Method of Moments Using Monte Carlo Simulation

Andrew GELMAN*

We present a computational approach to the method of moments using Monte Carlo simulation. Simple algebraic identities are used so that all computations can be performed directly using simulation draws and computation of the derivative of the log-likelihood. We present a simple implementation using the Newton–Raphson algorithm with the understanding that other optimization methods may be used in more complicated problems. The method can be applied to families of distributions with unknown normalizing constants and can be extended to least squares fitting in the case that the number of moments observed exceeds the number of parameters in the model. The method can be further generalized to allow "moments" that are any function of data and parameters, including as a special case maximum likelihood for models with unknown normalizing constants or missing data. In addition to being used for estimation, our method may be useful for setting the parameters of a Bayes prior distribution by specifying moments of a distribution using prior information. We present two examples—specification of a multivariate prior distribution in a constrained-parameter family and estimation of parameters in an image model. The former example, used for an application in pharmacokinetics, motivated this work. This work is similar to Ruppert's method in stochastic approximation, combines Monte Carlo simulation and the Newton–Raphson algorithm as in Penttinen, uses computational ideas and importance sampling identities of Gelfand and Carlin, Geyer, and Geyer and Thompson developed for Monte Carlo maximum likelihood, and has some similarities to the maximum likelihood methods of Wei and Tanner.

Key Words: Bayesian computation; Compositional data; Estimation; Importance sampling; Least squares; Maximum likelihood; Missing data; Newton–Raphson; Prior distribution; Stochastic approximation; Unnormalized densities.

1. INTRODUCTION

The method of moments—estimating the parameters of a probability distribution by matching theoretical moments to specified values—can be useful in statistical estimation or in constructing a Bayes prior distribution. In problems of estimation, the method of moments can be preferable to other approaches such as maximum likelihood if the family of probability models is in doubt and one wishes to make sure the chosen model accurately fits certain aspects of the data that can be expressed as moments. A different

*Assistant Professor, Department of Statistics, University of California, Berkeley, CA 94720, email: gelman@stat.berkeley.edu

Journal of Computational and Graphical Statistics, Volume 4, Number 1, Pages 36–54

application is in Bayesian analysis, in which it is often convenient to set the parameters of an informative prior distribution by specifying a few moments, such as prior means and variances.

Unfortunately, in complicated models, moments are generally expressed as intractable integrals, in which case matching moments requires the solution to an integral equation. If it is possible to draw Monte Carlo simulations from the distributions in the parametric family, then one can estimate the theoretical moments using sample moments of the simulation and then solve for an approximate moments estimate by trial and error or by using a stochastic approximation method as in Robbins and Monro (1951). Such a search can be difficult in the case of a vector parameter, however.

In this article we present an iterative approach to matching moments using a numerical equation-solving algorithm (Newton–Raphson) applied to Monte Carlo estimates of moments and their derivatives, as in Ruppert (1985). Our method can be applied to parametric families with unknown normalizing constants. We also generalize to a least squares fit in the case that moments are specified with error, when the number of moments given exceeds the number of parameters in the distribution. We discuss how our method of moments computations can be viewed as an extension of the Monte Carlo maximum likelihood methods of Gelfand and Carlin (in press), Geyer (in press a, b), Geyer and Thompson (1992), Moyeed and Baddeley (1991), Penttinen (1984), and Wei and Tanner (1990). We present two examples—a specification of a multivariate prior distribution in a constrained-parameter family, which motivated this work, and an estimation of the two parameters of an image model by matching two moments to data. We conclude with a brief discussion of the practical uses of moments estimation in applied statistics and the practical difficulties that arise when implementing Newton–Raphson.

The most important contribution of this article is the generalization to the method of moments of Monte Carlo methods that have been used for maximum likelihood. Throughout we present the procedures using Newton–Raphson, which is convenient to implement in this context, with the understanding that the Monte Carlo estimates can be applied to more complicated optimization (or root-finding) algorithms as well. We envision the family of methods described in this article as a useful addition to a modeler's toolkit, not as a stand-alone general approach to statistical computation.

2. THE BASIC METHOD

2.1 NOTATION AND MATHEMATICAL FORMULATION

Suppose we have a family of distributions, $p(x|\theta)$, and we wish to estimate the d-dimensional parameter vector θ by matching a d-dimensional vector of moments, $\mu(\theta) = \mathrm{E}(h(x)|\theta)$, to a fixed vector μ_0. In the usual formulation of the method of moments, μ_0 is the vector of sample moments. For the purposes of this article, however, we do not require μ_0 to be specified with reference to any observed data.

If $\mu(\theta)$ can be expressed analytically in closed form, we can obtain the moments estimate $\hat{\theta}$ using the Newton–Raphson method, as follows. Start with a guessed value,

θ_1. Then, for $t = 1, 2, \ldots$, update the guess to

$$\theta_{t+1} = \theta_t + [\mu'(\theta_t)]^{-1}(\mu_0 - \mu(\theta_t)), \tag{2.1}$$

where μ' is the matrix of derivatives of $\mu(\theta)$ with respect to θ.

Here we are concerned with problems for which $\mu(\theta)$ cannot be computed in closed form; instead, we can estimate it, for any given value of θ, by simulation of N draws of x from the distribution $p(x|\theta)$:

$$\hat{\mu}(\theta) = \frac{1}{N}\sum_{i=1}^{N} h(x_i). \tag{2.2}$$

The draws of x may be obtained by direct sampling from $p(x|\theta)$ if possible, or else by an iterative algorithm such as the algorithm of Metropolis et al. (1953) or the Gibbs sampler (e.g., Gelfand and Smith 1990), which is always possible as long as the density function $p(x|\theta)$ can be computed. The derivative matrix $\mu'(\theta)$ can be computed at any point θ by using the following formula:

$$\begin{aligned} \mu'(\theta) &= \frac{d}{d\theta}\mathrm{E}(h(x)|\theta) \\ &= \int h(x)\frac{d}{d\theta}p(x|\theta)dx \qquad (2.3) \\ &= \mathrm{E}(h(x)U(x,\theta)^T), \qquad (2.4) \end{aligned}$$

where M^T denotes the transpose of matrix M. Equation (2.4) holds assuming it is possible to differentiate under the integral sign, and assuming the limits of integration in (2.3) do not depend on θ. We use the notation

$$U(x,\theta) = \frac{d}{d\theta}\log p(x|\theta)$$

by analogy to the potential function in statistical physics. The practical advantage of (2.4) is that it does not require knowledge of the distribution of $h(x)$ or the analytic form of h. The obvious Monte Carlo estimator of μ' is then

$$\hat{\mu}'(\theta) = \frac{1}{N}\sum_{i=1}^{N} h(x_i)U(x_i,\theta)^T, \tag{2.5}$$

where all vectors are interpreted as column vectors and thus the expression inside the sum is an outer product matrix. If one can compute $p(x|\theta)$, then one should have no difficulty computing $U(x,\theta)$; for standard models, the log-density function can easily be differentiated analytically.

