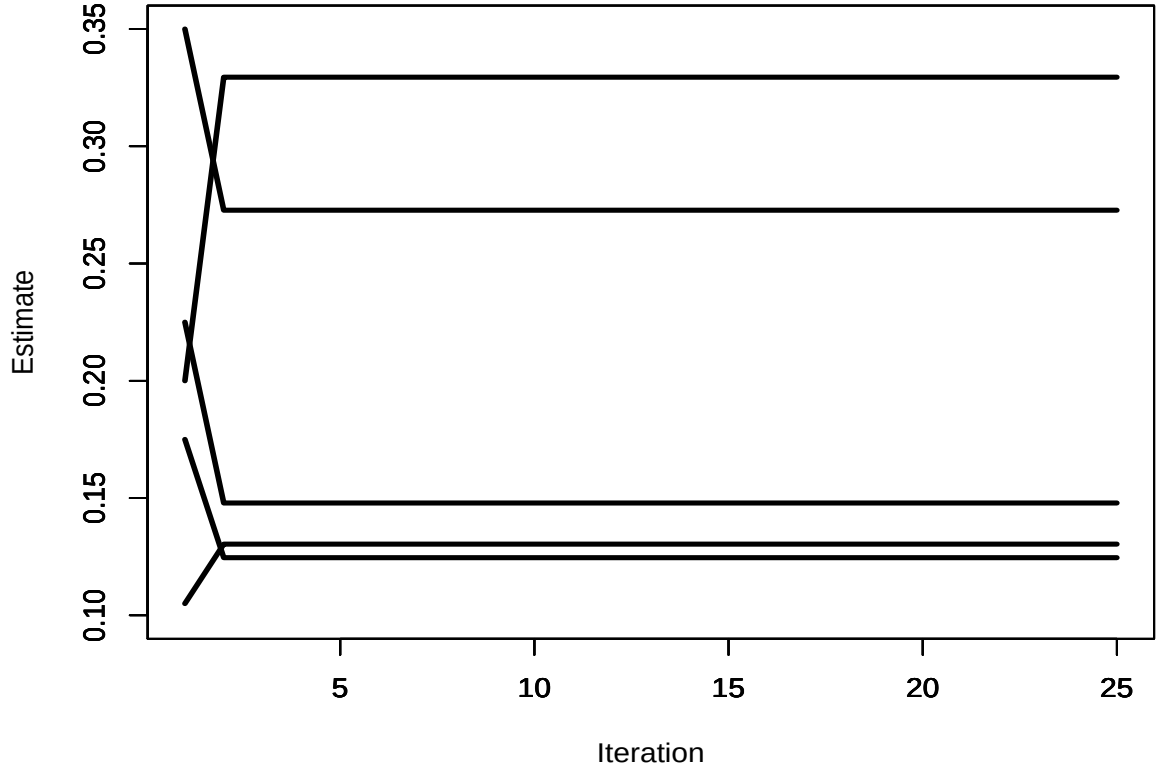
- leading to the EM iterations

$$\mathrm{E}(Z_i|\mathbf{p}^{(t)}) = (n_i+1)\frac{\hat{p}_5^{(t)}}{1 - \hat{p}_5^{(t)}}, \quad \hat{p}_j^{(t+1)} = \frac{W_j}{\sum_{i=1}^{m} \mathrm{E}(Z_i|\mathbf{p}^{(t)}) + \sum_{j'=1}^{5} W_{j'}}$$

for $j = 1, \ldots, 4$.

Cellular Phones: EM

- The MLE of $\mathbf{p}$ is $(0.273, 0.329, 0.148, 0.125, 0.125)$; convergence is very rapid.
- EM sequence for cellular phone data, 25 iterations



Cellular phone Gibbs

- Now we use the Gibbs sampler to get our solution. From the complete data likelihood and the missing data distribution we have

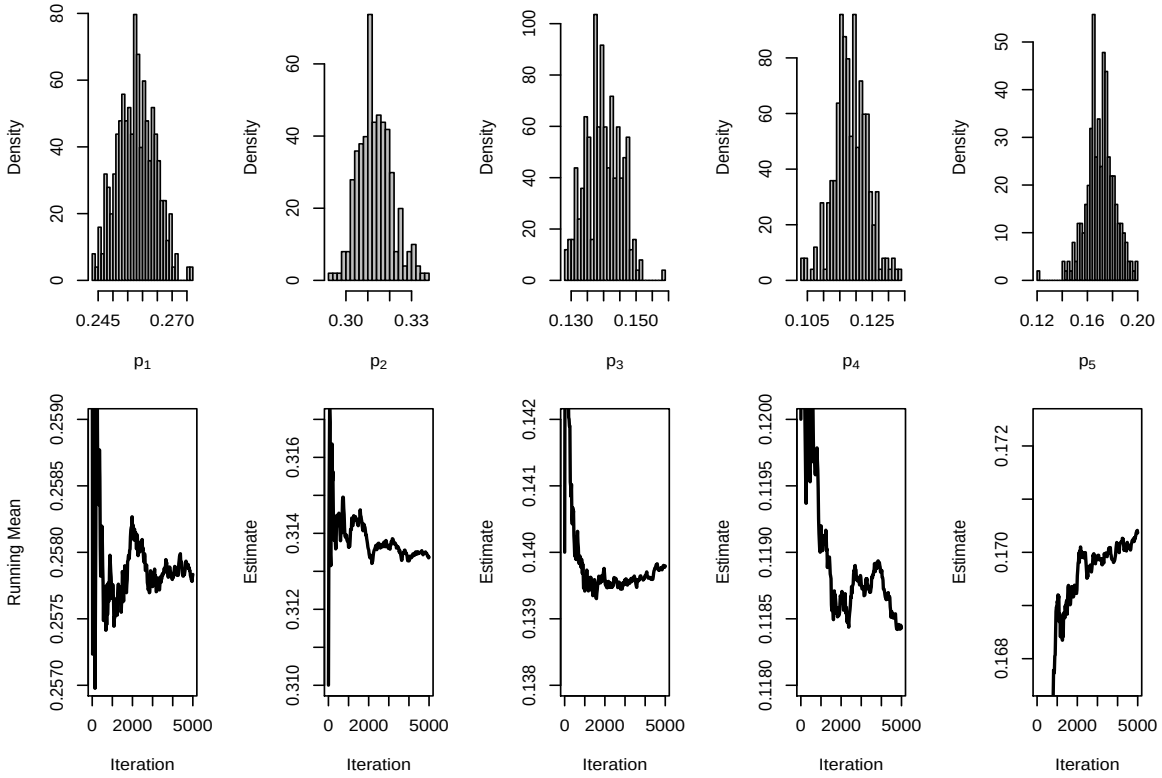
$$\mathbf{p}|W_1, W_2, \ldots, W_5, \sum_i Z_i \sim \mathcal{D}(W_1 + 1, W_2 + 1, \ldots, W_5 + \sum_i Z_i + 1)$$

$$\sum_i Z_i \sim \mathcal{N}eg\left(\sum_{i=1}^{m} n_i + m, 1 - p_5\right).$$

- The point estimates agree with those of the EM algorithm, $\hat{p} = (0.258, 0.313, 0.140, 0.118, 0.170)$, with the exception of $\hat{p}_5$, which is larger than the MLE.

Cellular phone Gibbs

- Gibbs output for cellular phone data, 5000 iterations



Section 9.1.4: The Hammersley–Clifford Theorem

- A most surprising feature of the Gibbs sampler is that the
  - conditional distributions contain sufficient information to produce a sample from the joint distribution.

- This is the case for both two-stage and multi-stage Gibbs
  - The full conditional distributions perfectly summarize the joint density,
  - although the set of marginal distributions obviously fails to do so

The Hammersley–Clifford Theorem

- The following result then shows that the joint density can be directly and constructively derived from the conditional densities.

  Theorem: The joint distribution associated with the conditional densities $f_{Y|X}(y|x)$ and $f_{X|Y}(x|y)$ has the joint density

$$f(x,y) = \frac{f_{Y|X}(y|x)}{\int \left[f_{Y|X}(y|x)/f_{X|Y}(x|y) \right] dy}.$$

- Note that the joint is written using conditionals

The Hammersley–Clifford Theorem – Proof

- $f(x, y) = f(x|y)f(y) = f(y|x)f(x)$, so
- $\frac{f(y)}{f(x)} = \frac{f(y|x)}{f(x|y)}$, and
- $\int \frac{f(y)}{f(x)}\, dy = \frac{1}{f(x)} = \int \frac{f(y|x)}{f(x|y)}\, dy$
- So the marginal is written only with conditionals and

$$f(x, y) = f(y|x)f(x) = \frac{f(y|x)}{\int \frac{f(y|x)}{f(x|y)}\, dy}$$

The Multi-Stage Gibbs Sampler

- Suppose that for some $p > 1$, the random variable $\mathbf{X} \in \mathcal{X}$ can be written as $\mathbf{X} = (X_1, \ldots, X_p)$, where the X_i's are either uni- or multidimensional.
- Moreover, suppose that we can simulate from the corresponding univariate conditional densities $f_1, \ldots, f_p$, that is, we can simulate

  $$X_i | x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_p \sim f_i(x_i | x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_p)$$

  for $i = 1, 2, \ldots, p$.

The Multi-Stage Gibbs Sampler

Given $\mathbf{x}^{(t)} = (x_1^{(t)}, \ldots, x_p^{(t)})$, generate

1. $X_1^{(t+1)} \sim f_1(x_1 | x_2^{(t)}, \ldots, x_p^{(t)})$;

2. $X_2^{(t+1)} \sim f_2(x_2 | x_1^{(t+1)}, x_3^{(t)}, \ldots, x_p^{(t)})$,

$\vdots$

p. $X_p^{(t+1)} \sim f_p(x_p | x_1^{(t+1)}, \ldots, x_{p-1}^{(t+1)})$.

- The densities $f_1, \ldots, f_p$ are called the *full conditionals*
- These are the only densities used for simulation, even in a high-dimensional problem.

Hierarchical Models - Introduction

- A hierarchical model is of the form

$$\begin{aligned} \mathbf{X} &\sim f(\mathbf{x}|\theta) \\ \theta &\sim g(\theta|\beta) \\ \beta &\sim h(\beta|\lambda) \\ \lambda &\sim k(\lambda) \end{aligned}$$

  - All hyperparameters specified at deepest level
  - Effect of deeper hyperparameters is lower
- Easy to get joint distribution
- Easy to pick off full conditionals

Hierarchical Models - Introduction - 2

- Hierarchical Model

$$\begin{aligned} \mathbf{X} &\sim f(\mathbf{x}|\theta) \\ \theta &\sim \pi(\theta|\beta) \\ \beta &\sim \pi(\beta|\lambda) \\ \lambda &\sim \pi(\lambda) \end{aligned}$$

- Joint distribution

$$f(\mathbf{x}|\theta) \times \pi(\theta|\beta) \times \pi(\beta|\lambda) \times \pi(\lambda)$$

- Full Conditionals

$$\pi(\theta|\mathbf{x}, \beta, \lambda) \propto \text{ terms in joint involving } \theta$$

etc...

Hierarchical Models - Introduction - 3

- Normal Hierarchical Model (Conjugate)

$$\begin{aligned} \mathbf{X} &\sim N(\theta, \sigma^2) \\ \theta &\sim N(\theta_0, \tau^2\sigma^2) \\ \sigma^2 &\sim \text{Inverted Gamma}(a, b) \end{aligned}$$

- Here θ_0, τ^2, a, b are specified
  - Usual to take $\tau^2 \approx 10$ (variance ratio)
  - Choose a, b to give prior a big variance

Normal Hierarchical Models

- Normal Hierarchical Model

$$\begin{aligned} X_i &\sim N(\theta, \sigma^2), \quad i = 1, \ldots, n \\ \theta &\sim N(\theta_0, \tau^2\sigma^2) \\ \sigma^2 &\sim \text{Inverted Gamma}(a, b) \end{aligned}$$

- Joint Distribution

$$f(\mathbf{x}, \theta, \sigma^2) \propto \left[\frac{1}{\sigma} e^{-\sum_i (x_i - \theta)^2/(2\sigma^2)}\right] \times \left[\frac{1}{\tau\sigma} e^{-(\theta-\theta_0)^2/(2\tau^2\sigma^2)}\right] \times \left[\frac{1}{(\sigma^2)^{a+1}} e^{1/b\sigma^2}\right]$$

Normal Hierarchical Models -2

- Joint Distribution

$$f(\mathbf{x},\theta,\sigma^2) \propto \left[\frac{1}{\sigma}e^{-\sum_i(x_i-\theta)^2/(2\sigma^2)}\right] \times \left[\frac{1}{\tau\sigma}e^{-(\theta-\theta_0)^2/(2\tau^2\sigma^2)}\right] \times \left[\frac{1}{(\sigma^2)^{a+1}}e^{1/b\sigma^2}\right]$$

