Markov Chain Monte Carlo Methods

4. Statistical Applications

K B Athreya, Mohan Delampady and T Krishnan

K B Athreya is a Professor at Cornell University. His research interests include mathematical analysis, probability theory and its application and statistics. He enjoys writing for *Resonance*. His spare time is spent listening to Indian classical music.

Mohan Delampady is at the Indian Statistical Institute, Bangalore. His research interests include robustness, nonparametric inference and computing in Bayesian statistics.

T Krishnan is now a full-time Technical Consultant to Systat Software Asia-Pacific Ltd., in Bangalore, where the technical work for the development of the statistical software Systat takes place. His research interests have been in statistical pattern recognition and biostatistics.

1. The Changing Face of Statistics

Classical statistical methodology depended essentially on mathematical tools for its development. In order to make problems analytically tractable, a number of assumptions had to be made on the nature of the data, models formulated had to be fairly simple even if unrealistic, and criteria for inference had to be reasonably manageable. Methods developed from such an approach led to fairly easy computations on mechanical calculators and later, electronic calculators. In the last quarter of the 20th century, due to rapid developments in computing technology, there was no longer any need to constrain methodological developments or applications to limited computing resources. In statistical applications, users are now prepared to analyse huge data sets, formulate highly complex models when called for, formulate criteria not necessarily amenable to analytically tractable and easily computable solutions, etc. The early part of this development saw a host of Monte Carlo simulation exercises, increasing use of randomization techniques, and the emergence of resampling methods such as bootstrap and other cross-validation methods. (See [1],[2]).

During this process, while statistical applications were getting more and more daring in terms of formulating highly nonlinear and otherwise complicated models, Bayesians[1], who till then were limiting their efforts to fairly simple priors such as conjugate priors [2] to achieve analytical tractability, started formulating more complex and realistic priors resulting in analytically in-

[1] those statisticians who use the Bayesian approach

[2] those families of priors p such that the posteriors p' are also of the same family (for instance, beta prior for the Binomial(n, p) problem)

tractable situations { for instance, posterior distributions whose moments are not easily worked out analytically. (For an introduction to Bayesian statistics and for explanations of terms used here, see [3].) One reason for this is that the integration needed to ¯nd the normalizing constant was not tractable, thus making the density incompletely speci¯ed. In some situations, numerical integration solved the problem reasonably well. However, simulation-based methods such as Monte Carlo integration often give more e±cient solutions to these problems, especially in higher dimensions. But under certain circumstances, direct generation of random samples for Monte Carlo integration is not possible and more complex Monte Carlo methods such as Markov Chain Monte Carlo (MCMC) methods are needed.

Such cases arise for instance, when the target distribution (for us, the posterior distribution $\pi(\mu|x)$) is incompletely speci¯ed or speci¯ed indirectly, say, a joint distribution being speci¯ed in terms of several conditional and marginal distributions. In many of these situations, so long as the target distribution is uniquely de¯ned from the given speci¯cations, it is possible to adopt an iterative random sampling (Monte Carlo) procedure, which at the point of convergence will deliver a random draw from the target distribution. These iterative Monte Carlo procedures typically generate a random sequence with the Markov property such that the Markov chain is ergodic with a limiting distribution coinciding with the target distribution. There is a whole family of such iterative procedures collectively called Markov Chain Monte Carlo (MCMC) procedures, di®erent procedures being suitable for di®erent situations.

In the earlier parts of this series of articles the focus was on the theoretical aspects of the MCMC methods. In this part, we discuss a few statistical applications.

1. Simple Monte Carlo, *Resonance*, Vol.8, No.4, p.17, 2003.
2. The Markov Chain Case, *Resonance*, Vol.8, No.7, p.63, 2003.
3. Statistical Concepts, *Resonance*, Vol.8, No.10, p.30, 2003.

Keywords
Bayesian approach, posterior distribution, Dirichlet prior, Metropolis–Hastings algorithm, rejection sampling, Gibbs sampler, proposal density, Rao–Blackwellisation, binomial, multinomial, Gamma, uniform.

2. Application of Metropolis–Hastings Algorithm

We shall illustrate applications of the Metropolis–Hastings (M–H) algorithm described in Part 2 of this series of articles to the generation of a random sample from a gamma distribution and from the posterior distribution of Weibull parameters in a Bayesian context. But before that we discuss some preliminaries.