We can now construct a Monte Carlo algorithm in which, for $t = 1, 2, \ldots$, we first draw N_t values of x from the distribution $p(x|\theta_t)$, given the current guess θ_t, then perform one Newton–Raphson optimization step using (2.2) and (2.5). For the algorithm to converge, N_t must be increased as t increases, so that the Monte Carlo error approaches zero as the Newton–Raphson algorithm approaches convergence. Ruppert (1985) discussed the convergence of a similar algorithm that numerically estimates μ'

without using the analytic derivative in formula (2.4). In addition to the inconvenience of requiring a schedule of increasing N (which should ideally be determined by estimating the Monte Carlo accuracy at each step of the simulation), our algorithm (and Ruppert's) is inefficient because it requires a new set of draws of x at each step of the iteration. In addition, if the x_i's are themselves drawn by an iterative method such as that of Metropolis et al. (1953), the nested looping is awkward and may require repeated checks for convergence of the simulations.

If the support of the distribution $p(x|\theta)$ depends on θ, as in censored data, an additional term must be added to (2.3) to account for the derivative of the limits of integration with respect to θ.

2.2 USING IMPORTANCE SAMPLING TO USE THE SIMULATIONS MORE EFFICIENTLY

We can do better and make full use of all the sample draws by using importance sampling in the manner of Geyer (in press) and Geyer and Thompson (1992). Suppose we have drawn samples $x_1, \ldots, x_N$ from a density $g(x)$ that may differ from $p(x|\theta)$; the rule of importance sampling yields the estimates,

$$\begin{aligned} \hat{\mu}(\theta) &= \frac{1}{N}\sum_{i=1}^{N} h(x_i)\frac{p(x_i|\theta)}{g(x_i)} \\ \hat{\mu}'(\theta) &= \frac{1}{N}\sum_{i=1}^{N} h(x_i)U(x_i,\theta)^T\frac{p(x_i|\theta)}{g(x_i)}. \end{aligned} \tag{2.6}$$

These estimates will be useful as long as $g(x)$ is close to $p(x|\theta)$, as is generally understood in importance sampling (see, e.g., Hammersley and Handscomb 1964).

We define an *optimization step* as a single step of the optimization algorithm (e.g., Newton–Raphson) based on a single set of simulations; using the importance sampling estimates of $\hat{\mu}(\theta)$ and $\hat{\mu}'(\theta)$, one can perform any number of successive optimization steps using a fixed set of simulations. We define a *simulation step* as a new set of simulation draws of x from the current guess of θ. We can now improve our earlier algorithm by applying a suggestion used for maximum likelihood calculations by Gelfand and Carlin (in press) and Wei and Tanner (1990): performing several optimization steps after each simulation step. However, a single simulation step (that is, just sampling a large number N draws x_i from the initial guess θ_1) will generally not be enough, because if the initial guess is far from the actual moments estimate, the importance ratios in the estimates (2.6) will become too variable as θ_t moves from its initial guess. However, a few simulation steps, each followed by several optimization steps, should bring the estimate θ_t close enough to the goal that further importance ratios will be well behaved. At this point one can sample a large number N draws from $p(x|\theta_t)$ and iterate Newton–Raphson to approximate convergence. If there is a concern that the moments estimate is not unique, then one can run the algorithm several times starting in different regions of parameter space and see if they converge to the same value of θ.

In practice, one can monitor the simulations and increase N when the variability from step to step gets larger than the systematic movement toward convergence. Similarly, one

can run optimization steps after each simulation draw, stopping to draw more simulations when the optimization steps have moved so far from the last simulation distribution that the importance ratios are ill behaved (e.g., see Kong 1992 for discussion of how to diagnose poor importance ratios).

3. GENERALIZING TO DENSITIES WITH UNKNOWN NORMALIZING CONSTANTS

It is common in complicated statistical models for a family of probability distributions to be specified up to an unknown normalizing constant that depends on the model parameters; that is, we can write the density function as,

$$p(x|\theta) = \frac{q(x|\theta)}{z(\theta)}, \tag{3.1}$$

where the *unnormalized density function* $q(x|\theta)$ can be easily computed, but the normalizing function $z(\theta)$ can be expressed only as an intractable integral. We can easily extend our importance sampling method to unnormalized densities, using identities that were applied to similar problems for maximum likelihood estimation by Geyer (in press) and Geyer and Thompson (1992). Here we derive the analogous identities for the purposes of moments estimation. The vector of theoretical moments can be expressed as

$$\begin{aligned} \mu(\theta) &= \int h(x)\frac{q(x|\theta)}{z(\theta)}dx \\ &= \frac{\mathrm{E}_g\left(h(x)\frac{q(x|\theta)}{g(x)}\right)}{\mathrm{E}_g\left(\frac{q(x|\theta)}{g(x)}\right)}, \end{aligned} \tag{3.2}$$

where E_g is the expectation under the density proportional to $g(x)$, and g is also allowed to be an unnormalized density. The natural Monte Carlo extension of (2.6) is then

$$\hat{\mu}(\theta) = \frac{\frac{1}{N}\sum_{i=1}^{N} h(x_i)\frac{q(x_i|\theta)}{g(x_i)}}{\frac{1}{N}\sum_{i=1}^{N}\frac{q(x_i|\theta)}{g(x_i)}}.$$

Estimating $\mu'(\theta)$ requires one further step. Define

$$\begin{aligned} U_q(x,\theta) &= \frac{d}{d\theta}\log q(x|\theta) \\ &= U(x,\theta) + \frac{d}{d\theta}\log z(\theta). \end{aligned}$$