- θ full conditional

$$\pi(\theta|\mathbf{x},\sigma^2) \propto \left[\frac{1}{\sigma}e^{-\sum_i(x_i-\theta)^2/(2\sigma^2)}\right] \times \left[\frac{1}{\tau\sigma}e^{-(\theta-\theta_0)^2/(2\tau^2\sigma^2)}\right] \times \left[\frac{1}{(\sigma^2)^{a+1}}e^{1/b\sigma^2}\right]$$

$$= \text{Normal}$$

- σ^2 full conditional

$$\pi(\sigma^2|\mathbf{x},\theta) \propto \left[\frac{1}{\sigma}e^{-\sum_i(x_i-\theta)^2/(2\sigma^2)}\right] \times \left[\frac{1}{\tau\sigma}e^{-(\theta-\theta_0)^2/(2\tau^2\sigma^2)}\right] \times \left[\frac{1}{(\sigma^2)^{a+1}}e^{1/b\sigma^2}\right]$$

$$= \text{Inverted Gamma}$$

Normal Hierarchical Models -3

- To estimate θ and σ^2

$$\begin{aligned} X_i &\sim N(\theta, \sigma^2), \quad i = 1, \ldots, n \\ \theta &\sim N(\theta_0, \tau^2\sigma^2) \\ \sigma^2 &\sim \text{Inverted Gamma}(a, b) \end{aligned}$$

- Use a Gibbs sampler with

$$\theta \sim \mathrm{N}\left(\frac{1}{1+n\tau^2}\,\theta_0 + \frac{n\tau^2}{1+n\tau^2}\,\bar{x},\ \frac{\sigma^2\tau^2}{1+n\tau^2}\right)$$

$$\frac{1}{\sigma^2} \sim \mathrm{Gamma}\left(\frac{n+1}{2} + a,\ \frac{1}{\frac{\sum_i (x_i-\theta)^2}{2} + \frac{(\theta-\theta_0)^2}{2} + \frac{1}{b}}\right)$$

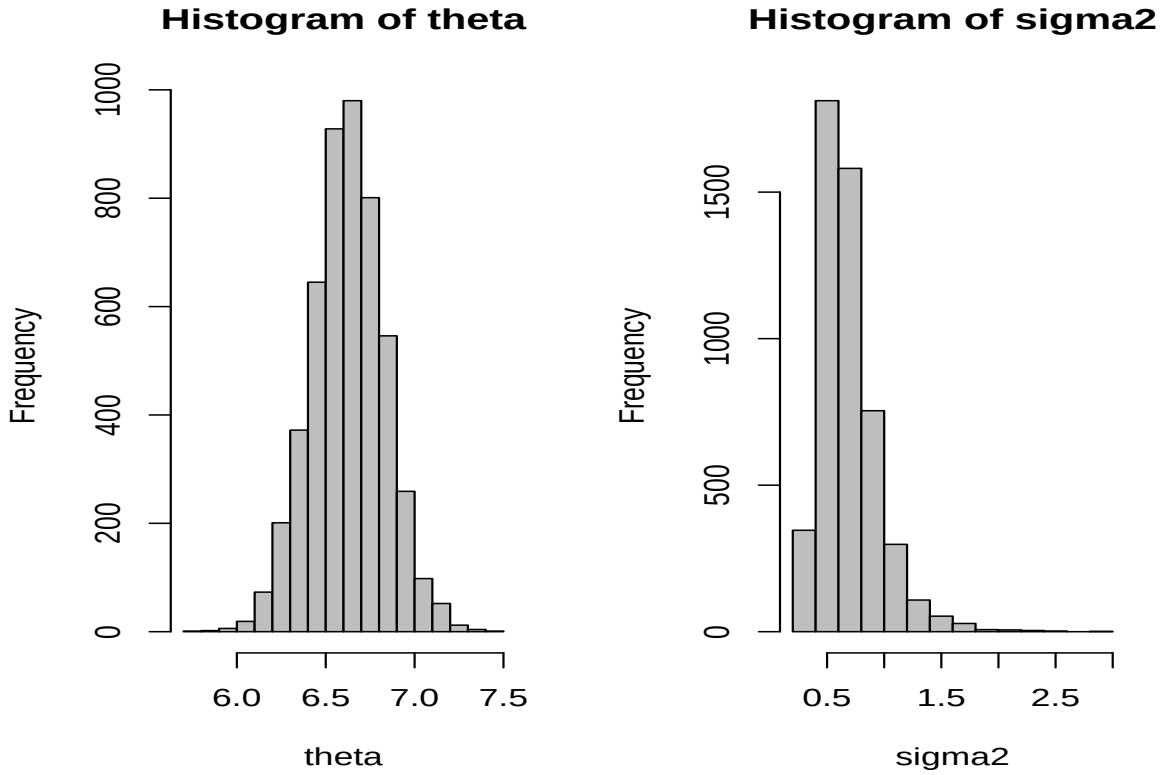
Example

- Energy Intake (Megajoules) over 24 hours, 15 year old females

91	504	557	609	693	727	764	803
857	929	970	1043	1089	1195	1384	1713

Example

- Energy Intake (Megajoules) over 24 hours, 15 year old females
- R program NormalHierarchy-1



Normal Hierarchical Models -3a

- To avoid specifying θ_0 use the hierarchy

$$\begin{aligned} X_i &\sim N(\theta, \sigma^2), \quad i = 1, \ldots, n \\ \theta &\sim \text{Uniform}(-\infty, \infty) \\ \sigma^2 &\sim \text{Inverted Gamma}(a, b) \end{aligned}$$

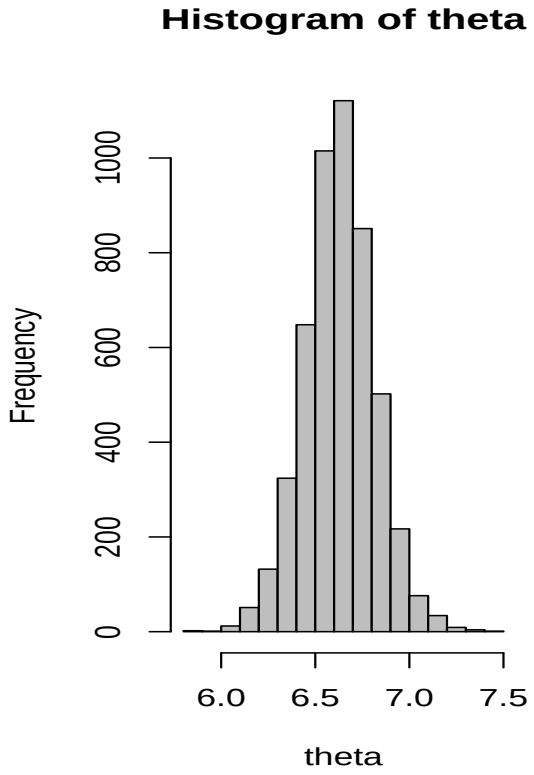
- which gives a Gibbs sampler with

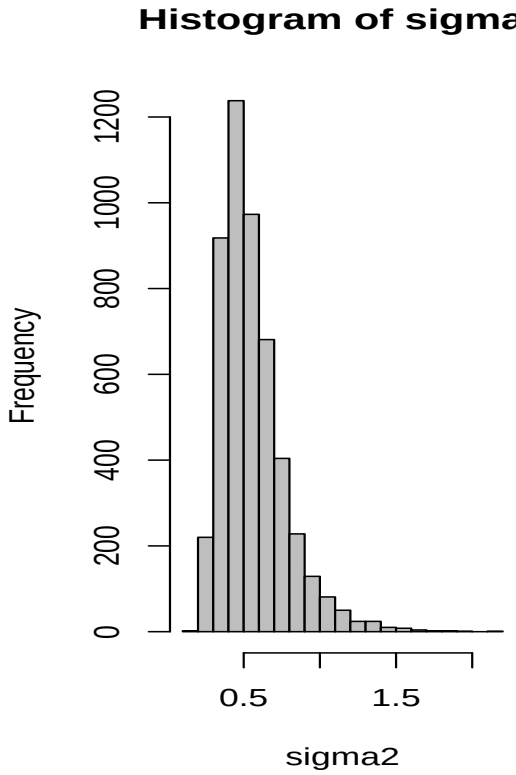
$$\theta \sim \mathrm{N}\left(\bar{x},\ \sigma^2\right)$$

$$\frac{1}{\sigma^2} \sim \text{Gamma}\left(\frac{n}{2} + a,\ \frac{1}{\frac{\sum_i (x_i - \theta)^2}{2} + \frac{1}{b}}\right)$$

Example

- Energy Intake (Megajoules) over 24 hours, 15 year old females
- R program NormalHierarchy-2





Normal Hierarchical Models -4

- A bit more complicated - oneway anova: $Y_{ij} = \mu + \alpha_i + \varepsilon_{ij}$
- A full hierarchical specification

$$\begin{aligned} Y_{ij} &\sim N(\mu + \alpha_i, \sigma^2), \quad i = 1, \ldots, k, \quad j = 1, \ldots, n_i \\ \mu &\sim \text{Uniform}(-\infty, \infty) \\ \alpha_i &\sim N(0, \tau^2), \quad i = 1, \ldots, k \\ \sigma^2 &\sim \text{Inverted Gamma}(a_1, b_1) \\ \tau^2 &\sim \text{Inverted Gamma}(a_2, b_2) \end{aligned}$$

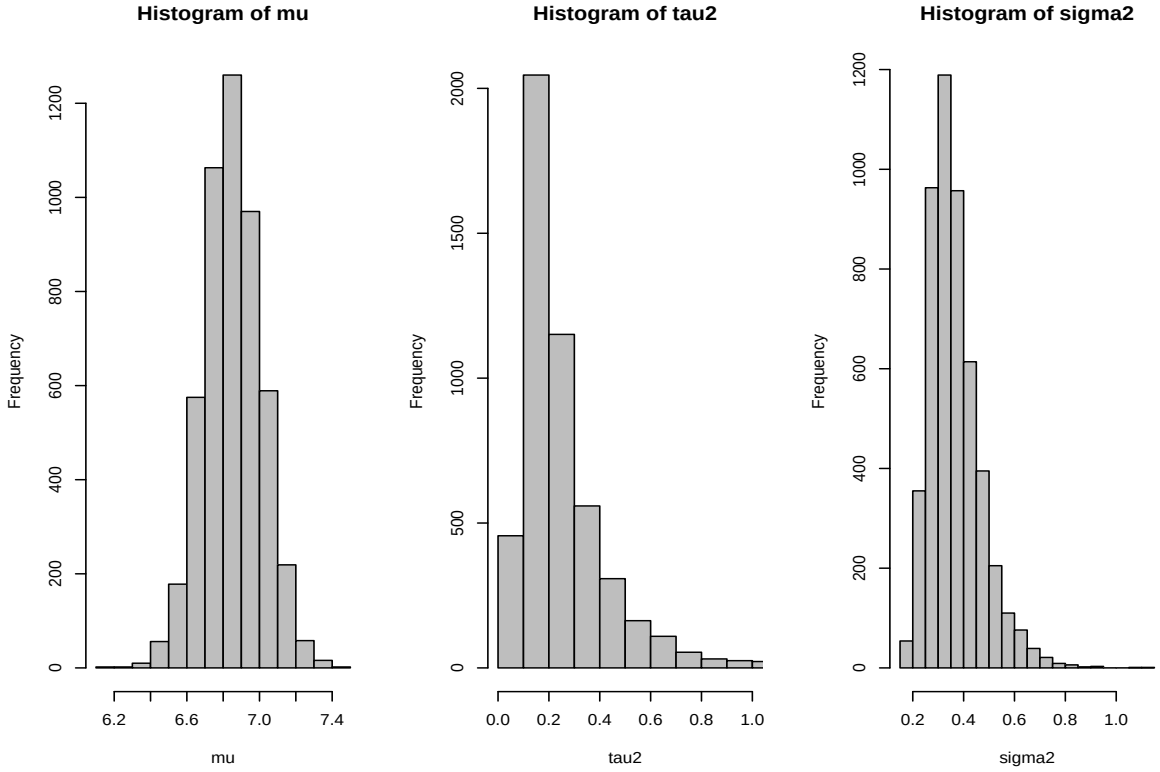
Normal Hierarchical Models -4a

- Oneway anova: $Y_{ij} = \mu + \alpha_i + \varepsilon_{ij}$
- with Gibbs sampler

$$\begin{aligned}
\mu &\sim N\left(\bar{y} - \bar{\alpha},\ \frac{\sigma^2}{\sum_i n_i}\right) \\
\alpha_i &\sim N\left(\frac{n_i\sigma^2\tau^2}{\sigma^2 + n_i\tau^2}(\bar{y}_i - \mu),\ \frac{\sigma^2\tau^2}{\sigma^2 + n_i\tau^2}\right) \\
\frac{1}{\sigma^2} &\sim \text{Gamma}\left(\frac{\sum_i n_i}{2} + a_1,\ \frac{1}{\frac{\sum_{ij}(y_{ij}-\alpha_i-\mu)^2}{2} + \frac{1}{b_1}}\right) \\
\frac{1}{\tau^2} &\sim \text{Gamma}\left(\frac{k}{2} + a_2,\ \frac{1}{\frac{\sum_i \alpha_i^2}{2} + \frac{1}{b_2}}\right)
\end{aligned}$$

Example

- Energy Intake (Megajoules) over 24 hours, 15 year old females and 15 year old males
- R program NormalHierarchy-3



Age Distribution of Chinook Salmon - 1

- Chinook salmon spawn in fresh water and the juveniles hatch and swim out to sea
  - They return to their natal stream to spawn 3 to 7 years later.
  - Fish of multiple ages return to the stream
- We want estimates of the age composition
  - Take scales from a sample of fish and count the annuli.
  - This is time-consuming and expensive
  - Use length as a proxy for age - easier and faster to obtain
- Now we will use both length and age.