Inverse CDF Method:

It is well known ([4] pp.54–55) that if the distribution function $F(x) \stackrel{\text{def}}{=} P(X \leq x)$ of a random variable X is continuous and strictly increasing then $U = F(X) \sim U[0,1]$, meaning that U is uniformly distributed in the interval $[0,1]$. In this case, F^{-1} is easily defined. Thus a sample U from $U[0,1]$ can be used to get a random sample $X = F^{-1}(U)$ with distribution function F. For example,

$$F(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 - e^{-\lambda x} & \text{for } x \geq 0 \end{cases};$$

where $0 < \lambda < 1$, is the distribution function of an $\exp(\lambda)$ random variable. Since $F^{-1}(u) = -\frac{\ln(1-u)}{\lambda}$, for $0 < u < 1$, if U is a uniform [0, 1] random variable, then $-\frac{\ln(1-U)}{\lambda}$ will have $\exp(\lambda)$ distribution. Note that since $1 - U$ is also a uniform [0, 1] random variable, $-\frac{\ln U}{\lambda}$ will also work.

Next consider the problem of generating a sample from the gamma(α, λ) distribution.

Definition: A gamma distribution with parameters (α, λ), written as $G(\alpha, \lambda)$, is a continuous probability distribution on $(0, \infty)$ with probability density

$$f_{\alpha,\lambda}(x) = \frac{\lambda^{\alpha}}{\Gamma(\alpha)} e^{-\lambda x} x^{\alpha-1};$$

where $\Gamma(\alpha) = \int_0^{\infty} e^{-x} x^{\alpha-1} dx$ for $0 < \alpha < \infty$ is the well-known gamma function.

If $U_1; U_2; : : : ; U_k$ are independent identically distributed (i.i.d.) $U[0; 1]$ variables then it can be shown (see [4] p.248) that

$$X_{k;\lambda} = -\frac{1}{\lambda} \sum_{i=1}^{k} \ln U_i$$

has a $G(k; \lambda)$ distribution. Thus if $\alpha = k$ is a positive integer, then this problem has an easy solution. If not, we need to use other methods. One such method is Rejection Sampling.

Rejection Sampling

Example 1:

Let a random variable X have density specified as $f(x) = \frac{3}{2}(1 - x^2); 0 \leq x \leq 1$. The CDF is 0, for $x < 0$; is $\frac{3}{2}(x - \frac{x^3}{3})$, for $0 \leq x \leq 1$; and 1, for $x > 1$. The CDF is a cubic in x and the inverse CDF method involves solving a cubic equation, which involves some effort. However, the following Rejection Sampling method is quite easy to implement: Choose a random sample Y from uniform [0, 1]. Accept Y as a random sample from the density f with probability $(1 - Y^2)$; else reject it and repeat the procedure until a Y is accepted. Keep repeating this procedure until the desired number of samples are accepted. In order to decide to accept Y with probability $(1 - Y^2)$, select a random sample U from uniform [0, 1] and if $U \leq (1 - Y^2)$, accept.

For Rejection Sampling, an envelope or a majorizing density function g_Y of another random variable Y is needed such that $cg_Y(x) \geq f_X(x)$ $\forall x$, for some constant $c > 0$. Further, for the method to work, it should be easy to draw random samples from Y. cg_Y is called the majorizing function. g_Y is also called the proposal density. The algorithm is:

1. Draw Y from $g_Y(:)$:
2. Draw U from $U(0; 1)$.

3. If $U \cdot f_X(Y)=cg_Y(Y)$ then accept Y as the desired realization;
   else return to Step 1.
   Repeat until one Y is accepted.

Repeat this to select the desired number of samples. For distributions with ¯nite support and bounded density, g can always be chosen as uniform.

In Example 1, g is the uniform [0, 1] density and $c = \frac{3}{2}$.

Now, the probability of drawing an accepted value in the interval $(x; x + dx)$ is proportional to

$$g(x)dx\frac{f(x)}{cg(x)} = \frac{1}{c}f(x)dx:$$

Thus the accepted sample is from f(x) and accepted proportion of sampled values is $\frac{1}{c}$.