We seek a computing formula for $\mu'(\theta)$ that can be written in terms of U_q, which is easily computable, rather than U, which depends on the unknown function z. The key step is the well-known identity,

$$\begin{aligned} \frac{d}{d\theta}\log z(\theta) &= \int \frac{1}{z(\theta)}\frac{d}{d\theta}q(x|\theta)dx \\ &= \int \left(\frac{d}{d\theta}\log q(x|\theta)\right)\frac{q(x|\theta)}{z(\theta)}dx \\ &= \mathrm{E}(U_q(x,\theta)), \end{aligned} \tag{3.3}$$

with the expectation taken over $p(x|\theta)$, once again assuming that one can differentiate under the integral sign and that the range of integration does not depend on θ. We can then write,

$$\begin{aligned}\mu'(\theta) &= \mathrm{E}\left(h(x)U(x,\theta)^T\right) \\ &= \mathrm{E}\left(h(x)U_q(x,\theta)^T\right) - \mathrm{E}(h(x)\left(\frac{d}{d\theta}\log z(\theta)\right)^T\right) \\ &= \mathrm{E}\left(h(x)U_q(x,\theta)^T\right) - \mathrm{E}(h(x))\mathrm{E}(U_q(x,\theta))^T. \end{aligned} \tag{3.4}$$

We can now apply the importance sampling identity to transform into an expression based on simulations from the density proportional to $g(x)$:

$$\mu'(\theta) = \frac{\mathrm{E}_g\left(h(x)U_q(x,\theta)^T\frac{q(x|\theta)}{g(x)}\right)}{\mathrm{E}_g\left(\frac{q(x|\theta)}{g(x)}\right)} - \frac{\mathrm{E}_g\left(h(x)\frac{q(x|\theta)}{g(x)}\right)}{\mathrm{E}_g\left(\frac{q(x|\theta)}{g(x)}\right)}\frac{\mathrm{E}_g\left(U_q(x,\theta)\frac{q(x|\theta)}{g(x)}\right)^T}{\mathrm{E}_g\left(\frac{q(x|\theta)}{g(x)}\right)},$$

which can also be obtained directly by differentiating (3.2). The estimate $\hat{\mu}'$ is obtained by replacing E_g by $1/N\sum_{i=1}^N$ everywhere and x by x_i in the previous expression.

4. LEAST SQUARES FIT WHEN THERE ARE MORE MOMENTS THAN PARAMETERS

4.1 LEAST SQUARES FIT TO MOMENTS

Suppose now the problem is overdetermined, with more moments specified than parameters in the model, and we would like the θ that gives the best least-squares fit, minimizing $||\mu_0 - \mu(\theta)||^2$. The normal equations are $\mu'(\theta)(\mu_0 - \mu(\theta)) = 0$, which we can again solve by Newton–Raphson, using iterative least squares. Starting out at a guess θ_1, for $t = 1, 2, \ldots$, the updated guess is

$$\theta_{t+1} = \theta_t + [\text{least squares regression of } (\mu_0 - \mu(\theta_t)) \text{ on the matrix } \mu'(\theta_t)].$$

Ideally, this iteration converges to a (local) least squares fit. The least squares updating is identical to the step (2.1) if the number of moments equals the number of parameters, in which case $\mu'(\theta)$ is a square matrix.

One can apply the Monte Carlo method as before, using the estimates $\hat{\mu}(\theta)$ and $\hat{\mu}'(\theta)$ from the previous sections and converging to an approximate least squares fit by simulating a large number N of draws once the estimate θ_t is close to convergence.

In addition, one can trivially extend to a weighted least squares fit by replacing the linear regression described previously by a weighted linear regression. In many problems, a reasonable weighting can be chosen based on the natural scales of the functions $h(x)$. We illustrate with an example in Section 6.1 in which the moments are rescaled so that an unweighted least squares fit is desirable.

4.2 LEAST SQUARES FIT TO A FUNCTION OF THE MOMENTS

We can further extend the method to find a least squares fit to a multivariate function of the moments, $f(\mu)$, thus minimizing $||f(\mu_0) - f(\mu(\theta))||^2$. Least squares fitting to a

function of the moments is almost the same as described, except that the normal equations become $f'(\mu)\mu'(\theta)(f(\mu_0) - f(\mu(\theta))) = 0$. The optimization step becomes

$$\theta_{t+1} = \theta_t + [\text{least squares regression of } \quad (f(\mu_0) - f(\mu(\theta_t))) \quad \text{on the matrix } \quad f'(\mu(\theta_t))\mu'(\theta_t)],$$

using the estimates $\hat{\mu}(\theta_t)$ and $\hat{\mu}'(\theta_t)$.

An example of such a problem is fitting the mean and standard deviation of a scalar random variable x to their theoretical values. In a two-parameter model we can match the mean and standard deviation exactly by matching the first two moments; that is, $h(x) = (x, x^2)$, and $\mu = (\mathrm{E}(x), \mathrm{E}(x^2))$. If we wish to match both moments in a single-parameter model, however, we cannot fit them exactly, and it may make more sense to fit the mean and standard deviation—not the mean and $\mathrm{E}(x^2)$—by least squares. In the described notation,

$$f(\mu_1, \mu_2) = \left(\mu_1, \sqrt{\mu_2 - \mu_1^2}\right),$$

and

$$f'(\mu) = \begin{pmatrix} 1 & 0 \\ \frac{-\mu_1}{\sqrt{\mu_1\mu_2 - \mu_1^2}} & \frac{1/2}{\sqrt{\mu_1\mu_2 - \mu_1^2}} \end{pmatrix}.$$

At each step of the iteration, this matrix should be computed based on the latest Monte Carlo estimate, $\hat{\mu}(\theta_t)$.

5. RELATION TO MAXIMUM LIKELIHOOD

The concept of method of moments can be generalized to include maximum likelihood as a special case. The Monte Carlo Newton–Raphson method for maximum likelihood was proposed by Penttinen (1984) and was referred to by Ripley (1988, pp. 64–65). In this section we extend the method presented in Sections 2 and 3 to rederive the maximum likelihood (or maximum posterior density) method of Gelfand and Carlin (in press) and Moyeed and Baddeley (1991), which are related to the Monte Carlo EM algorithm of Wei and Tanner (1990). The key is to interpret the equation setting the derivative of the log-likelihood to zero as an equation specifying a vector of moments. None of the maximum likelihood methods discussed in this section are new; we present them here to illustrate the generality of the method of moments paradigm for these problems.