Age Distribution of Chinook Salmon - 2

- Observe (y_i, x_i), $y_i =$ Age, $x_i =$ length, where

$$f(y_i, x_i|\mathbf{p}, \boldsymbol{\mu}, \boldsymbol{\sigma}) \propto \prod_{j=3}^{7} p_j^{I(y_i=j)} \frac{1}{\sigma_{y_i}} \exp\left\{-\frac{(x_i - \mu_{y_i})^2}{2\sigma_{y_i}^2}\right\}$$

- And we can write the full likelihood as

$$L(\mathbf{p}, \boldsymbol{\mu}, \boldsymbol{\sigma}|\mathbf{y}, \mathbf{x}) \propto \prod_{j=3}^{7} p_j^{n_j} \frac{1}{\sigma_j^{n_j}} \exp\left\{-\frac{n_j s_j^2 + n_j(\bar{x}_j - \mu_j)^2}{2\sigma_j^2}\right\}$$

  - $n_j = (\#y_i = j)$, $\sum_j n_j = n$
  - $\bar{x}_j = \frac{1}{n_j} \sum_{i:y_i=j} x_i$
  - $s_j^2 = \frac{1}{n_j} \sum_{i:y_i=j} (x_i - \bar{x}_j)^2$

Age Distribution of Chinook Salmon -3

- With no missing y_i the likelihood factors

$$L(\mathbf{p}, \boldsymbol{\mu}, \boldsymbol{\sigma}|\mathbf{y}, \mathbf{x}) \propto \left[\prod_{j=3}^{7} p_j^{n_j}\right] \left[\prod_{j=3}^{7} \frac{1}{\sigma_j^{n_j}} \exp\left\{-\frac{n_j s_j^2 + n_j(\bar{x}_j - \mu_j)^2}{2\sigma_j^2}\right\}\right]$$

$$\hat{p}_j = \frac{n_j}{\sum_j n_j}$$

$$\hat{\mu}_j = \bar{x}_j$$

$$\hat{\sigma}_j^2 = s_j^2$$

Age Distribution of Chinook Salmon - 4

- A Bayesian Analysis
- Prior specifications
  - $\mathbf{p} \sim$ Dirichlet$(\alpha_3, \ldots, \alpha_7)$
  - $\mu_j \sim$ Normal(μ_{j0}, τ_j^2)
  - $\sigma_j^2 \sim$ Inverted Gamma(a, b)
- Full conditionals for a Gibbs Sampler
  - $\mathbf{p} \sim$ Dirichlet$(n_3 + \alpha_3, \ldots, n_7 + \alpha_7)$
  - $\mu_j \sim$ Normal $\left(\frac{n_j \tau_j^2}{n_j \tau_j^2 + \sigma_j^2} \, \bar{x}_j + \frac{\sigma_j^2}{n_j \tau_j^2 + \sigma_j^2} \, \mu_{j0}, \; \frac{\sigma_j^2 \tau_j^2}{n_j \tau_j^2 + \sigma_j^2} \right)$
  - $\sigma_j^2 \sim$ Inverted Gamma $\left(\frac{n_j}{2} + a, \frac{n_j s_j^2}{2} + \frac{1}{b} \right)$
- Notice that $\mathbf{p}$ only depends on n_j.

Age Distribution of Chinook Salmon - 5

- With missing y_i things are more interesting
  - Write $n = n_{\text{obs}} + n_m$ = Observed + Missing
  - $\mathbf{y} = \mathbf{y}_{\text{obs}} + \mathbf{y}_m$ = Observed + Missing
- Now the likelihood is

$$L(\mathbf{p}, \boldsymbol{\mu}, \boldsymbol{\sigma}|\mathbf{y}, \mathbf{x}) \propto \prod_{j=3}^{7} p_j^{n_j} \frac{1}{\sigma_j^{n_j}} \exp\left\{-\frac{n_j s_j^2 + n_j(\bar{x}_j - \mu_j)^2}{2\sigma_j^2}\right\} \times \prod_{i=1}^{n_m} \prod_{j=3}^{7} \left(p_j \frac{1}{\sigma_j} \exp\left\{-\frac{(x_i - \mu_j)^2}{2\sigma_j^2}\right\}\right)^{I(y_{mi}=j)}$$

  - where $n_j, \bar{x}_j, s_j^2$ are defined for the observed data.

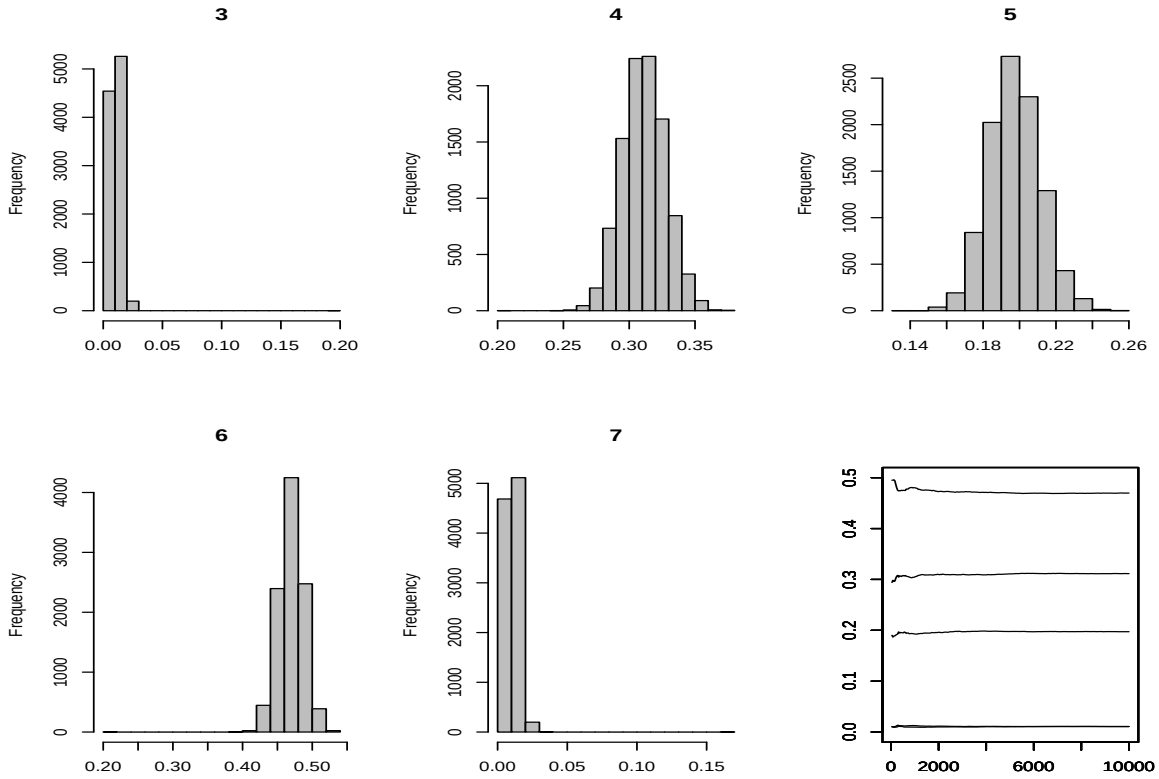
Age Distribution of Chinook Salmon - 7

- The Gibbs sampler fills in missing Age data

$$Y_{mi} \sim \text{Multinomial}\left(p_j \frac{1}{\sigma_j} \exp\left\{-\frac{(x_i-\mu_j)^2}{2\sigma_j^2}\right\}\right)$$

- Then updates the parameters
  - $\mathbf{p} \sim$ Dirichlet$(n_3+\alpha_3, \ldots, n_7+\alpha_7)$
  - $\mu_j \sim$ Normal $\left(\frac{n_j\tau_j^2}{n_j\tau_j^2+\sigma_j^2}\,\bar{x}_j + \frac{\sigma_j^2}{n_j\tau_j^2+\sigma_j^2}\,\mu_{j0},\ \frac{\sigma_j^2\tau_j^2}{n_j\tau_j^2+\sigma_j^2}\right)$
  - $\sigma_j^2 \sim$ Inverted Gamma $\left(\frac{n_j}{2}+a, \frac{n_j s_j^2}{2}+\frac{1}{b}\right)$
- where n_j, $\bar{x}_j$ and s_j^2 are recalculated for each new Y_m.

Age Distribution of Chinook Salmon - 8



A lazy hierarchical specification

$$\begin{aligned} Y_{ij} &\sim N(\mu + \alpha_i, \sigma^2), \quad i = 1, \ldots, k, \quad j = 1, \ldots, n_i \\ \alpha_i &\sim N(0, \tau^2), \quad i = 1, \ldots, k \end{aligned}$$

- The classical random effects model
- We can set up a Gibbs sampler

Random Effects Model

$$\begin{aligned} Y_{ij} &\sim N(\mu + \alpha_i, \sigma^2), \quad i = 1, \ldots, k, \quad j = 1, \ldots, n_i \\ \alpha_i &\sim N(0, \tau^2), \quad i = 1, \ldots, k \end{aligned}$$

- with Gibbs sampler

$$\begin{aligned} \alpha_i &\sim \mathrm{N}\left(\frac{n_i\tau^2}{n_i\tau^2 + \sigma^2}\,(\bar{y}_i - \mu),\ \frac{n_i\tau^2\sigma^2}{n_i\tau^2 + \sigma^2}\right) \\ \mu &\sim \mathrm{N}\left(\bar{y} - \bar{\alpha},\ \frac{\sigma^2}{\sum_i n_i}\right) \\ \frac{1}{\sigma^2} &\sim \mathsf{Gamma}\left(\frac{\sum_i n_i}{2} - 1,\ \frac{2}{\sum_{ij}(y_{ij} - \mu - \alpha_i)^2}\right) \\ \frac{1}{\tau^2} &\sim \mathsf{Gamma}\left(\frac{k}{2} - 1,\ \frac{2}{\sum_i \alpha_i^2}\right) \end{aligned}$$

Problem!!

- This is not a Gibbs sampler
- Conditional distributions do not exist!
- Result of using improper priors
  - Improper priors sometimes OK
  - Sometimes: bad conditionals
  - Sometimes: good conditionals, bad posterior ← REAL BAD
  - Extremely hard to detect
- Moral: Best to use proper priors

It's better to be lucky than good

- Looking for simple example (Am. Statistician 1992)

$$X|Y = y \sim ye^{-yx}, \qquad Y|X = x \sim xe^{-xy}$$

It's better to be lucky than good

- Looking for simple example (Am. Statistician 1992)

$$X|Y = y \sim ye^{-yx}, \qquad Y|X = x \sim xe^{-xy}$$

- This is not a Gibbs sampler
- No joint distribution exists!
- Hammersley-Clifford $\Rightarrow$

$$f(x, y) = \frac{e^{-xy}}{\int_0^\infty \frac{1}{y}\, dy}$$

Hierarchical Models: Animal epidemiology

- Research in animal epidemiology sometimes uses data from groups of animals, such as litters or herds.
- Such data may not follow some of the usual assumptions of independence, etc., and, as a result, variances of parameter estimates tend to be larger ("overdispersion")
- Data on the number of cases of clinical mastitis in dairy cattle herds over a one year period.

Hierarchical Models: Animal epidemiology

- $X_i \sim \mathcal{P}(\lambda_i)$, where λ_i is the underlying rate of infection in herd i
- To account for overdispersion, put a gamma prior distribution on the Poisson parameter. A complete hierarchical specification is

$$\begin{aligned} X_i &\sim \mathcal{P}(\lambda_i), \\ \lambda_i &\sim \mathcal{G}a(\alpha, \beta_i), \\ \beta_i &\sim \mathcal{G}a(a, b), \end{aligned}$$

  where α, a, and b are specified.