Example 2: Drawing from $G(®; 1)$

We discuss Rejection Sampling from $G(®; ¯)$ when ¯ = 1, when ® is not an integer. Let us use the notation [®] for the integer part of ®, i.e., if $k \cdot ® < k + 1$, k an integer, then [®] = k. It can be checked that the density of $G([®]; \frac{[®]}{®})$ is a majorizing density with

$$c = \frac{¡([®])}{¡(®)}\frac{(®)^®}{[®]^{[®]}}e^{¡(®¡[®])}:$$

The method thus is:

(a) Draw Y from $G([®]; \frac{[®]}{®})$.

(b) Accept Y with probability

$$p = \left(\frac{ey\exp(¡y=®)}{®}\right)^{®¡[®]}$$

which is the ratio of the density of $G(®; 1)$ to c times the density of $G([®]; \frac{[®]}{®})$. [See [5], pp.94, 242.]

M–H Algorithm for Gamma (α, 1) Sampling

As discussed in Part II of this series, the general Metropolis–Hastings (M–H) algorithm is as follows: The target density $f(x)$ is specified (possibly without the constant term) and a 'suitable' proposal conditional density $h(y|x)$ is chosen such that for all x, $f(y) > 0$ if and only if $h(y|x) > 0$. Then the algorithm is as follows:

Start at $t = 0$ with a value x_0 in the support of the target distribution, i.e., a possible value from that distribution. At step t:

(a) Draw Y_t from $h(y|X_t)$.

(b) Let

$$X_{(t+1)} = \begin{cases} Y_t & \text{with probability } p_t \\ X_t & \text{otherwise,} \end{cases}$$

where $p_t = \min\{\frac{h(X_t|Y_t)f(Y_t)}{h(Y_t|X_t)f(X_t)}, 1\}$.

(c) Set $t = t + 1$ and go to step (a).

Run this until $t = n$ (a suitably chosen large value). Then X_n has a density that is close to the target density $f(x)$.

A special case of the M–H algorithm is the so-called Independence M–H algorithm, where in step (a), the distribution does not depend on the current value of x_t, whereas in the general form of the M–H algorithm, it does. In the general case, the proposal distribution is a conditional distribution $Y_t|X_t$, which is illustrated in Example 3 below.

For the above problem with the $G(\alpha; 1)$ distribution, an Independent M–H algorithm with the proposal distribution the same as the majorising density of the Rejection Sampling is as follows:

Start at $t = 0$ with some value $x_0, 0 < x_0 < 1$. At step

t, (a) Draw Y_t from G([®]; $\frac{[®]}{®}$).

(b) Let

$$X_{(t+1)} = \begin{cases} Y_t & \text{with probability } p_t \\ x_t & \text{otherwise,} \end{cases}$$

where

$$p_t = \min\{1, \frac{Y_t}{x_t} \exp(\frac{x_t - Y_t}{®})^{®-[®]}\}; 1g:$$

(c) Set t=t+1 and go to step (a).

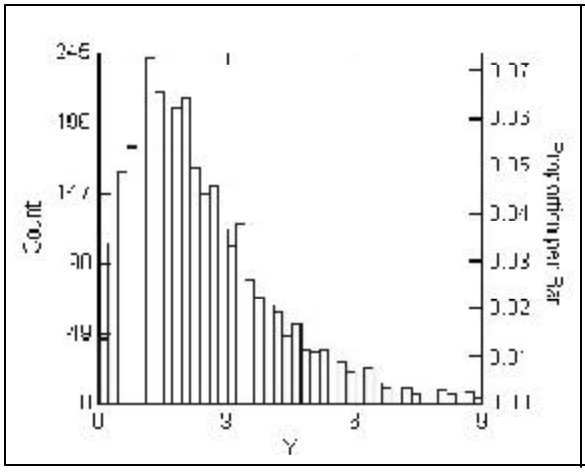
Run this until t=n (a suitably chosen large value).