5.1 MAXIMUM LIKELIHOOD WITH UNKNOWN NORMALIZING CONSTANTS

We first address the problem of maximum likelihood given unnormalized density functions, using the notation (3.1) for unnormalized densities. Using the normalized and unnormalized density formulation, the likelihood equation becomes

$$\frac{d}{d\theta} \log q(x_0|\theta) - \frac{d}{d\theta} \log z(\theta) = 0,$$

where x_0 is the observed data set. Using (3.3), this becomes

$$U_q(x_0, \theta) = \mathrm{E}(U_q(x, \theta)).$$

We can interpret this as a method of moments equation, with the left and right sides corresponding to the observed and theoretical moments, respectively:

$$\begin{aligned} \mu_0(\theta) &= h(x_0, \theta) = U_q(x_0, \theta) \\ \mu(\theta) &= \mathrm{E}(h(x, \theta)) = \mathrm{E}(U_q(x, \theta)). \end{aligned} \tag{5.1}$$

Strictly speaking, these expressions are not moments because the function h and the "observed moments" μ_0 depend on the unknown parameter θ as well as x (unless the model is an exponential family, in which case the generalized moments are in fact moments—the expectations of the sufficient statistics under the model). However, we can generalize our method by solving for the equation, $\mu(\theta) - \mu_0(\theta) = 0$. As before,

$$\hat{\mu}(\theta) = \frac{\frac{1}{N}\sum_{i=1}^{N} U_q(x_i, \theta)\frac{q(x_i|\theta)}{g(x_i)}}{\frac{1}{N}\sum_{i=1}^{N} \frac{q(x_i|\theta)}{g(x_i)}}.$$

Now that h depends on θ, however, we must also differentiate with respect to h to determine μ'. The new expression, replacing (2.4) and (3.4), is,

$$\begin{aligned} \mu'(\theta) &= \frac{d}{d\theta}\mathrm{E}(h(x, \theta)|\theta) \\ &= \mathrm{E}\left(\frac{d}{d\theta}h(x, \theta)\right) + \mathrm{E}\left(h(x, \theta)U(x, \theta)^T\right) \\ &= \mathrm{E}\left(\frac{d}{d\theta}h(x, \theta)\right) + \mathrm{E}\left(h(x, \theta)U_q(x, \theta)^T\right) \\ &\quad -\mathrm{E}(h(x, \theta))\mathrm{E}(U_q(x, \theta))^T. \end{aligned} \tag{5.2}$$

For the maximum likelihood problem, expression (5.2) has a nice symmetric form:

$$\mu'(\theta) = \mathrm{E}(U_{2q}(x, \theta)) + \mathrm{E}\left(U_q(x, \theta)U_q(x, \theta)^T\right) - \mathrm{E}(U_q(x, \theta))\mathrm{E}(U_q(x, \theta))^T, \tag{5.3}$$

where the matrix function U_{2q} is defined by

$$\begin{aligned} U_{2q}(x, \theta) &= \frac{d}{d\theta}h(x, \theta) \\ &= \frac{d^2}{d\theta^2}\log q(x, \theta). \end{aligned}$$

Formula (5.3) appears in Wei and Tanner (1990) in the context of missing data models. Applying the importance sampling identity yields,

$$\begin{aligned} \mu'(\theta) = &\frac{\mathrm{E}_g\left(U_{2q}(x, \theta)\frac{q(x|\theta)}{g(x)}\right)}{\mathrm{E}_g\left(\frac{q(x|\theta)}{g(x)}\right)} + \frac{\mathrm{E}_g\left(U_q(x, \theta)U_q(x, \theta)^T\frac{q(x|\theta)}{g(x)}\right)}{\mathrm{E}_g\left(\frac{q(x|\theta)}{g(x)}\right)} \\ &- \frac{\mathrm{E}_g\left(U_q(x, \theta)\frac{q(x|\theta)}{g(x)}\right)}{\mathrm{E}_g\left(\frac{q(x|\theta)}{g(x)}\right)} \frac{\mathrm{E}_g\left(U_q(x, \theta)\frac{q(x|\theta)}{g(x)}\right)^T}{\mathrm{E}_g\left(\frac{q(x|\theta)}{g(x)}\right)}, \end{aligned}$$

with the estimate $\hat{\mu}'$ obtained by replacing E_g by $1/N \sum_{i=1}^{N}$ everywhere and x by x_i. In most models for which q is known, the vector and matrix functions U_q and U_{2q} can also be determined, and so the Monte Carlo estimates $\hat{\mu}$ and $\hat{\mu}'$ can be computed directly. The optimization step is then

$$\theta_{t+1} = \theta_t + [\hat{\mu}'(\theta_t) - \mu_0'(\theta_t)]^{-1}(\mu_0(\theta_t) - \hat{\mu}(\theta_t)),$$

where $\mu_0(\theta_t) = U_q(x_0, \theta_t)$ from (5.1), and

$$\mu_0'(\theta_t) = \frac{d}{d\theta}\mu_0(\theta_t) = U_{2q}(x_0, \theta_t).$$

Applying the algorithm of Sections 2 and 3 is straightforward and ideally will converge to a point of zero derivative of the likelihood function as the number of steps and iterations approach infinity. As with other maximization algorithms, if several modes are suspected, the iteration can be started from several points to explore parameter space. In addition, the log-likelihood function or its derivatives can itself be computed using the importance sampling formula at any point of interest using simulations based on a nearby value of θ, as suggested by Geyer and Thompson (1992). Gelfand and Carlin (in press) suggest maximizing the Monte Carlo-computed log-likelihood by stepwise ascent, which could be expected to be slower than Newton–Raphson but more reliable, especially for the early iterations far from a mode.