- The posterior density of λ_i, $\pi(\lambda_i|\mathbf{x}, \alpha)$, can now be simulated via the Gibbs sampler

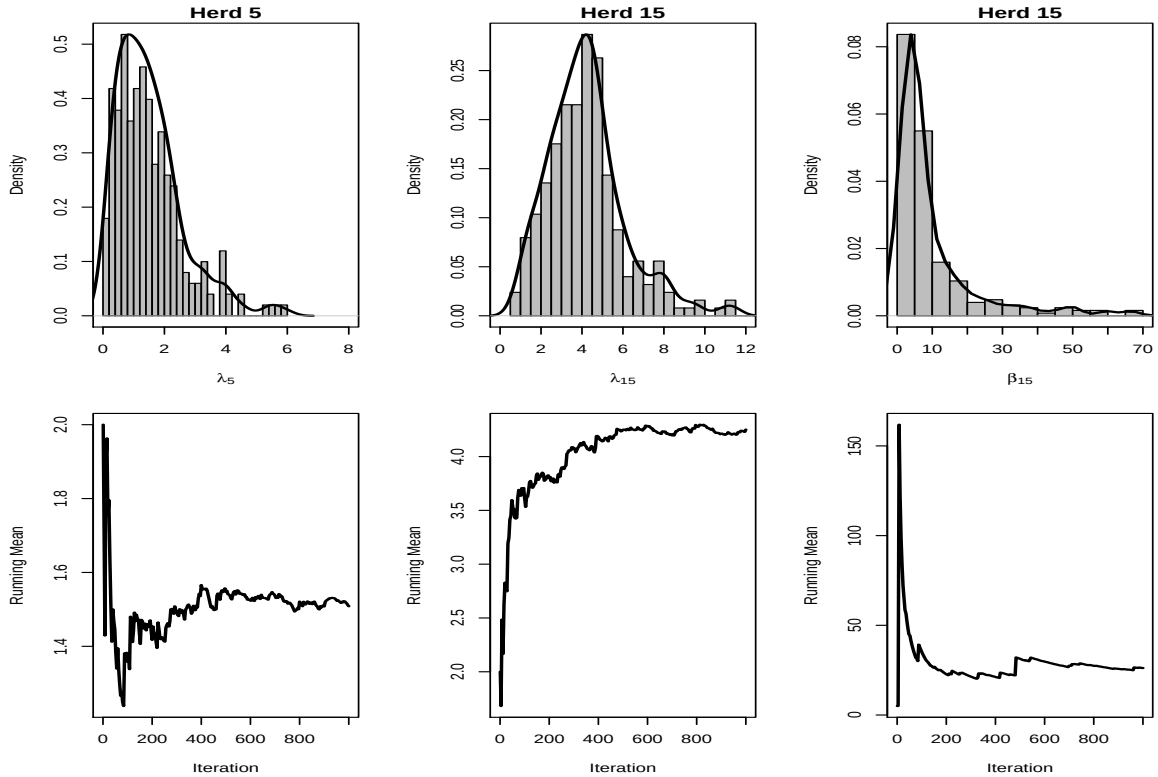
$$\begin{aligned} \lambda_i &\sim \pi(\lambda_i|\mathbf{x}, \alpha, \beta_i) = \mathcal{G}a(x_i + \alpha, 1 + \beta_i), \\ \beta_i &\sim \pi(\beta_i|\mathbf{x}, \alpha, a, b, \lambda_i) = \mathcal{G}a(\alpha + a, \lambda_i + b)\,. \end{aligned}$$

Animal Epidemiology R code

```
xdata <-c(0,0,1,1,2,2,2,2,2,2,4,4,4,5,5,5,5,5,5,6,6,8,8,8,9,9,9,
      10,10,12,12,13,13,13,13,18,18,19,19,19,19,20,20,22,22,22,23,25)
nx<-length(xdata)
nsim<-1000;
lambda<-array(2,dim=c(nsim,nx));beta<-array(5,dim=c(nsim,nx));
alpha<-.1;a<-1;b<-1;
for(i in 2:nsim){
for(j in 1:nx){
beta[i,j]<-1/rgamma(1,shape=alpha+a,scale=1/(lambda[i-1,j]+(1/b)));
lambda[i,j]<-rgamma(1,shape=xdata[j]+alpha,scale=1/(1+(1/beta[i,j])))
}
}
```

Gibbs sampler output

- Selected estimates of λ_i and β_i.



Prediction - Introduction

- For the simple model

$$\begin{aligned} \mathbf{X} &\sim f(\mathbf{x}|\theta) \\ \theta &\sim g(\theta) \end{aligned}$$

- The predictive density of a new $\mathbf{X}$ is

$$\pi(x_{\text{new}}|\mathbf{x}_{\text{old}}) = \int f(x_{\text{new}}|\theta)\pi(\theta|\mathbf{x}_{\text{old}})d\theta$$

  - $\pi(\theta|\mathbf{x}_{\text{old}})$ is the posterior density
  - Averages over the parameter values

- If $\theta_1, \ldots, \theta_M \sim \pi(\theta|\mathbf{x}_{\text{old}})$

$$\pi(x_{\text{new}}|\mathbf{x}_{\text{old}}) \approx \frac{1}{M}\sum_i f(x_{\text{new}}|\theta_i)$$

Prediction - Introduction -2

- For the hierarchical model

$$\begin{aligned} \mathbf{X} &\sim f(\mathbf{x}|\theta) \\ \theta &\sim g(\theta|\beta) \\ \beta &\sim h(\beta|\lambda) \\ \lambda &\sim k(\lambda) \end{aligned}$$

- the Gibbs sampler give us a $(\theta_i, \beta_i, \lambda_i)$, $i = 1, \ldots, M$
  - A sample from the joint distribution.
- Using Monte Carlo sums

$$\pi(x_{\text{new}}|\mathbf{x}_{\text{old}}) \approx \frac{1}{M} \sum_i f(x_{\text{new}}|\theta_i)$$

- A Conditionally Independent Hierarchical Model

Oneway Anova Predictive Density

- Energy Intake - oneway anova: $Y_{ij} = \mu + \alpha_i + \varepsilon_{ij}$
- A full hierarchical specification

$$\begin{aligned} Y_{ij} &\sim N(\mu + \alpha_i, \sigma^2), \quad i = 1, \ldots, k, \quad j = 1, \ldots, n_i \\ \mu &\sim \text{Uniform}(-\infty, \infty) \\ \alpha_i &\sim N(0, \tau^2), \quad i = 1, \ldots, k \\ \sigma^2 &\sim \text{Inverted Gamma}(a_1, b_1) \\ \tau^2 &\sim \text{Inverted Gamma}(a_2, b_2) \end{aligned}$$

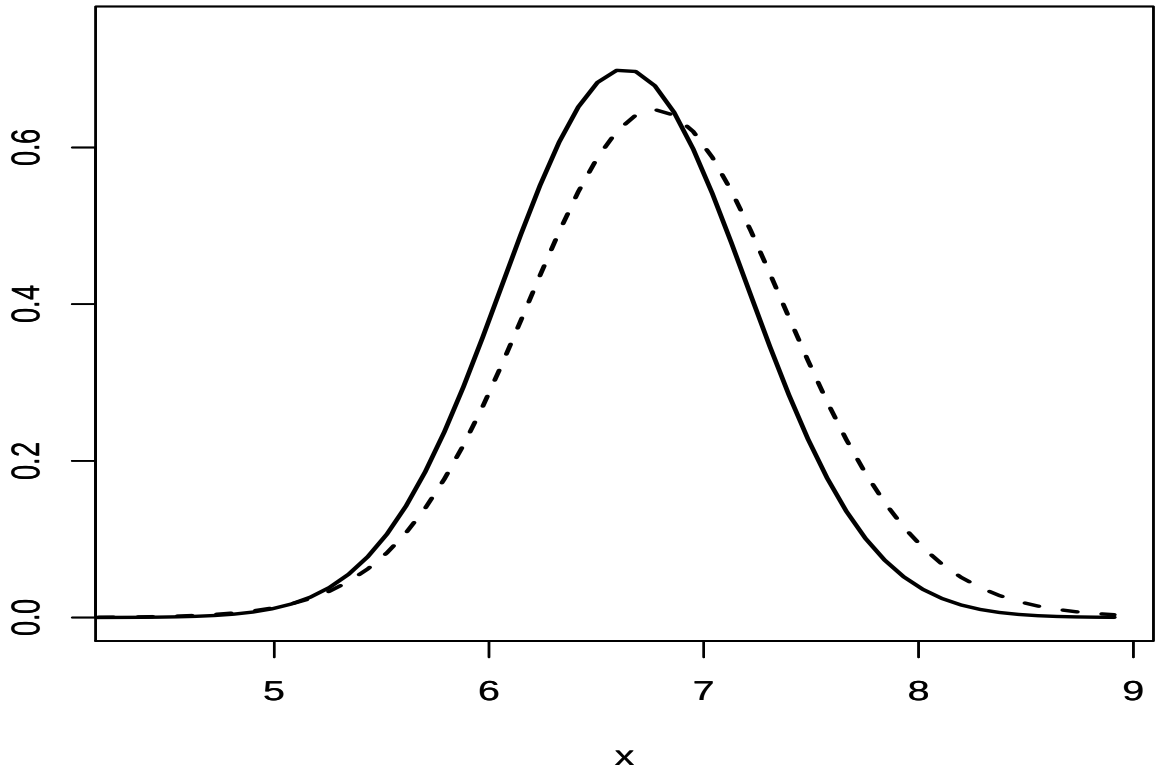
- Predictive Density for Group i

$$\pi(y_{\text{new}}|\mathbf{y}) = \frac{1}{M} \sum_{j=1}^{M} \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-.5(y_{\text{new}} - \mu_j - \alpha_{ij})^2/\sigma_j^2}$$

where $(\mu_j, \alpha_{ij}, \sigma_j^2)$ are a sample from the posterior distribution.

Energy Intake - Predictive density for females

- R program NormalPrediction-3



- solid = "naive" prediction
- dashed = predictive density

PKPD Medical Models

- ***Pharmacokinetics*** is the modeling of the relationship between the dosage of a drug and the resulting concentration in the blood.
- Gilks *et al.* (1993) approach:
  - Estimate pharmacokinetic parameters using mixed-effects model and nonlinear structure
  - Also robust to the outliers common to clinical trials
- For a given dose d_i administered at time 0 to patient i, the measured log concentration in the blood at time t_{ij}, X_{ij}, is assumed to follow a normal distribution

$$X_{ij} \sim N(\log g_{ij}(\lambda_i), \sigma^2),$$

PKPD Medical Models

- $X_{ij} \sim N(\log g_{ij}(\lambda_i), \sigma^2)$,
- $\lambda_i = (\log C_i, \log V_i)'$ are parameters for the ith individual, σ^2 is the measurement error variance, and g_{ij} is given by

$$g_{ij}(\lambda_i) = \frac{d_i}{V_i} \exp\left(-\frac{C_i\, t_{ij}}{V_i}\right).$$

  - C_i represents $clearance$
  - V_i represents $volume$ for patient i.
- We complete the hierarchical specification with

$$\log C_i \sim \mathcal{N}(\mu_C, \sigma_C^2) \text{ and } \log V_i \sim \mathcal{N}(\mu_V, \sigma_V^2).$$

  with $\mu_C, \sigma_C^2, \mu_V, \sigma_V^2$ fixed.

PKPD Medical Models

- The posterior density is proportional to

$$\pi(C_i, V_i) \propto \prod_i \prod_j \left(\exp\left\{ -\frac{x_{ij} - \log g_{ij}}{2\sigma^2} \right\} \right) \times \exp\left\{ -\frac{\log C_i - \mu_C}{2\sigma_C^2} \right\} \times \exp\left\{ -\frac{\log V_i - \mu_V}{2\sigma_V^2} \right\},$$

- The full conditional of C_i is

$$\pi(C_i) \propto \prod_i \prod_j \exp\left\{ -\frac{x_{ij} - \log g_{ij}}{2\sigma^2} \right\} \times \exp\left\{ -\frac{\log C_i - \mu_C}{2\sigma_C^2} \right\}$$

- Note that to get the full conditional, we "pick off" all terms with C_i.

PKPD Medical Models

- The full conditional of C_i is

$$\pi(C_i) \propto \prod_i \prod_j \exp\left\{-\frac{x_{ij} - \log g_{ij}}{2\sigma^2}\right\} \times \exp\left\{-\frac{\log C_i - \mu_C}{2\sigma_C^2}\right\}$$

- We can write this as

$$\pi(C_i) \propto \exp\left\{\frac{(C_i - V_i B/A)^2}{V_i^2 \sigma^2}\right\} \times \exp\left\{-\frac{\log C_i - \mu_C}{2\sigma_C^2}\right\}$$

  with $A = \sum_j t_{ij}^2$ and $B = \sum_j t_{ij}(X_{ij} + log(d_i/V_i)$.

- Sampling from this is a challenge

PKPD Medical Models

- The full conditional is

$$\pi(C_i) \propto \exp\left\{\frac{(C_i - V_i B/A)^2}{V_i^2 \sigma^2}\right\} \times \exp\left\{-\frac{\log C_i - \mu_C}{2\sigma_C^2}\right\}$$

- Some options - use Metropolis
  - Candidate is

$$\mathcal{N}\left(V_i B/A, V_i^2 \sigma^2\right)$$

  - Use Taylor: $\log C_i = \log \mu_C + \frac{C_i - \mu_C}{\mu_C}$ to get candidate

$$\mathcal{N}\left(\frac{\sigma_c^2 \mu_C^2 V_i B/A + \sigma^2 V_i^2 \mu_C}{\sigma_c^2 \mu_C^2 + \sigma^2 V_i^2}, \frac{\sigma_c^2 \mu_C^2 \sigma^2 V_i^2}{\sigma_c^2 \mu_C^2 + \sigma^2 V_i^2}\right)$$

  - V_i is even harder

PKPD Medical Models

- Plan B: Use WinBugs
- Uses Metropolis with Adaptive Rejection Sampling
- But.... Lets start simple with `WinBugs`

Specifying Models with WinBugs

- There are three steps to producing an MCMC model in WinBugs:
  - Specify the distributional features of the model, and the quantities to be estimated.
  - Compile the instructions into the run-time program.
  - Run the sampler which produces Markov chains.
- Remember that the first step must identify the full distributions for each variable in the model.