The step `with probability' can be implemented by using a draw from U[0; 1]. Denoting the target density by f and the proposal density by h(x), p_t is obtained as

$$\min\{\frac{f(Y_t)h(Y_t)}{f(x_t)h(x_t)}, 1g:$$

Notice that in this expression, the constants in both f and h cancel out; however, for implementing step (a), full knowledge of the density of Y_t is required. Thus the M{H algorithm can be carried out to draw from a distribution for which the constant is not speci¯ed. In the gamma distribution example, the Rejection Sampling method is easier to implement than the M{H algorithm, since the full density of G(®; 1) is anyway known. However, we illustrate the M{H algorithm with this example.

We generated 3336 samples by this method where n was chosen to be 1000. The histogram of the sampled values and the actual gamma(2.43, 1) density are given in Figures 1 and 2 respectively. Descriptive statistics based on these samples are given in Table 1. From these it is clear that the M{H algorithm provides a good way of generating random samples whose distribution is close to that of an incompletely speci¯ed distribution.



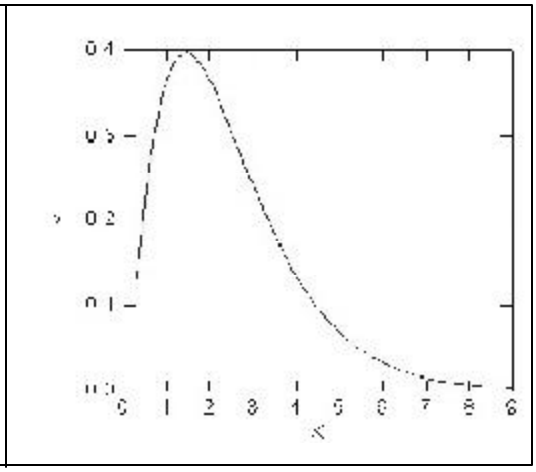


Figure 1 (left). Histogram of gamma (2.43, 1) density generated using Metropolis–Hastings algorithm.
Figure 2 (right). Theoretical Gamma (2.43, 1) density.

Statistic	From simulation	Theoretical
No. of cases	3336	{
Minimum	0.02359	0
Maximum	13.37880	1
Median	2.02555	2.1060
Mean	2.42552	2.43
SD	1.72762	1.5588
Variance	2.98465	2.43

Table 1. Descriptive statistics from data generated for gamma (2.43, 1) density by Metropolis–Hastings algorithm.

Here we used a simple ¯xed-length stopping rule for deciding when to stop a Markov chain and accept the last value as a sample from the stationary distribution. However, the issue of convergence diagnostics and stopping rules for MCMC are more complex and research on these issues is still going on. See [5] (Example 6.3.3, pp.242{243) for a discussion of the gamma example described above.

Example 3: Bayesian Inference for Weibull Distribution

A Weibull distribution has the following density form:

$$f(x) \acute{} f(x|\alpha;\acute{}) / \alpha\acute{} x^{\alpha i\ 1} e^{i\ \acute{}\ x^{\alpha}};\ 0 < x < 1$$

(where / means `proportional to') with parameters (α, γ). Consider the prior distribution for the parameter (α, γ)

$$\pi(\alpha, \gamma) \propto e^{-\alpha\gamma - \tau} e^{-\nu\gamma}$$

and i.i.d. observations $x_1, x_2, \ldots, x_N$ from $f(x)$. The posterior distribution of (α, γ) given the data $(x_1, x_2, \ldots, x_N)$ has density

$$g(\alpha, \gamma) = \pi(\alpha, \gamma | x_1, x_2, \ldots, x_N)$$

$$\propto f(x_1) f(x_2) \ldots f(x_N) \pi(\alpha, \gamma).$$

To get a sample from the posterior density one may use the M–H algorithm with proposal distribution

$$q(\alpha', \gamma' | \alpha, \gamma) = \frac{1}{\alpha\gamma} e^{-\frac{\alpha'}{\alpha} - \frac{\gamma'}{\gamma}},$$

that is a product of two independent exponential distributions with means α, γ. (See [6], Example 6.3.2, p. 305.)