The Hessian matrix of the log-likelihood, which can be used as an asymptotic inverse covariance matrix for the maximum likelihood estimate, is estimated as a by-product of the Newton–Raphson iteration; in our notation, the estimated Hessian is $\hat{\mu}'(\theta)$ evaluated at the maximum likelihood estimate θ. Essentially the same result was noted by Gelfand and Carlin (in press), and Geyer (in press), and Wei and Tanner (1990).

In the special case that the normalized density functions are known, $q(x|\theta)$ reduces to $p(x|\theta)$, $\mu(\theta)$ is identically zero, μ' is the Hessian of the log-likelihood, and, in the limit of large N, our algorithm becomes equivalent to simple Newton–Raphson maximization of the likelihood.

5.2 Maximum Likelihood With Missing Data

As discussed by Gelfand and Carlin (in press), Geyer (in press), and Thompson and Guo (1991), the problem of missing data is mathematically very similar to unknown normalizing constants. A standard approach to maximizing likelihood with missing data is the EM algorithm (Dempster, Laird, and Rubin 1977). We show here that maximum likelihood with missing data can also be viewed as a generalized method of moments problem. The identities developed for our method are identical to those used for the Monte Carlo EM algorithm of Wei and Tanner (1990).

Let x be the observed data, y be the unobserved missing (or "latent") data, and $p(x, y|\theta)$ be the "complete-data" likelihood, which is assumed known. The goal of maximum likelihood with missing data is to maximize the observed data likelihood,

$$p(x|\theta) = \int p(x, y|\theta)dy,$$

or, equivalently, setting to zero the derivative of the observed-data log-likelihood, as in Fisher scoring,

$$\begin{aligned}\frac{d}{d\theta}\log p(x|\theta) &= \int\left(\frac{d}{d\theta}\log p(x,y|\theta)\right)\frac{p(x,y|\theta)}{p(x|\theta)}dy \\ &= E\left(\frac{d}{d\theta}\log p(x,y|\theta)\right),\end{aligned}$$

where the expectation averages over y in the distribution $p(y|x,\theta)$. We immediately recognize this as a generalized method of moments expression, in which $\mu_0 = 0$,

$$h(y) = U((x,y),\theta) = \frac{d}{d\theta}\log p(x,y|\theta),$$

and x is taken as fixed and equal to the observed data, x_0. The unobserved data y and the conditional likelihood $p(y|x,\theta)$ play the role of x and $p(x|\theta)$ in our earlier notation, and the Monte Carlo algorithm requires simulation of y from $p(y|x,\theta)$. As usual, these draws can be obtained directly for many standard models or using iterative simulation in general. If the joint density is only known in unnormalized form as $q(x,y|\theta)$, the method can be generalized as in Section 5.1.

6. EXAMPLES

6.1 SPECIFYING A PRIOR DISTRIBUTION IN A CONSTRAINED-PARAMETER FAMILY

Bois, Gelman, Jiang, and Maszle (1994) present an analysis of toxicology data using a pharmacokinetic model that describes metabolism of a toxin in the body in terms of a set of 19 physiological parameters. The physiological parameters are themselves characterized by a prior distribution that captures variability among persons in the general population. As a prelude to a Bayesian analysis, the prior distribution for the physiological parameters was specified based on a review of the relevant biological literature. It was deemed acceptable, from a scientific framework, to assign independent lognormal prior distributions to most of the 19 parameters, with parameters assigned based on the informal literature review. The major technical difficulty came with four parameters relating to blood flow, which were constrained to sum to 1. We now discuss how we assigned a prior distribution to these parameters that matched specified prior moments while satisfying the constraint.

For the purposes of this article we label the parameters x_1, x_2, x_3, x_4, with the constraint $\sum_{j=1}^{4} x_j = 1$. The information from the literature search was summarized as prior

Table 1. Specified Prior Moments for the Constrained-Parameter Example

Parameter	*E(log x_j)*	*sd(log x_j)*
x_1	log(.48)	log(1.2)
x_2	log(.20)	log(1.2)
x_3	log(.07)	log(1.2)
x_4	log(.25)	log(1.1)

means and standard deviations on the logarithms of the parameters, as displayed in Table 1. (Specification in terms of the logarithms makes sense for the lognormal distributions of the other parameters in the model. In practice, the prior variances are low enough that specifying the mean and coefficient of variation on the untransformed scale of ϕ would give virtually identical results.)

We first construct a parametric family for the prior distribution of x, given hyperparameters θ, and then determine θ by matching to the eight transformed moments given in Table 1, using the algorithm of Section 4.2. Given the constraint on x, we do not expect to have a full eight-parameter model, and so we use the least-squares fit to the given moments. For this application, an inexact fit is acceptable, since the numbers in Table 1 are only approximations based on a literature review.

The most familiar model for variables that sum to 1 is the Dirichlet. Applying our method, we fit the four parameters of the Dirichlet model to the eight moments given in Table 1. In computing the logarithm of the Dirichlet density and its derivative, we must compute the log-gamma function and its derivative, which are fortunately easy to calculate numerically using standard computer programs. We start the iteration at the point $\theta = (48, 20, 7, 25)$, which roughly fits the means and standard deviations in the first column of Table 1. We proceed with 20 steps of simulation and Newton–Raphson with $N = 2{,}000$, followed by one simulation of $N = 10{,}000$ and three Newton–Raphson steps. A final check was applied by another simulation of $N = 20{,}000$ and three more Newton–Raphson steps. For a comparison we ran another simulation, starting at the point $\theta = (240, 100, 35, 125)$. In both simulations the moments had reached approximate convergence, but not to the desired moments in Table 1. For example, the standard deviation of x_1 in the best method of moments fit is $\log(1.07)$, compared to the desired value of $\log(1.2)$. In fact, the Dirichlet model is well known to be too restrictive for many practical constrained-parameter situations (see, e.g., Aitchison 1986).