WinBugs - Starting Simple

- Normal Hierarchical Model (Conjugate)

$$\begin{aligned} \mathbf{X} &\sim N(\theta, \sigma^2) \\ \theta &\sim N(\theta_0, \tau^2\sigma^2) \\ \sigma^2 &\sim \text{Inverted Gamma}(a, b) \end{aligned}$$

- Here θ_0, τ^2, a, b are specified
  - Each variable must be specified, or have a distribution
  - NO improper priors allowed

To Run WinBugs

- Model
  - Specification Tool: Highlight and Check Model
  - Data: Highlight and Load Data
  - Compile
  - Inits: Highlight and Load
- Inference
  - Sample Monitor Tool: enter nodes (parameters) stats, trace density
- Model Update Tool

To Run WinBugs - 2

$$\begin{aligned} \mathbf{X} &\sim N(\theta, \sigma^2) \\ \theta &\sim N(\theta_0, \tau^2\sigma^2) \\ \sigma^2 &\sim \text{Inverted Gamma}(a, b) \end{aligned}$$

```
model
    {  for( i in 1 : N )
        {
        X[i] ~ dnorm(theta,sigma2)
        }
       theta ~  dnorm(theta0,v)
        v <- tau2*sigma2
        sigma2 ~ dgamma(1,1)
        theta0 <- 6
        tau2 <- 10
      }
```

- WinBugs - Simple.odc

WinBugs - Another Example

- Logistic Regression

$$Y \sim \text{Bernoulli}(p(\mathbf{x})), \quad \text{logit}p(\mathbf{x}) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2$$

  - Y = emergency room use
  - x_1= health category
  - x_2= health care provider

- Model

```
{
for( i in 1 : N ) {
logit(p[i]) <- alpha0 + alpha1 * metq[i] + alpha2 * np[i]
er[i] ~ dbern(p[i])
}
alpha0 ~ dnorm(0.0,0.1)
alpha1 ~ dnorm(0.0,0.1)
alpha2 ~ dnorm(0.0,0.1)
}
```

- WinBugs ER.odc

Return to PKPD Medical Models

- Model

$$\begin{aligned} X_{ij} &\sim N(\log g_{ij}(\lambda_i), \sigma^2) \\ g_{ij}(\lambda_i) &= \frac{d_i}{V_i} \exp\left(-\frac{C_i\, t_{ij}}{V_i}\right) \\ \lambda_i &= (\log C_i, \log V_i)' \\ \log C_i &\sim \mathcal{N}(\mu_C, \sigma_C^2), \quad \log V_i \sim \mathcal{N}(\mu_V, \sigma_V^2). \end{aligned}$$

- Model: WinBugs PKWinBugs.odc

```
for( i in 1 : N ) {
for( j in 1 : T ) {
X[i , j] ~ dnorm(g[i , j],sigma)
g[i , j] <- (30/V[i]) *exp(-C[i]*t[j]/V[i])}
        C[i]<-exp(LC[i]); V[i]<-exp(LV[i])
        LC[i] ~ dnorm(mC,sigmaC); LV[i]~ dnorm(mV,sigmaV)}

sigma ~ dgamma(0.01,0.01)
mC ~ dnorm(0.0,1.0E-3)
sigmaC ~ dgamma(0.01,0.01)
mV ~ dnorm(0.0,1.0E-3)
sigmaV ~ dgamma(0.01,0.01)
```

PKPD Medical Models

- Alternative Specification:

$$\frac{X_{ij} - \log g_{ij}(\lambda_i)}{\sigma\sqrt{\nu/(\nu-2)}} \sim \mathcal{T}_\nu.$$

- This is easy for the Gibbs sampler:

$$\mathcal{T}_\nu(x|\mu,\sigma^2) = \int \mathcal{N}(x|\mu,\sigma^2\frac{\nu}{w})\ \text{Gamma}(w|\frac{\nu}{2},\frac{1}{2})\ dw$$

  So to generate $X \sim \mathcal{T}_\nu(x|\mu,\sigma^2)$:

$$\begin{aligned} X|W &\sim \mathcal{N}(x|\mu,\sigma^2\frac{\nu}{W}) \\ W &\sim \text{Gamma}(w|\frac{\nu}{2},\frac{1}{2}) \end{aligned}$$

  which fits right in to the Gibbs sampler

- `WinBugs` PKWinBugs2.odc

PKPD Models - Prediction

- Model

$$\begin{aligned} X_{ij} &\sim N(\log g_{ij}(\lambda_i), \sigma^2) \\ g_{ij}(\lambda_i) &= \frac{d_i}{V_i} \exp\left(-\frac{C_i\, t_{ij}}{V_i}\right) \\ \lambda_i &= (\log C_i, \log V_i)' \\ \log C_i &\sim \mathcal{N}(\mu_C, \sigma_C^2), \quad \log V_i \sim \mathcal{N}(\mu_V, \sigma_V^2). \end{aligned}$$

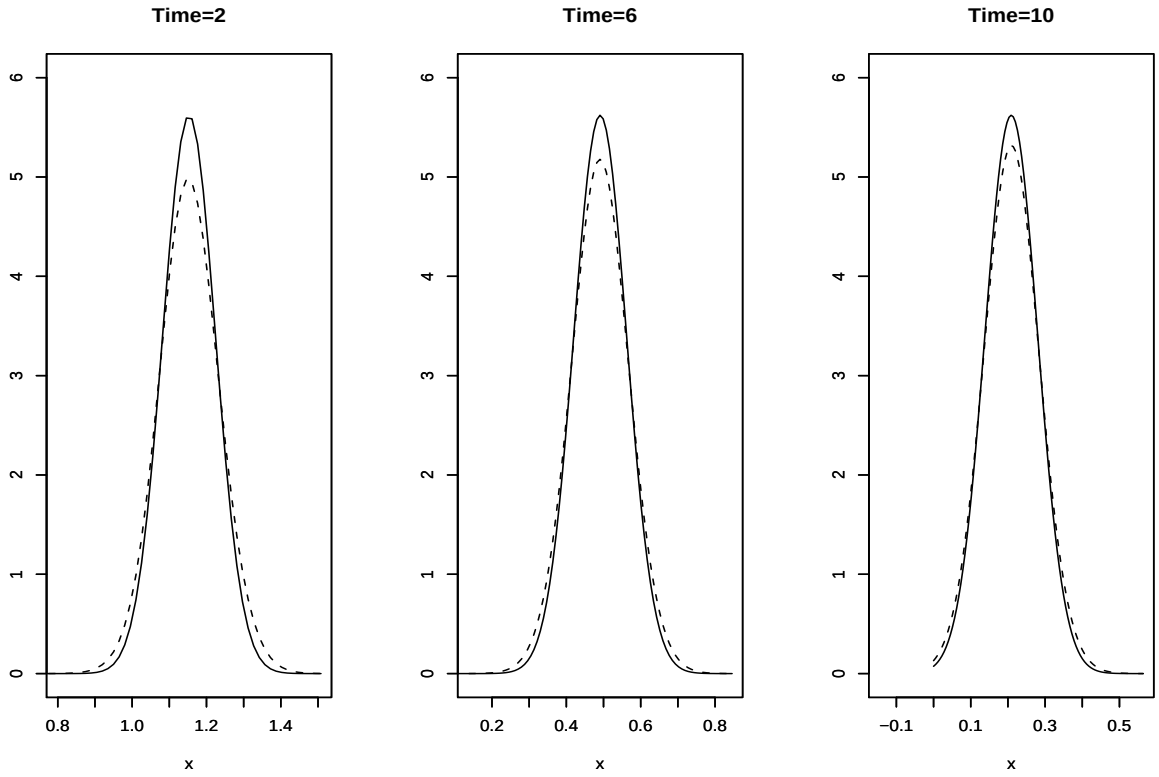
- Predictive density for individual i at time j is

$$\begin{aligned} \pi(x|\mathbf{x}) &= \int\int \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\log(g_{ij}(\lambda_i))^2}{2\sigma^2}} \pi(\lambda_i, \sigma^2|\mathbf{x}) d\lambda_i d\sigma^2 \\ &\approx \frac{1}{M} \sum_{k=1}^{M} \frac{1}{\sqrt{2\pi\sigma^{2(k)}}}\, e^{-\frac{[x-\log(g_{ij}(\lambda_i^{(k)})]^2}{2\sigma^{2(k)}}} \end{aligned}$$

- $(\lambda_i^{(k)}, \sigma^{2(k)}), k = 1, \ldots, M$ output from `WinBugs`

PKPD Models - Prediction for individual 1

- Average over $(\lambda_i^{(k)}, \sigma^{2(k)})$ for individual 1



- solid = "naive" prediction
- dashed = predictive density

PKPD Models - Prediction -2

- Predictive density for *any* individual at time j is

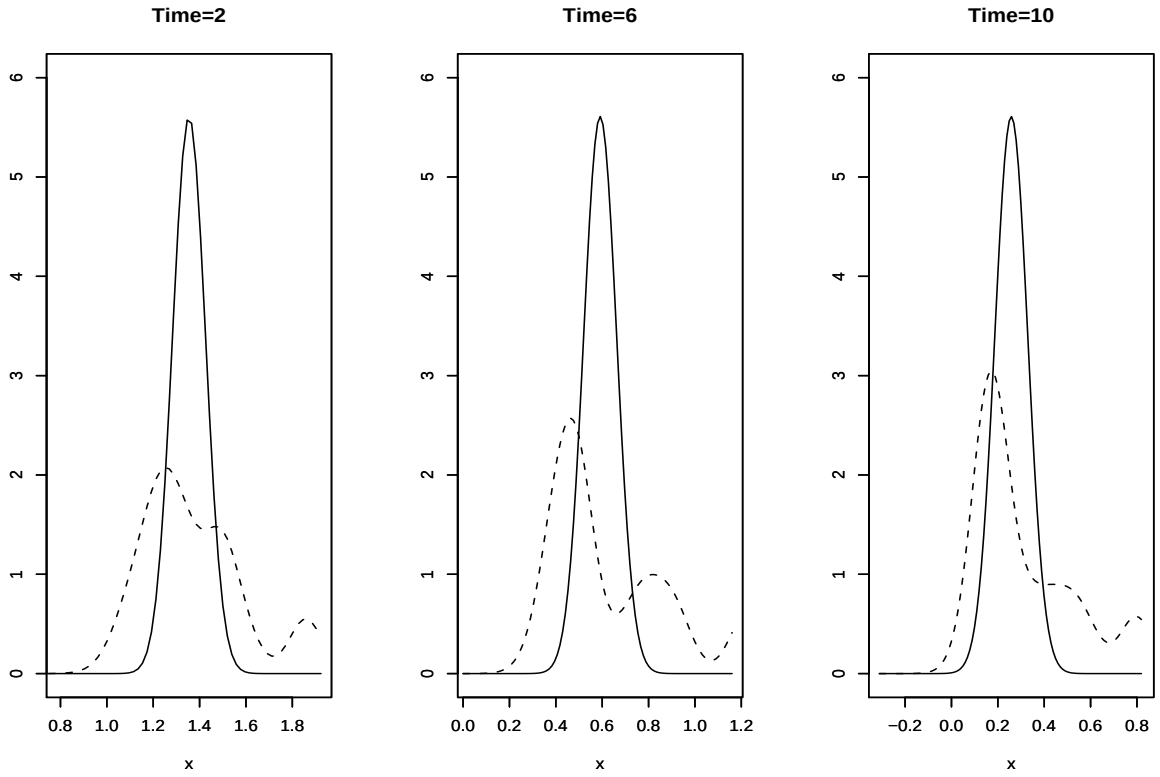
$$\pi(x|\mathbf{x}) = \int\int \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\log(g_{ij}(\lambda))^2}{2\sigma^2}} \pi(\lambda|\mathbf{x}) d\lambda d\sigma^2$$

$$\approx \frac{1}{nM} \sum_{i=1}^{n} \sum_{k=1}^{M} \frac{1}{\sqrt{2\pi\sigma^{2(k)}}} e^{-\frac{[x-\log(g_{ij}(\lambda_i^{(k)})]^2}{2\sigma^{2(k)}}}$$

- $(\lambda_i^{(k)}, \sigma^{2(k)}), k = 1, \ldots, M, \quad i = 1, \ldots n$ output from `WinBugs`
- Increased variability
  - Takes into account variation between individuals
  - Out-of-sample prediction

PKPD Models - Prediction for new individual

- Average over $(\lambda_i^{(k)}, \sigma^{2(k)})$ for all individuals

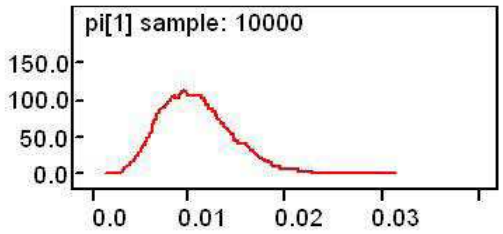


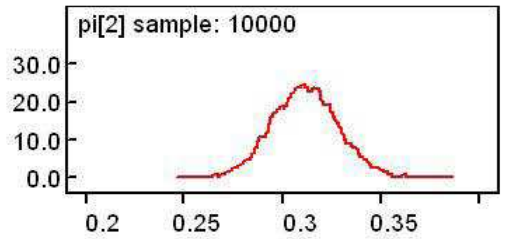
- solid = "naive" prediction
- dashed = predictive density

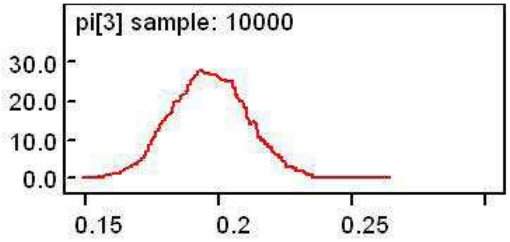
Age Distribution of Chinook Salmon - Winbugs