Thus the M–H algorithm for this problem is:

1. Data $x_1, x_2, \ldots, x_N$ with $N = 10$ are given as 0.645, 0.647, 0.422, 0.899, 0.228, 1.083, 0.450, 0.985, 1.106, 0.701.
2. Prior distribution parameters τ, ν are given as 1.5 and 0.5 respectively.
3. Initial values $\alpha^{(0)}, \gamma^{(0)}$ are given as 1.5 and 1 respectively.
4. Generate M–H Markov chains of length $n = 250$ and retain the n^{th} element of the chain. Call the chain $(\alpha^{(t)}, \gamma^{(t)})$; $t = 0, 1, 2, \ldots$ Iterations proceed as follows from $(\alpha^{(t)}, \gamma^{(t)})$ to $(\alpha^{(t+1)}, \gamma^{(t+1)})$:

   (a) Generate a random draw from

$$q(\alpha', \gamma' | \alpha^{(t)}, \gamma^{(t)}) = \frac{1}{\alpha^{(t)}\gamma^{(t)}} e^{-\frac{\alpha'}{\alpha^{(t)}} - \frac{\gamma'}{\gamma^{(t)}}}.$$

Statistic	ALPHA	ETA
No. of cases	100	100
Minimum	4.142530	0.875560
Maximum	14.489841	4.358850
Median	7.917805	1.911510
Mean	8.234183	2.023315
SD	2.212684	0.730947
Variance	4.895969	0.534283
Correlation	¡ 0:3206	

Table 2. Descriptive statistics of the posterior distribution of a, h generated by M–H algorithm.

(b) Compute

$$\frac{1}{2}(\alpha^0, \gamma^0, \alpha^{(t)}, \gamma^{(t)}) =$$

$$\min\left\{\frac{g(\alpha^0, \gamma^0)}{g(\alpha^{(t)}, \gamma^{(t)})} \frac{q(\alpha^{(t)}, \gamma^{(t)} | \alpha^0, \gamma^0)}{q(\alpha^0, \gamma^0 | \alpha^{(t)}, \gamma^{(t)})}, 1\right\}.$$

(c) Draw R from uniform (0,1).

(d) If $R \cdot \frac{1}{2}(\alpha^0, \gamma^0, \alpha^{(t)}, \gamma^{(t)})$, then
let $\alpha^{(t+1)}, \gamma^{(t+1)} = \alpha^0, \gamma^0$
else let $\alpha^{(t+1)}, \gamma^{(t+1)} = \alpha^{(t)}, \gamma^{(t)}$.

(e) Stop when $t = n = 250$. Output $(\alpha^{(n)}, \gamma^{(n)})$.

Repeat this $m = 100$ times.

Some descriptive statistics of this joint posterior distribution of (α, γ) are given in Table 2 and Figure 3.

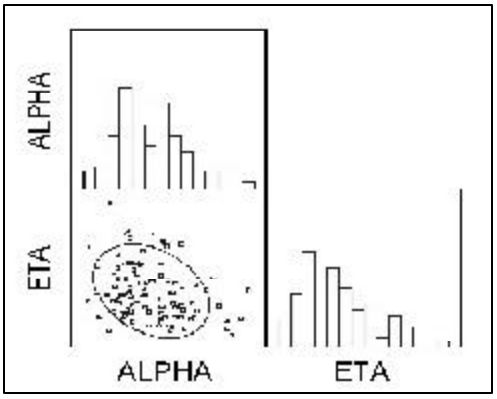


Figure 3. Histograms and scatter plot of Weibull posterior of parameters a and h generated by M–H algorithm.

There are issues regarding the appropriate choice of n, choice of the proposal density, etc., which we shall not address in this article. It must be noted that when the M{H chain is run upto n = 250, the value $(\alpha^{(n)}, \gamma^{(n)})$ has a distribution that is only close to but not exactly equal to the target distribution, i.e., the posterior distribution.

3. Application of Gibbs Sampling

Example 4: Gibbs Sampling for Multinomial

Data	Model
$n_O = 176$	r^2
$n_A = 182$	$p^2 + 2pr$
$n_B = 60$	$q^2 + 2qr$
$n_{AB} = 17$	$2pq$
Total	435

Consider the following multinomial model with data. Here the 4-cell multinomial probability vector is a function of the parameters p, q, r with $p + q + r = 1$. One may wish to formulate a Dirichlet prior for p, q, r. But it will not be conjugate to the 4-cell multinomial probability (likelihood in terms of p, q, r) from the data and this makes it difficult to work out the posterior distribution of p, q, r. Although no data are missing in the real sense of the term, it is profitable to split each of the n_A and n_B cells into two: n_A into n_{AA}, n_{AO} with corresponding probabilities $p^2, 2pr$ and n_B into n_{BB}, n_{BO} with corresponding probabilities $q^2, 2qr$. Consider the 6-cell multinomial problem as a complete problem with n_{AA}, n_{BB} as 'missing' data.