For a more flexible model, we use a transformed normal family, in which

$$x_i = \frac{e^{\phi_i}}{e^{\phi_1} + e^{\phi_2} + e^{\phi_3} + e^{\phi_4}} \text{ for } i = 1, \ldots, 4,$$

and the random variables ϕ_i have independent normal distributions:

$$\phi_i \sim \mathrm{N}\left(\theta_i, \theta_{i+4}^2\right).$$

We would like to determine the parameter vector θ for which the random variables x have the moments specified in Table 1. Actually, the transformed normal model has only seven free parameters, because we can add a constant to the means $\theta_1, \ldots, \theta_4$ and not change the distribution for x. In our iterations we keep θ_1 fixed and seek the least squares solution as a function of the other seven parameters. In applying our method the random variables ϕ play the role of x in our earlier notation; we simulate the vector ϕ from its normal distribution, given θ, and compute x and thus the estimated moments and their derivatives from ϕ; there is no need to determine the analytic form of the distribution of x. We start the iteration with the first four components of θ (the means of the components of ϕ) set to the values in the first column of Table 1 and the second four components (the standard deviations) set to the values in the second column of Table 1.

Table 2. Results of Two Runs of the Estimation Procedure for the Transformed Normal Model

	Specified values	*Last two iterations of simulation run 1*		*Last two iterations of simulation run 2*	
θ_1		-.734	-.734	-.734	-.734
θ_2		-1.586	-1.602	-1.586	-1.582
θ_3		-2.651	-2.773	-2.622	-2.633
θ_4		-1.371	-1.669	-1.375	-1.365
θ_5		.297	.237	.309	.298
θ_6		.177	.218	.182	.187
θ_7		.212	.305	.219	.205
θ_8		.002	.153	.003	.029
E(log(x_1))	log(.48)	log(.47)	log(.47)	log(.47)	log(.47)
E(log(x_2))	log(.20)	log(.20)	log(.20)	log(.20)	log(.20)
E(log(x_3))	log(.07)	log(.07)	log(.07)	log(.07)	log(.07)
E(log(x_4))	log(.25)	log(.25)	log(.25)	log(.25)	log(.25)
sd(log(x_1))	log(1.2)	log(1.17)	log(1.17)	log(1.19)	log(1.18)
sd(log(x_2))	log(1.2)	log(1.22)	log(1.23)	log(1.28)	log(1.23)
sd(log(x_3))	log(1.2)	log(1.27)	log(1.27)	log(1.31)	log(1.27)
sd(log(x_4))	log(1.1)	log(1.15)	log(1.14)	log(1.17)	log(1.16)

NOTE: The top eight rows are parameter estimates; the bottom eight rows are the estimated functions of moments, $f_1(\hat{\mu}(\theta)), \ldots, f_8(\hat{\mu}(\theta))$. Differences across the columns indicate that the iterations have not yet converged, but the convergence is acceptable for practical purposes.

We then apply the algorithm of Section 4.2 with the same simulation schedule as for the Dirichlet model. The results are displayed in Table 2. The estimates of θ are somewhat different, indicating that the algorithm has not reached convergence, but an examination of the moments shows that they have converged for all practical purposes. The moments depend very weakly on θ, so it is hard to find the exact solution to the least squares problem. For the applied problem at hand, we would not attempt to specify the prior mean and standard deviations of the parameters to more accuracy than the decimal places displayed in Table 3. The final fit is good, but not perfect, most notably in that the standard deviation of $\log(x_4)$ is not as low as the specified value of $\log(1.10)$. After consultation with the biologist who perused the substantive literature, it was decided that the best fit transformed normal model was an acceptable summary of the prior information for the problem. (In addition, the normal family is convenient for this application because it can easily be generalized into a hierarchical model; see, e.g., Gilks et al. 1993.)

6.2 ESTIMATION FOR A TWO-PARAMETER IMAGE MODEL

6.2.1 The Model and Simulation

We illustrate both the method of moments and the maximum likelihood computations with a nonlinear model for image analysis introduced by Geman and McClure (1987). In this problem the random variable x is a gray-level image defined on an $n \times n$ grid. To keep computation time down for this simple example, we set $n = 8$. The probability

density for x depends on two parameters and can be written in unnormalized form as

$$q(x|\theta_1, \theta_2) = \exp\left[\theta_1 \sum_{i,j} c_{ij} \frac{1}{1 + ((x_i - x_j)/\theta_2)^2}\right],$$

where the coefficients c_{ij} are defined by

$$\begin{aligned} c_{ij} &= 1 && \text{if } i \text{ and } j \text{ are orthogonal neighbors} \\ &= 1/\sqrt{2} && \text{if } i \text{ and } j \text{ are diagonal neighbors} \\ &= 0 && \text{otherwise.} \end{aligned}$$

(In this notation, i and j index the n^2 pixels, with physical location coded by the matrix c_{ij}.) In addition, the gray levels x_i are restricted to a finite range, which we set to $[0, 64)$. For $\theta_1 > 0$, the model favors nearly constant gray levels at adjacent pixels. However, the function inside the exponent is bounded (unlike a Gaussian log-likelihood); hence, when the gray levels of adjacent pixels are not nearly constant, they can differ by a large amount. The result can be used to model an image with nearly constant plateaus and occasional large changes.

Geman and McClure (1987) discussed the difficulties of estimating θ_1 and θ_2 from indirect tomographic data. For simplicity, we apply our methods to estimate the parameters from direct data x, although our methods could be applied to indirect data as well.

We know of no simple method of direct simulation of x, given θ, for this model, so we construct a Markov chain of images $x^1, x^2, \ldots$, as follows. One pixel of x is altered at a time, as in the Gibbs sampler, but with changes made according to the algorithm of Metropolis et al. (1953)—the candidate value of the component x_i is drawn from a normal distribution centered at the current value and with standard deviation σ, with wrap around if the drawn value is less than 0 or greater than 64. The Metropolis rule is then used to decide whether to accept or reject the candidate image, $\tilde{x}$. That is, with probability $\min(q(\tilde{x}|\theta)/q(x^{\text{old}}|\theta), 1)$, we set $x^{\text{new}} = \tilde{x}$; otherwise, the Markov chain stays still and we set $x^{\text{new}} = x^{\text{old}}$ The value of σ is altered adaptively so that the Metropolis acceptance rate is neither too low or too high, with a target acceptance rate of .4 (see Gelman, Roberts, and Gilks [in press] for a motivating argument for why .4 might be approximately optimal for one-dimensional Metropolis jumps). The

Table 3. Simulated Data x_0 From the Image Model With $\theta = (1,12)$

31	10	24	24	29	26	18	20
30	33	19	27	31	32	34	29
18	24	43	26	31	33	34	41
32	24	16	25	32	34	25	31
38	31	35	12	43	43	37	15
43	46	46	34	39	36	56	34
39	56	29	44	38	38	36	33
50	30	25	41	41	63	37	36

NOTE: Pixel intensities are on a 0 to 64 scale and are rounded to the nearest integer.

resulting algorithm is far from optimal for this problem—more sophisticated algorithms have been developed for this sort of image model, and even restricting to one-pixel-at-a-time Metropolis jumping, the .4 acceptance rate is not necessarily optimal here, but it is acceptably fast for the purposes of the simulations presented here. More complicated algorithms, based on altering several components of x at a time, could be expected to greatly increase the efficiency of our simulations (see Besag and Green 1993).