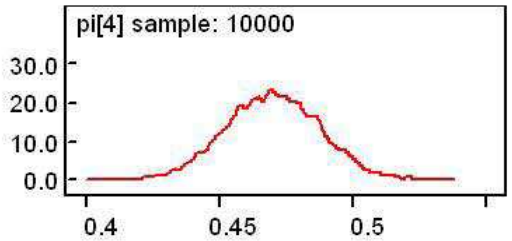
- Recall the model
- Sampling Model
  - $Y_i \sim$ Catagorical$(\mathbf{p})$
  - $X_i \sim$ Normal$(\mu_{y_i}, \sigma^2_{y_i})$
- Prior Specifications
  - $\mathbf{p} \sim$ Dirichlet$(\alpha_3, \ldots, \alpha_7)$
  - $\mu_j \sim$ Normal(μ_{j0}, τ^2_j)
  - $\sigma^2_j \sim$ Inverted Gamma(a, b)

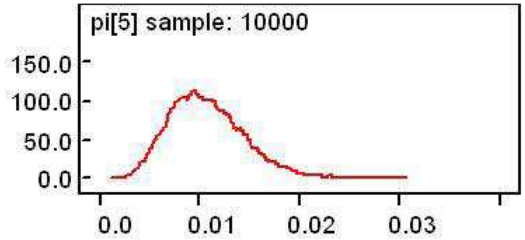
Age Distribution of Chinook Salmon - Winbugs Estimates





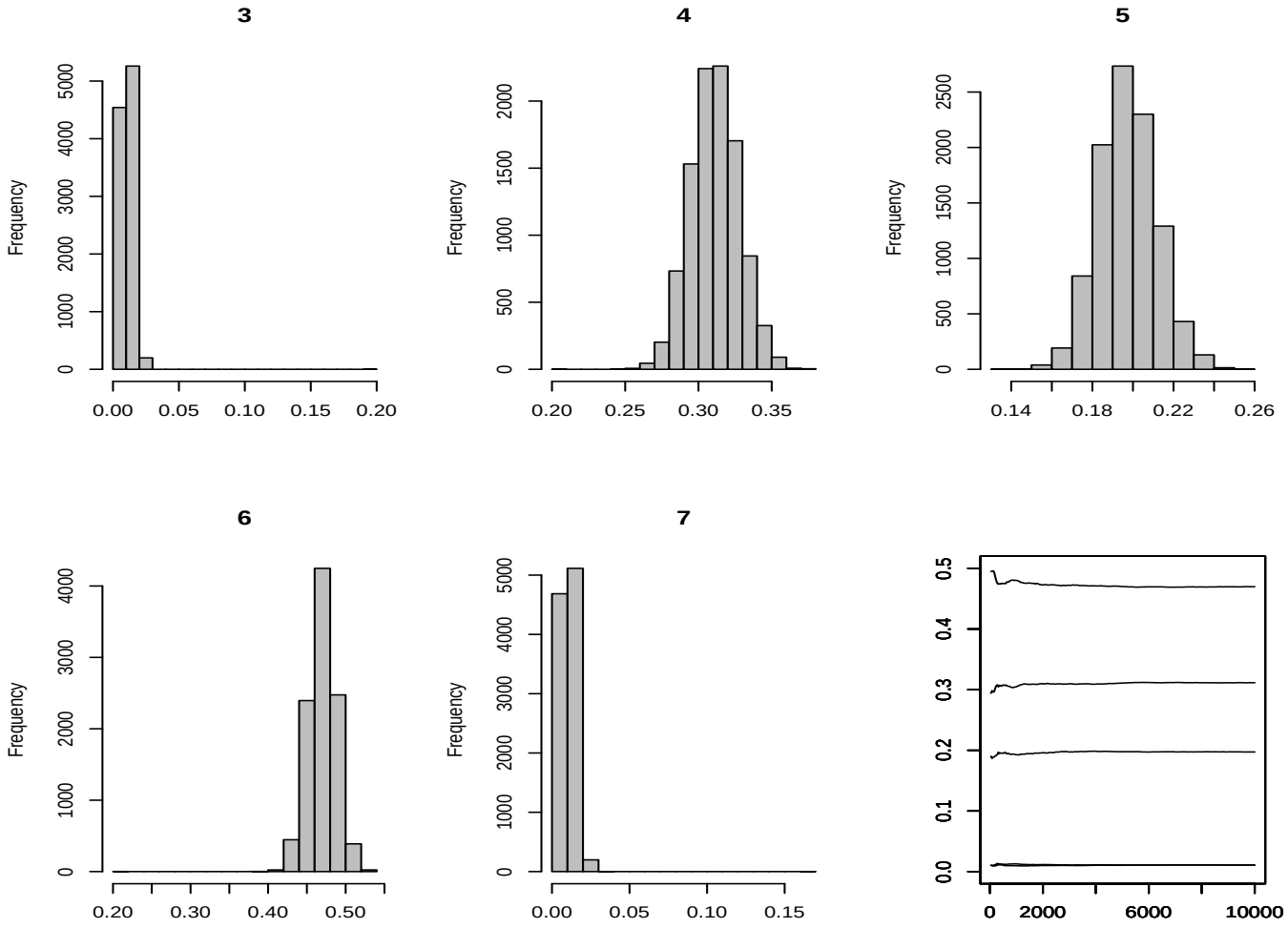






node	mean	sd
pi[1]	0.01083	0.003764
pi[2]	0.3115	0.01657
pi[3]	0.1968	0.01425
pi[4]	0.4702	0.01779
pi[5]	0.01072	0.003719

Age Distribution of Chinook Salmon - R Estimates



Chapter 12: Diagnosing Convergence
Convergence Criteria

- There are three (increasingly stringent) types of convergence
  - Convergence to the Stationary Distribution
  - Convergence of Averages
  - Convergence to iid Sampling

Convergence to the Stationary Distribution

- Minimal requirement
- Theoretically, stationarity is only achieved asymptotically
- Not the major issue. Rather,
  - Speed of exploration of the support of f
  - Degree of correlation between the $\theta^{(t)}$'s.

Convergence of Averages

- Convergence of the empirical average

$$\frac{1}{T}\sum_{t=1}^{T} h(\theta^{(t)}) \to \mathrm{E}_f[h(\theta)]$$

  for an arbitrary function h.

- Most relevant in the implementation of MCMC
  - Convergence related to the *mixing* speed (Brooks and Roberts)

Convergence to iid Sampling

- How close a sample $(\theta_1^{(t)}, \ldots, \theta_n^{(t)})$ is to being iid.
- Can use $subsampling$(or $batch\ sampling$) to reduce correlation between the successive points of the Markov chain.

Multiple Chains

- There are methods involving one chain, and those involving multiple chains.
- By simulating several chains, variability and dependence on the initial values are reduced
- Can control convergence to the stationary distribution by comparing the estimation, using different chains, of quantities of interest.

Multiple Chains - some cautions

- An initial distribution which is too concentrated around a local mode of f does not contribute significantly more than a single chain to the exploration of f
- Slow algorithms, like Gibbs sampling, usually favor single chains
  - A unique chain with MT observations and a slow rate of mixing is more likely to get closer to the stationary distribution than M chains of size T

Overall Cautions

- It is somewhat of an illusion to think we can control the flow of a Markov chain and assess its convergence behavior from a few realizations of this chain.
- The heart of the difficulty is the key problem of statistics, where the uncertainty due to the observations prohibits categorical conclusions and final statements.
- But...We do out best!

Monitoring Convergence of Averages

- Example 12.10: Beta Generator
- The Markov chain $(X^{(t)})$

$$X^{(t+1)} = \begin{cases} Y \sim \mathcal{B}e(\alpha+1,1) & \text{with probability } x^{(t)} \\ x^{(t)} & \text{otherwise} \end{cases}$$

has stationary distribution

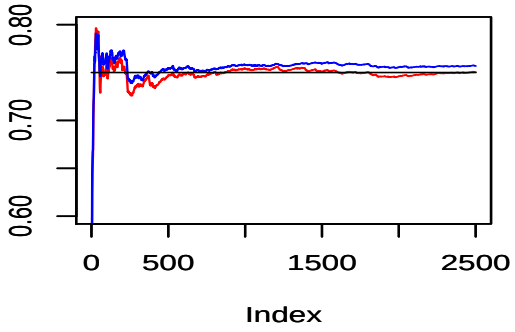
$$f(x) = \alpha x^{\alpha-1} \,,$$

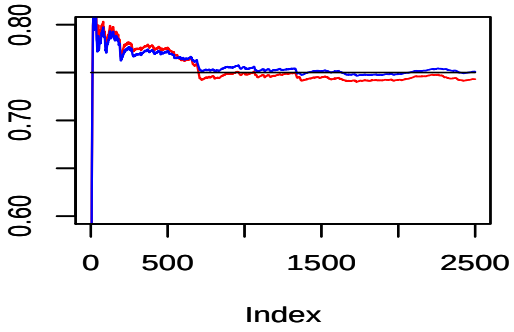
  - Can generate directly
  - Can also use Metropolis, which accepts y with probability $x^{(t)}/y$

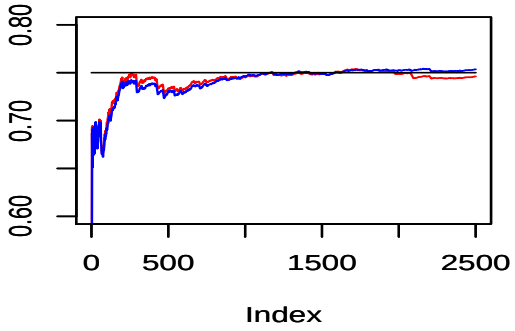
- Note $\mathrm{E}_f(X) = \frac{\alpha}{\alpha+1}$

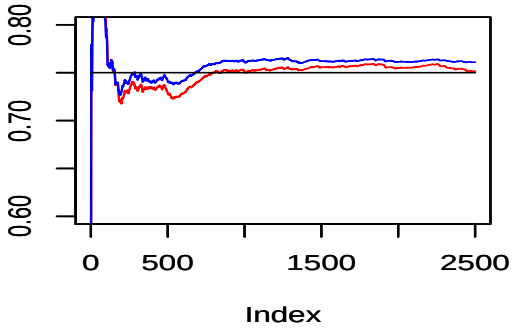
Beta Generator

- This is a very bad chain
- CLT doesn't hold
- Metropolis and Direct









Recall Example 1.2 : Normal Mixtures

- For a mixture of two normal distributions,

$$p\mathcal{N}(\mu, \tau^2) + (1-p)\mathcal{N}(\theta, \sigma^2) \ ,$$

- The likelihood proportional to

$$\prod_{i=1}^{n} \left[p\tau^{-1}\varphi\left(\frac{x_i - \mu}{\tau}\right) + (1-p)\,\sigma^{-1}\,\varphi\left(\frac{x_i - \theta}{\sigma}\right) \right]$$

  containing 2^n terms.

- Standard maximization techniques often fail to find the global maximum because of multimodality of the likelihood function.

Normal Mixture/Gibbs Sampling

- Two components with equal known variance and fixed weights,

$$p\,\mathcal{N}(\mu_1, \sigma^2) + (1-p)\,\mathcal{N}(\mu_2, \sigma^2)\,.$$

- $\mathcal{N}(0, c\sigma^2)$ prior distribution on both means μ_1 and μ_2
- Latent Variable model assumes
  - unobserved component indicators z_i of the observations x_i,

$$P(Z_i = 1) = 1 - P(Z_i = 2) = p,$$

and

$$X_i | Z_i = k \sim \mathcal{N}(\mu_k, \sigma^2)\,.$$

Normal Mixture/Gibbs Sampling-2

- The conditional distributions are

$$\mu_j \sim \mathcal{N}\left(\left(\frac{1}{1/c + n_j}\right)\sum_{z_i=j} x_i, \left(\frac{\sigma^2}{1/c + n_j}\right)\right) ,$$