Of course, $p + q + r = 1$. Let $n = n_O + n_A + n_B + n_{AB}$. Let us denote the observed data by $\underline{n} = (n_O, n_A, n_B, n_{AB})$.

Suppose one wants to do Bayesian estimation of p, q, r with a Dirichlet prior with parameters α, β, δ with the 'incomplete' observed data $\underline{n}$.

The likelihood upto a multiplicative constant is:

$$L(p, q, r) = r^{2n_O}(p^2 + 2pr)^{n_A}(q^2 + 2qr)^{n_B}(pq)^{n_{AB}}.$$

The posterior of (p, q, r) given $\underline{n}$, is proportional to

$$r^{2n_O + \delta - 1}(p^2 + 2pr)^{n_A}(q^2 + 2qr)^{n_B}(p)^{n_{AB} + \alpha - 1}(q)^{n_{AB} + \beta - 1}.$$

It is not easy to deal with this and work out mean,

median and such useful posterior integrals. This is the sort of situation where Gibbs sampler is useful.

Let $n_A = n_{AA} + n_{AO}$, $n_B = n_{BB} + n_{BO}$. Let us write n_{OO} for n_O for the sake of elegant and consistent notation. It is easy to see that if we have the 'complete' data, i.e., observations $\underline{n} = (n_{OO}; n_{AA}; n_{AO}; n_{BB}; n_{BO}; n_{AB})$, then the likelihood is, upto a multiplicative constant

$$(p^2)^{n_{AA}} (q^2)^{n_{BB}} (r^2)^{n_{OO}} (2pq)^{n_{AB}} (2qr)^{n_{BO}} (2pr)^{n_{AO}}$$

$$= p^{n_A^+} q^{n_B^+} r^{n_O^+};$$

where

$$n_A^+ = 2(n_{AA} + \frac{1}{2}n_{AB} + \frac{1}{2}n_{AO})$$

$$n_B^+ = 2(\frac{1}{2}n_{AB} + n_{BB} + \frac{1}{2}n_{BO})$$

$$n_O^+ = 2(\frac{1}{2}n_{AO} + \frac{1}{2}n_{BO} + n_{OO}:)$$

Thus the posterior distribution for $(p; q; r)$ can be seen to be Dirichlet with parameters $n_A^+ + \alpha - 1; n_B^+ + \beta - 1; n_O^+ + \gamma - 1$, when the prior is Dirichlet with parameters $(\alpha; \beta; \gamma)$. This simple solution to this 'complete' problem will now be exploited in the Gibbs Sampler.

It follows from the model and the assumed priors that the conditional distributions of $(n_{AA}; n_{BB})$ given $\underline{n}$ and $(p; q; r)$ is that of two independent binomials:

$$(n_{AA} | \underline{n}; p; q; r) \sim \text{Binomial}(n_A; \frac{p^2}{p^2 + 2pr}) \qquad (A1)$$

$$(n_{BB} | \underline{n}; p; q; r) \sim \text{Binomial}(n_B; \frac{q^2}{q^2 + 2qr}): \qquad (A2)$$

Also the posterior distribution of $(p; q; r)$ given (observed data, missing data), i.e., that of

$$(p; q; r | \underline{n}; n_{AA}; n_{BB}) \text{ is Dirichlet}$$

$$(n_A^+ + \alpha - 1; n_B^+ + \beta - 1; n_O^+ + \gamma - 1): \qquad (B)$$

Gibbs sampling for this problem is straightforward { starting from initial estimates for $p; q; r$, use random draws from (A1, A2) and (B) in turn until `convergence' to get a random sample from the joint distribution of $(p; q; r; n_{AA}; n_{BB})|n$. Suppose one has N such independent samples $(p^{(i)}; q^{(i)}, r^{(i)}; n_{AA}^{(i)}; n_{BB}^{(i)})$. Recall from Part III, the idea and technique of Rao-Blackwellisation for variance reduction. Estimates of posterior mean of $p; q; r$ are obtained by Rao{ Blackwellisation, namely by

$$\frac{1}{N}\sum_{i=1}^{N} E(p^{(i)}; q^{(i)}; r^{(i)}|n; n_{AA}^{(i)} n_{BB}^{(i)})$$

$$= \frac{1}{N}\sum_{i=1}^{N} (\alpha + n_A^+; \beta + n_B^+; \gamma + n_O^+)/(\alpha + \beta + \gamma + n):$$

We ran such a Gibbs sampler for this problem with N = 10,000 and the results are presented in Table 3 and Figure 4.