We check convergence of the Metropolis algorithm by starting two simulations at a constant image and an all-random image (corresponding to $\beta = \infty$ and 0, respectively, as in the simulations for the Ising model in Gelman and Rubin [1992a] and monitoring various univariate summaries, including the functions $h_1(x)$ and $h_2(x)$ defined in the following, using the between/within variance comparison of Gelman and Rubin (1992b). The simulations were run long enough so that, for each summary, the ratio of total variance to within-sequence variance was less than 1.1. When running simulations for the method of moments estimation, we continue each new set of simulation runs where the last runs ended, thus approximating the condition of very long runs in the limit of convergence of the moments estimate.

To illustrate our estimation methods, we first simulate a "data" image x_0 from the model, based on "true values" of $\theta_1 = 1$, $\theta_2 = 12$, and applying the Gibbs–Metropolis simulation algorithm for 2,000 steps. The result is displayed in Table 3.

6.2.2 Method of Moments Estimation

We start by defining two functions of x—the average difference in gray levels between neighboring pixels and the proportion of neighboring pixels that are nearly identical, differing in gray level by no more than 2. We will match the moments of these statistics.

$$h_1(x) = \frac{\sum_{ij} C_{ij}|x_i - x_j|}{\sum_{ij} C_{ij}}$$

$$h_2(x) = \frac{\sum_{ij} C_{ij} I_{|x_i - x_j| \le 2}}{\sum_{ij} C_{ij}},$$

where I is the indicator function, and

$$\begin{aligned} C_{ij} &= 1 \quad \text{if } i \text{ and } j \text{ are orthogonal or diagonal neighbors} \\ &= 0 \quad \text{otherwise.} \end{aligned}$$

The functions h_1 and h_2 roughly estimate the features of occasional large changes amid nearly constant plateaus that are modeled by the specified family of distributions. We purposely pick slightly awkward forms for h_1 and h_2 and set C_{ij} different from c_{ij} to illustrate the flexibility available in setting the moments. (However, we still use the known, correct values of c_{ij} in computing the density function.)

We now proceed as in Section 3. Starting at $\theta = (.5, 5)$, we run 20 steps of simulation and optimization at $N = 2{,}000$, with no importance sampling steps, followed by two steps at $N = 10{,}000$, each with three importance sampling steps. The Newton–Raphson steps in the replication behaved wildly in the early iterations, and so we started the

Table 4. Method of Moments Estimation for the Image Model

	Observed value	Last two iterations of simulation run 1		Last two iterations of simulation run 2	
θ_1		1.23	1.19	1.19	1.19
θ_2		3.40	3.75	3.70	3.65
μ_1	9.20	9.19	9.20	9.20	9.20
μ_2	.67	.67	.67	.67	.67

NOTE: Observed values of the moments come from the "data" in Table 3.

simulation again, beginning with 10 quick preliminary steps at $N = 500$ with the first 5 steps "decelerated" to a factor of .1 of their original size and the next 5 steps decelerated to a factor of .5. We replicate the procedure starting at $\theta = (2, 40)$, with convergence to approximately the same point. Rather than display all the iterations, we just show the final two steps (to indicate the simulation variability); they are displayed in the first row of Table 4. The fit to the moments is quite accurate, with some uncertainty still present in the parameter estimates, as is roughly shown by the comparisons between the simulation runs.

6.2.3 Maximum Likelihood Estimation

As a comparison we applied the method of Section 5.1 to estimate (θ_1, θ_2) by maximum likelihood from the "data" in Table 3 (p. 48). The results of the iteration, using the same simulation schedule as for the moments estimation, are shown in the first two rows of Table 5; as in the previous computation, the early iterations were decelerated to stop the Newton–Raphson steps from wandering away. In addition, the asymptotic-theory covariance matrix for (θ_1, θ_2) was estimated as the inverse of the Hessian computed from the last step of each simulation; the estimates are displayed at the bottom of Table 5. Given that the data consist of only a single 8×8 image, the normal-theory approximation to the likelihood is not necessarily accurate, but it may be useful for a rough estimate of uncertainty. From the asymptotic covariance matrices in Table 5 we estimate $\text{sd}(\hat{\theta}_1) = 0.23$, $\text{sd}(\hat{\theta}_2) = 7.1$ or 7.0, and $\text{corr}(\hat{\theta}_1, \hat{\theta}_2) = .83$ or .85. Given these uncertainties, the sampling variability that remains in the estimates of θ seems acceptable.

Table 5. Maximum Likelihood Estimation for the Image Model

	Parameter estimates			
	Last two iterations of simulation run 1		Last two iterations of simulation run 2	
θ_1	1.10	1.09	1.10	1.09
θ_2	13.14	12.84	13.15	13.02

Estimates of $\text{cov}(\hat{\theta}_1, \hat{\theta}_2)$

Simulation run 1			Simulation run 2	
$\begin{pmatrix} .051 & 1.32 \\ 1.32 & 50.02 \end{pmatrix}$	or		$\begin{pmatrix} .055 & 1.39 \\ 1.39 & 49.47 \end{pmatrix}$	

7. DISCUSSION

7.1 PRACTICAL VIRTUES OF THE METHOD OF MOMENTS

Perhaps the greatest practical applications of the methods of this article are in setting the parameters of probability models based on prior knowledge or speculation about moments. In a Bayesian analysis it is fairly common for prior information to be understood about moments but not in the form of any particular parametric model. In areas such as image analysis, complicated models are commonly used for likelihoods and prior distributions to encourage certain kinds of data fitting, but without any serious belief that the probability distributions correspond to reality. In such cases it may be reasonable to fit a distribution to moments, if only as an exploratory check on other methods such as maximum likelihood or Bayesian estimation that may be more sensitive to artifacts of the model. Besag (1986), Gelman (1994), and Ripley (1988, pp. 85–94) all discussed the fallibility of maximum likelihood and Bayesian inference for models in time series and spatial statistics that have smoothness assumptions not warranted by the data. If one is using a model for the purposes of smoothing, it may not be necessary for the model to fit the data, but the parameters in the model still must be estimated in some reasonable way, perhaps by matching relevant moments, although one must always worry about the lack of efficiency using such methods. The idea of estimating data using moments in the context of a model that is either not fully specified or has features that may not fit the data also appears in generalized estimating equations (Liang and Zeger 1986). In either case—fitting to prior knowledge or data moments—the option of least squares fitting to more moments than parameters provides additional flexibility and possibly robustness.