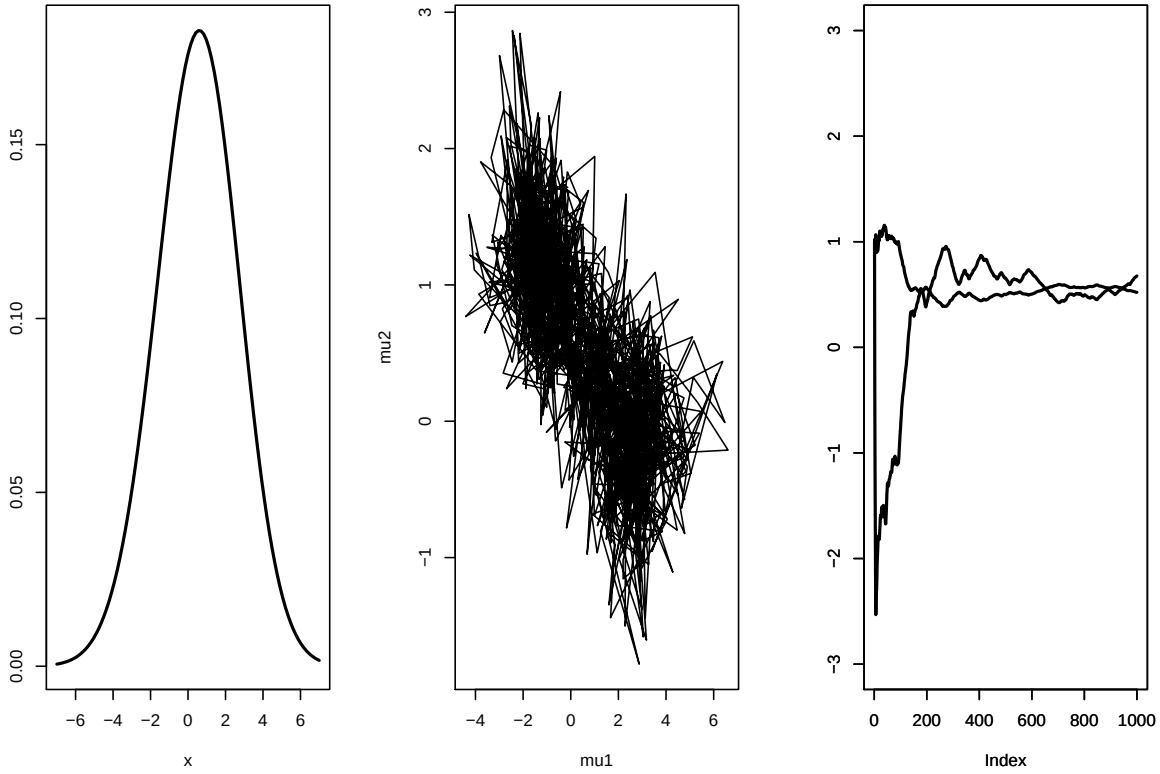
- $\mathbf{z}$ given (μ_1, μ_2) is a product of binomials, with

$$P(Z_i = 1|x_i, \mu_1, \mu_2) = \frac{p\exp\{-(x_i-\mu_1)^2/2\sigma^2\}}{p\exp\{-(x_i-\mu_1)^2/2\sigma^2\} + (1-p)\exp\{-(x_i-\mu_2)^2/2\sigma^2\}} .$$

Normal Mixture/Gibbs Sampling Example

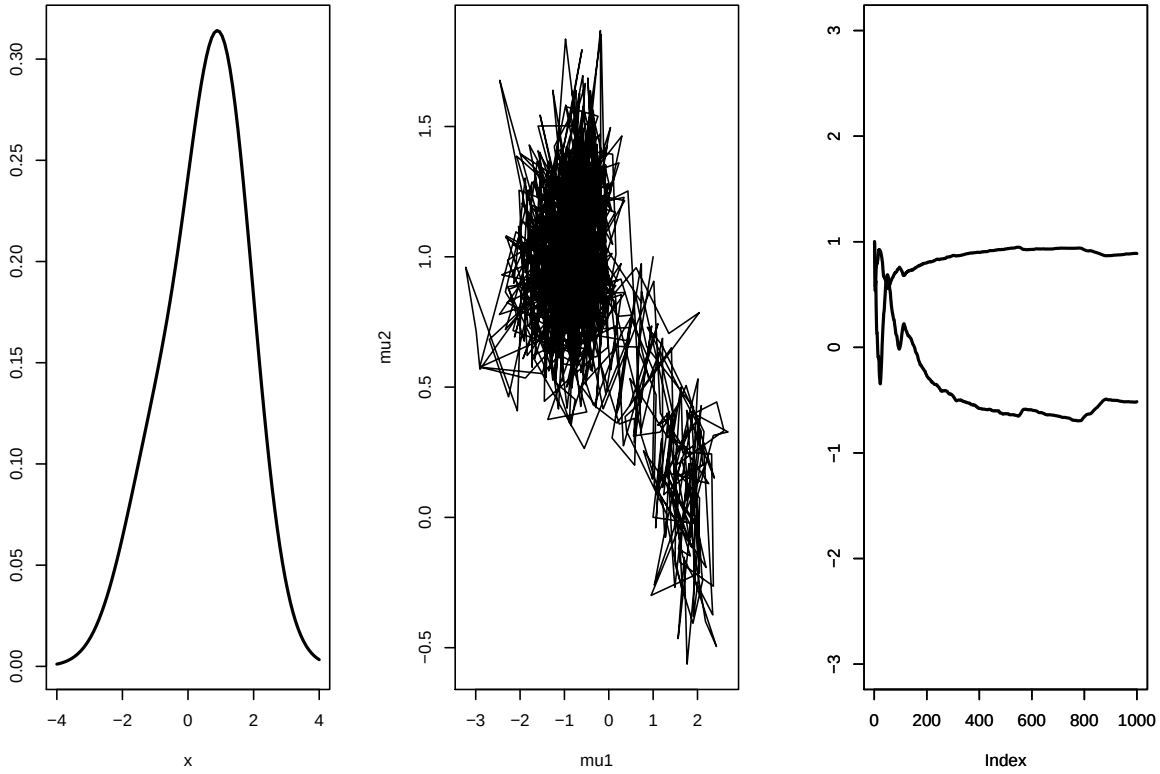
- Take $\mu_1 = -1$, $\mu_2 = 1$, and $p = .25$
- Vary $\sigma = .5, 1, 2$
- Start in one mode
  - `R` program "NormalMixtureGibbs"

Normal Mixture, $\mu_1 = -1$, $\mu_2 = 1$, $\sigma = 2$



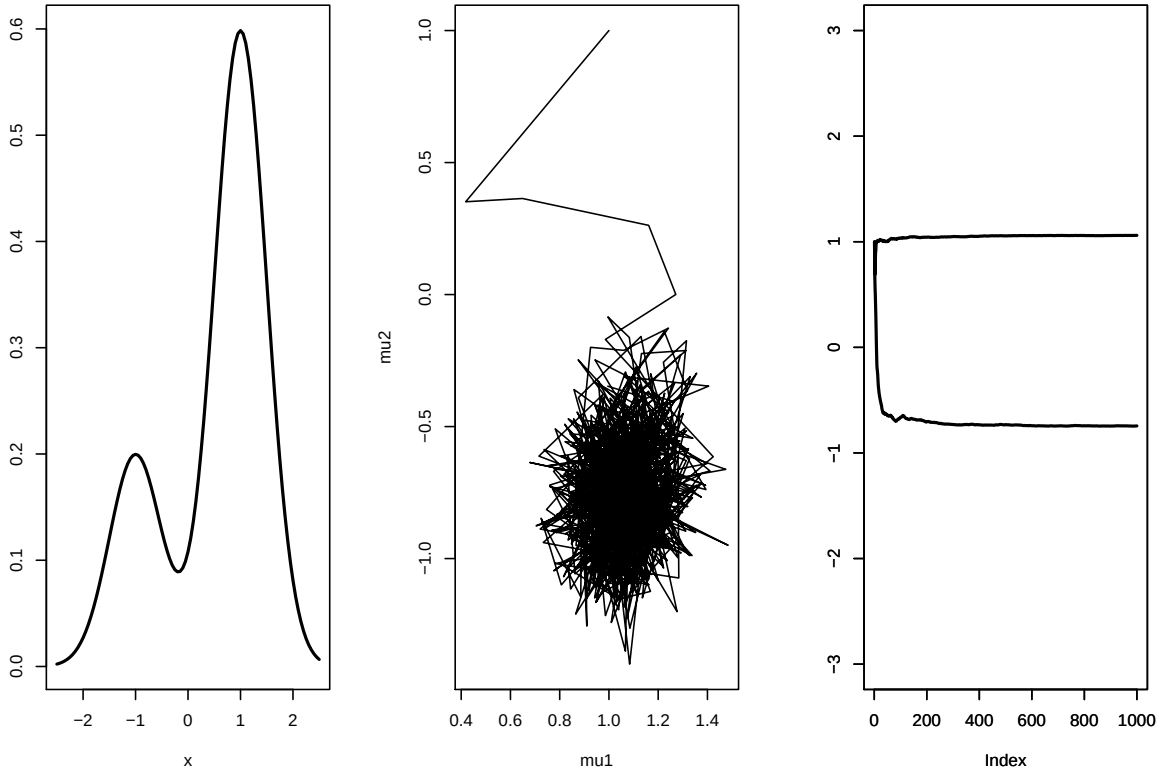
- Can't find underlying means
- Appears close to convergence
- Reasonable representation of density

Normal Mixture, $\mu_1 = -1$, $\mu_2 = 1$, $\sigma = 1$



- Closer to finding underlying means
- Appears close to convergence
- Reasonable representation of density

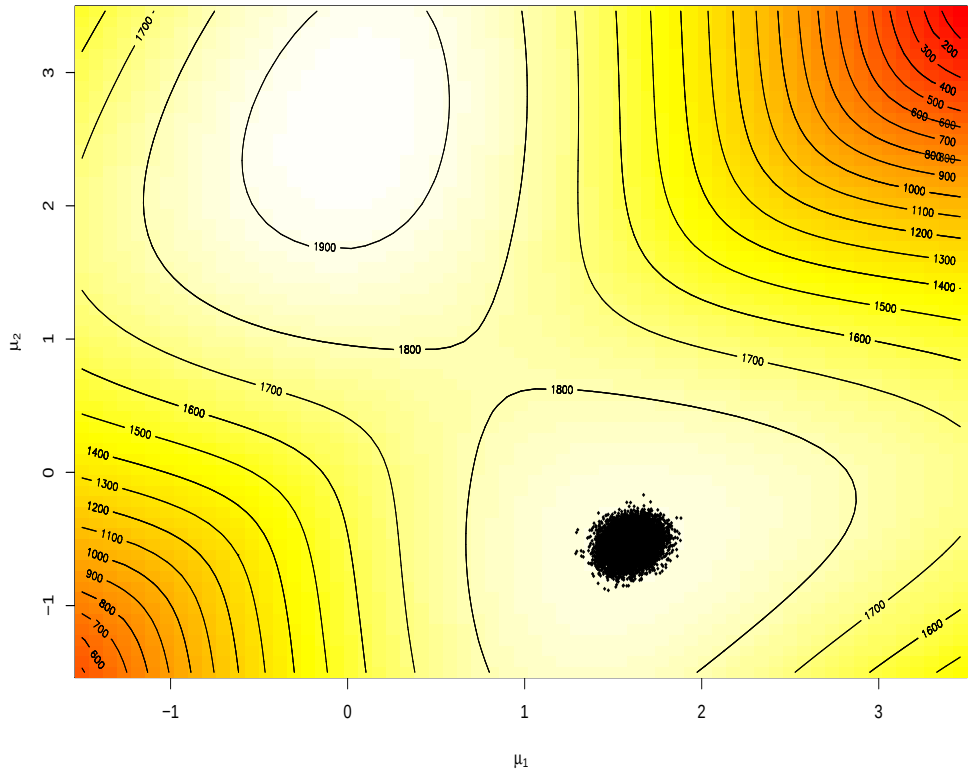
Normal Mixture, $\mu_1 = -1$, $\mu_2 = 1$, $\sigma = .5$



- Finds underlying means
- Appears close to convergence
- Reasonable representation of density

Normal Mixture/Gibbs Sampling-Two Dimensions

- In higher dimensions, the Gibbs sampler may not escape the attraction of the local mode when initialized close to that mode



Normal Mixture/Gibbs Sampling-5

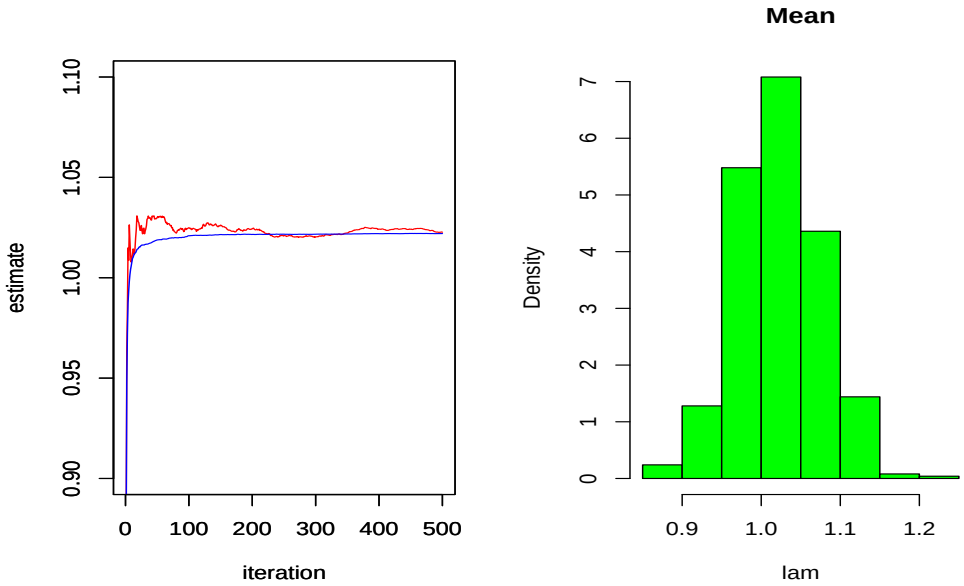
- This problem is common to single chain monitoring methods
  - Difficult to detect the existence of other modes
  - Or of other unexplored regions of the space

Multiple Estimates

- In most cases, the graph of the raw sequence doesn't help in the detection of stationarity or convergence.
- A more helpful indicator is the behavior of the averages in terms of T.
- Can use several convergent estimators of $\mathbb{E}_f[h(\theta)]$ based on the same chain
- Monitor until all estimators coincide

Monitoring Convergence of Averages -Poisson/Gibbs Example

- Two Estimators of Lambda
- Empirical Average or the Conditional Expectation
- Convergence Diagnostic $\longrightarrow$ Both estimators converge



Common Estimates

- The empirical average S_T
- The *conditional* (or Rao–Blackwellized) version of this average

$$S_T^C = \frac{1}{T} \sum_{t=1}^{T} \mathrm{E}[h(\theta)|\eta^{(t)}] \,,$$

- Importance sampling:

$$S_T^P = \sum_{t=1}^{T} w_t \, h(\theta^{(t)}) \,,$$

where $w_t \propto f(\theta^{(t)})/g_t(\theta^{(t)})$ and g_t is the true density used for the simulation. $\theta^{(t)}$.