The values of $\alpha; \beta; \gamma$ were 2, 2 and 2.

Table 3. Descriptive Statistics from Gibbs samples of mean and Rao–Blackwellised estimates of p, q, r.

The table and the histograms give estimates of $p; q; r$ by the method of sample means (called pmean, etc.) and by Rao{ Blackwellisation (called praobwell, etc). They show how useful Gibbs sampling method is. They also

Statistic	PMEAN	QMEAN	RMEAN	PRB	QRB	RRB
No. of cases	10000	10000	10000	10000	10000	10000
Minimum	0.20149	0.05579	0.60135	0.24116	0.08460	0.61160
Maximum	0.31637	0.12704	0.71819	0.29356	0.10922	0.66590
Median	0.25306	0.08988	0.65678	0.26418	0.09573	0.63994
Mean	0.25306	0.09022	0.65671	0.26430	0.09588	0.63982
SD	0.01561	0.00970	0.01437	0.00683	0.00295	0.00719
Variance	0.00024	0.00009	0.00021	0.00005	0.00001	0.00005
MLE for p, q, r respectively				0.26444	0.09317	0.64239

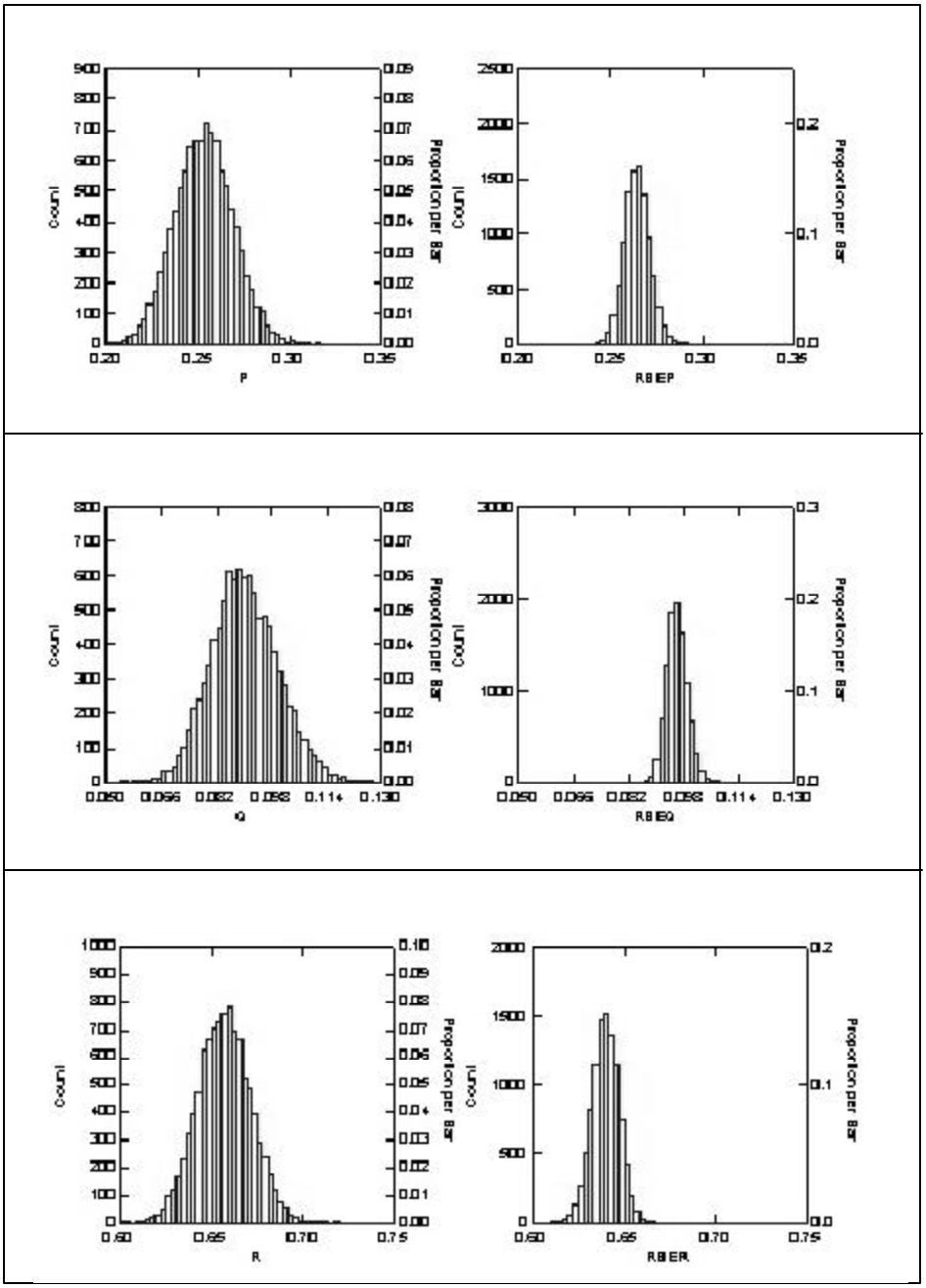


Figure 4. Histogram of mean estimates of p, q, r and Rao-Blackwellised Estimates of p, q, r (called RBEP, RBEQ, RBER).

show how much more e± cient the Rao{ Blackwellised (PRB, QRB, RRB) estimates are.

Concluding Remarks

In this four-part article we have attempted to give an introduction to Markov chain Monte Carlo methods, and through the examples tried to give some idea of the computational intensity with which these methods have to be applied. The theory and methodology of MCMC are still evolving, especially in respect of diagnostics for convergence of the Markov chains generated by the algorithms. We have not touched upon these rather dif- ¯cult but important and crucial aspects of the MCMC methodology. We hope that the collection of articles in