Example 12.12: Cauchy Posterior

- The hierarchical model

$$\begin{aligned} X_i &\sim \text{Cauchy}(\theta), \quad i = 1, \ldots, 3 \\ \theta &\sim N(0, \sigma^2) \end{aligned}$$

  has posterior distribution

$$\pi(\theta|x_1, x_2, x_3) \propto e^{-\theta^2/2\sigma^2} \prod_{i=1}^{3} \frac{1}{(1 + (\theta - x_i)^2)}$$

- We can use a Gibbs sampler

$$\eta_i|\theta, x_i \sim \mathcal{E}xp\left(\frac{1 + (\theta - x_i)^2}{2}\right),$$

$$\theta|x_1, x_2, x_3, \eta_1, \eta_2, \eta_3 \sim \mathcal{N}\left(\frac{\eta_1 x_1 + \eta_2 x_2 + \eta_3 x_3}{\eta_1 + \eta_2 + \eta_3 + \sigma^{-2}}, \frac{1}{\eta_1 + \eta_2 + \eta_3 + \sigma^{-2}}\right),$$

Example 12.12: Cauchy Posterior -2

- The Gibbs sampler is based on the latent variables η_i, where

$$\int e^{-\frac{1}{2}\eta_i(1+(x_i-\theta)^2)} d\eta_i = \frac{2}{1+(x_i-\theta)^2}$$

- so

$$\eta_i \sim \text{Exponential}\left(\frac{1}{2}(1+(x_i-\theta)^2)\right)$$

- Monitor with three estimates of θ
  - Empirical Average
  - Rao-Blackwellized
  - Importance sample

Monitor with three estimates of θ

- **Empirical Average**

$$\frac{1}{M}\sum_{j=1}^{M}\hat{\theta}^{(j)}$$

- **Rao-Blackwellized**

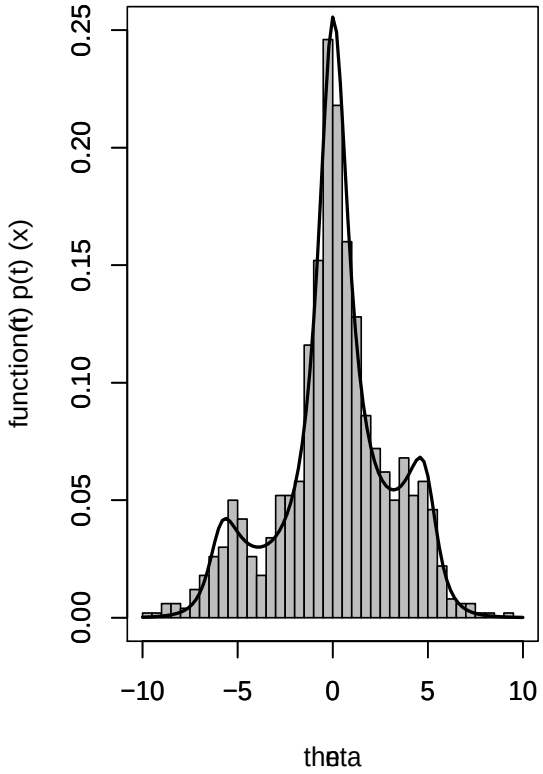
$$\begin{aligned}\mathrm{E}\theta &= C\int \theta e^{-\theta^2/2\sigma^2}\prod_{i=1}^{3}\frac{1}{(1+(\theta-x_i)^2)}d\theta\\ &= C\int\int \theta e^{-\theta^2/2\sigma^2}e^{-\frac{1}{2}\sum_i \eta_i(1+(\theta-x_i)^2)}d\theta d\eta_1 d\eta_2 d\eta_3\end{aligned}$$

  And so

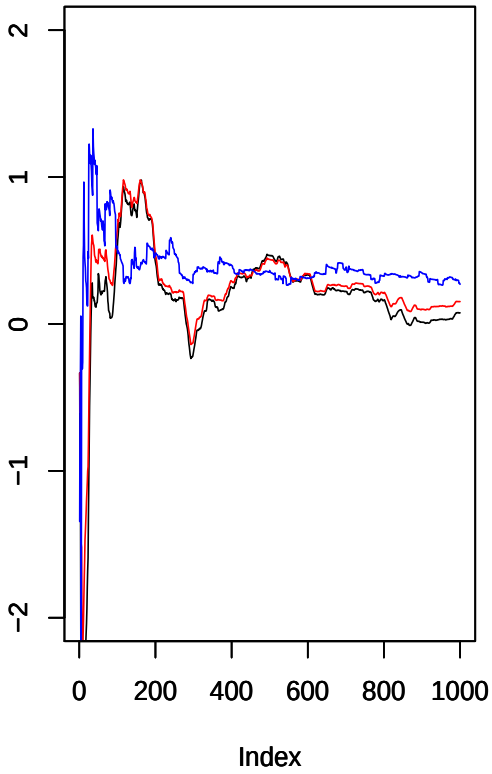
$$\theta|\eta_1,\eta_2,\eta_3 \sim N\left(\frac{\sum_i \eta_i x_i}{\frac{1}{\sigma^2}+\sum_i \eta_i}, \left[\frac{1}{\sigma^2}+\sum_i \eta_i\right]^{-1}\right)$$

- **Importance sampling** with Cauchy candidate

Cauchy Posterior Convergence



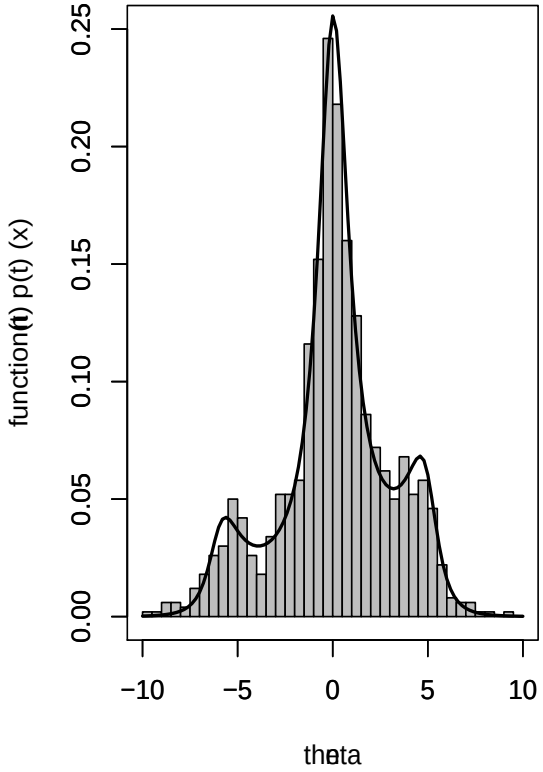
Gibbs Sample Histogram



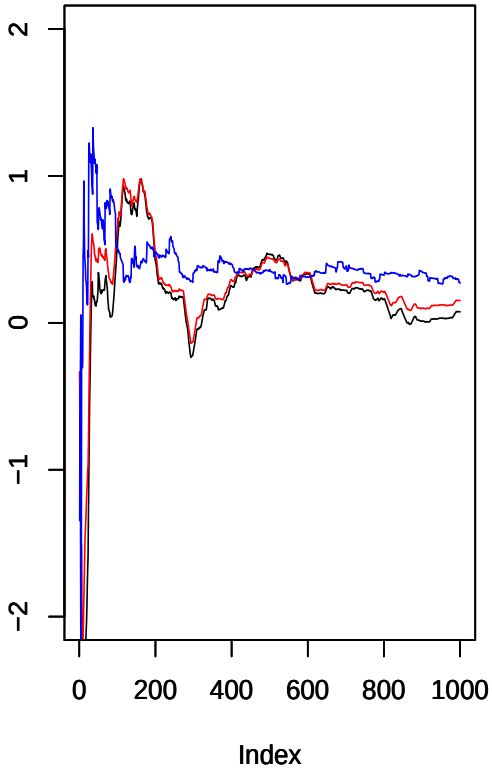
Emp. Avg RB IS

Multiple estimates

- Empirical Average and RB are similar - supports convergence
- IS poor - not yet converged



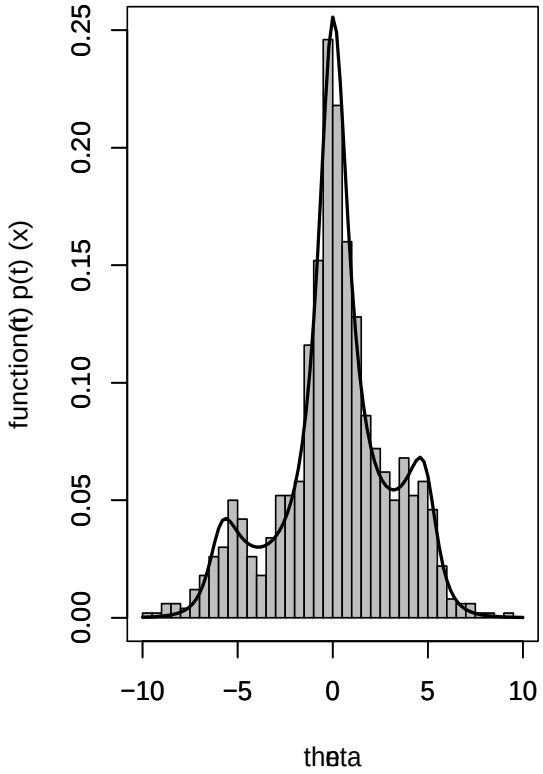
Gibbs Sample Histogram



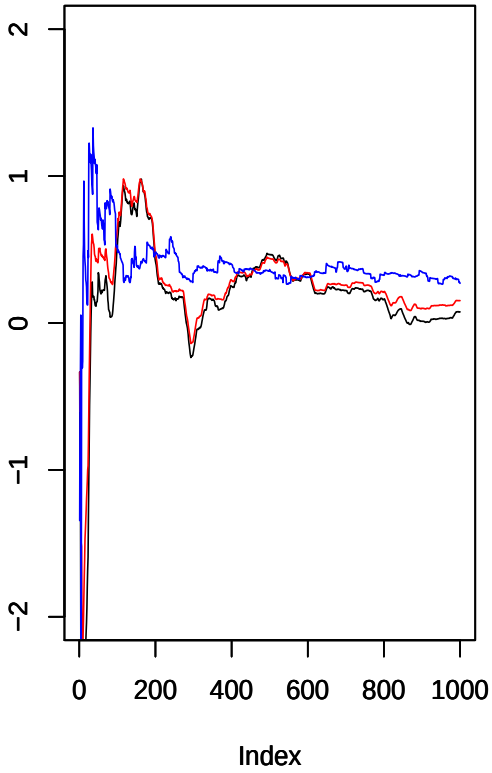
Emp. Avg RB IS

Multiple Estimate-Conclusions

- Limitations:
  - The method does not always apply
  - Intrinsically conservative (since the speed of convergence is determined by the slower estimate)
- Advantage: When applicable, superior diagnostic to single chain



Gibbs Sample Histogram



Emp. Avg RB IS

With and Between Variances

- Gelman/Rubin Criterion
- Criterion based on the difference between a weighted estimator of the variance and the variance of estimators from the different chains
- Need good (dispersed) starting values

With and Between Variances - Some Details

- Generate M chains, estimate $\xi = h(\theta)$
- Calculate

$$\begin{aligned} B_T &= \text{Between Variance} \\ W_T &= \text{Pooled Within Variance} \\ R_T &= \text{Adjusted Ratio of } B_T/W_T \end{aligned}$$

- Convergence when $R_T \longrightarrow 1$

To run Gelman-Rubin in WinBugs

- We need at least two chains in the Model Specification
- Select the B-G-R diag in the Sample Monitor Tool
- Modified by Brooks and Gelman (1998)

Brooks - Gelman -Rubin

- The plot shows
  - Green: Widths of pooled central 80% CI for R_T
  - Blue: Widths of average central 80% CI for R_T
  - Red: R_T
- Want $R_T \longrightarrow 1$
- Look at some examples
- Simple.odc
- PKWinBugs.odc

Gelman/Rubin Comments

- Method has enjoyed wide usage, in particular because of its simplicity and of its connections with the standard tools
- Gelman and Rubin (1992) suggest removing the first half of the simulated sample to reduce the dependence on the initial distribution
- The accurate construction of the initial distribution can be quite delicate and time-consuming.
- The method relies on normal approximations
- But it's not bad!