Resonance starting from Kunte [7,8] on Statistical Computing, and articles on bootstrap and Bayesian statistics that have followed, culminating in Chakraborty's [9,10] and these MCMC articles, give the reader the ° avour of modern computer-intensive statistical methods.

Suggested Reading

[1] B Efron and R J Tibshirani, *An Introduction to the Bootstrap*, London: Chapman & Hall, 1993.

[2] P Hall and A Sarkar, Bootstrap methods in Statistics: Resampling, *Resonance*, Vol. 5, No.9, pp. 41-48, 2000.

[3] M Delampady and T Krishnan, Bayesian Statistics, *Resonance*, Vol.7, No.4, pp. 27-38, 2002.

[4] G Casella and R L Berger, *Statistical Inference*, Second Edition. Pacific Grove, Ca: Duxbury, 2001.

[5] C P Robert, The Bayesian Choice, New York: Springer-Verlag, 2001.

[6] C P Robert and G Casella, *Monte Carlo Statistical Methods*, New York: Springer-Verlag, 1999.

[7] Arnab Chakraborty, Markov Chain Monte Carlo: Examples, *Resonance*, Vol.7, No.3, pp.25-34, 2002.

[8] Arnab Chakraborty, Markov Chain Monte Carlo: Examples, *Resonance*, Vol.7, No.5, pp.66-75, 2002.

[9] Sudhkar Kunte, Statistical Computing: 1. Understanding Randomness and Random Numbers, *Resonance*, Vol.4, No.10, pp.16-21, 1999.

[10] Sudhkar Kunte, Statistical Computing: 2. Technique of Statistical Simulation, *Resonance*, Vol.5, No.4, pp.18-27, 2000.

Address for Correspondence
K B Athreya
School of ORIE
Rhodes Hall
Cornell University, Ithaca
New York 14853, USA

Mohan Delampady
Indian Statistical Institute
8th Mile, Mysore Road
Bangalore 560 059, India.

T Krishnan
Systat Software Asia-Pacific Ltd.
Floor 5, 'C' Tower
Golden Enclave, Airport Road
Bangalore 560 017, India.