7.2 COMPUTATIONAL DIFFICULTIES

Computationally, the iterations are not always stable. In our experience so far, when the algorithm does not work, it usually fails dramatically in the first few steps. Thus we recommend looking at the early output of the estimated θ_t and $\hat{\mu}$ values. We have noticed the following typical problems (in addition to the usual programming errors).

- A starting point that is too far away from the solution, causing a jump into never-never land. This happened in some of our simulations for the image example in Section 6.2. Possible solutions include a search for a better starting point or restricting the first Newton–Raphson steps to jump fractional amounts, as we in fact did for the image example. Another approach is to go beyond Newton–Raphson to more flexible optimization methods such as the Levenberg–Marquardt algorithm (Marquardt 1963), as has been done by researchers in Monte Carlo maximum likelihood (e.g., Geyer and Thompson 1992).
- Too-variable importance ratios, caused by large jumps in θ. The solution is to simulate a new set of Monte Carlo draws from the new value of θ and not use the importance sampling formula until the jumps in θ become smaller.
- Jumps outside the boundary of parameter space. This occurred with the prior distribution example of Section 6.1, in which the standard deviations for the normal distributions were restricted to be positive. One possible solution is to

restrict the Newton–Raphson steps to go only part way toward the boundary. If the solution is itself on the boundary, one must abandon Newton–Raphson entirely.

- It is possible to have moments μ_0 for which no value of θ will produce corresponding theoretical moments, or for the least squares fit to be unacceptably far from the desired μ_0, as occured for the Dirichlet model in the prior distribution example of Section 6.1. In these cases, it may be necessary to re-evaluate the model or the specified moments in light of the disagreement. In fact, it may be considered a virtue of this method that such discrepancies can come to light.

Another problem, which we have not encountered but may occur, is for the estimate $\hat{\mu}'$ to be noninvertible, due to Monte Carlo variation, for the method of moments problem.

In addition to the concerns of stability of the optimization steps, the Monte Carlo simulations offer the usual question of how many simulations are needed at each iteration. In our examples, it has been possible to monitor the runs and increase N where necessary to increase stability. An interesting open problem is to set up a more quantitative approach, to allow relatively fast and automatic computation in large problems. The Monte Carlo simulations provide an internal estimate of error which, presumably, should optimally be the same order as the correction term in the optimization step. Further research is needed on this point, perhaps following Carter (1991, 1993), as well as in the related problems of Monte Carlo maximization and simulated annealing.

Finally, various open questions arise about the convergence of the optimization algorithm under suitable conditions and a suitable schedule for increasing N and accelerating or decelerating the optimization steps; the literature on stochastic approximation (e.g., Ruppert 1985) is relevant. Geyer (in press a, b) presents theoretical convergence results about related Monte Carlo optimization schemes and discusses why very strong conditions would be required for any general proofs of convergence of methods based on Newton–Raphson. The theoretical results tell us that the methods described in this article cannot be expected to automatically converge, even to a "local solution" and must be used with caution.

7.3 SUMMARY

The method of moments is rarely, if ever, the best approach to a statistical problem but is often useful as a robust alternative, an approximation, or a direct method of incorporating prior information. The Monte Carlo Newton–Raphson algorithm can have serious difficulties with convergence, but has the advantage of being straightforward to implement for a wide variety of problems and is often effective, even when the unknown parameter θ has several dimensions. More generally, the Monte Carlo estimates for generalized moments and their derivatives can be used with more sophisticated equation-solving or optimization algorithms. We envision the family of methods described in this article as a useful addition to a modeler's toolkit (as in Tanner 1991), not as a stand-alone general approach to statistics.

As a practical matter, the algorithm, in all its variations, has some useful redundancy

in that convergence can be noted in three ways: (1) the estimates θ_t should converge to a point; (2) the Monte Carlo-estimated moments $\hat{\mu}$ should approach the specified values μ_0; and (3) convergence should be attained from different starting points. Failure in any of these points could be due to zero or multiple solutions to the moments equations, lack of convergence of the Newton–Raphson algorithm, not enough Monte Carlo samples, or simply programming error. In actual implementations, it should be possible to investigate these possibilities.

[*Received January 1994. Revised August 1994.*]

ACKNOWLEDGMENTS

We thank Frederic Bois, Charles Geyer, Xiao-Li Meng, and several referees for helpful comments. We also thank the American Lung Association for financial support, and the National Science Foundation for grants SBR-9223637, DMS-9404305, and DMS-9457824.

REFERENCES

Aitchison, J. (1986), *The Statistical Analysis of Compositional Data*, London: Chapman and Hall.

Besag, J. (1986), "On the Statistical Analysis of Dirty Pictures" (with discussion), *Journal of the Royal Statistical Society,* Ser. B, 48, 259–302.

Besag, J., and Green, P. J. (1993), "Spatial Statistics and Bayesian Computation," *Journal of the Royal Statistical Society,* Ser. B, 55, 25–102.

Bois, F. Y., Gelman, A., Jiang, J., and Maszle, D. R. (1994), "A Toxicokinetic Analysis of Tetrachloroethylene Metabolism in Humans," Technical report, Cal-EPA, Berkeley, CA.

Carter, R. G. (1991), "On the Global Convergence of Trust Region Agorithms Using Inexact Gradient Information," *SIAM Journal on Numerical Analysis,* 28, 251–265.

——— (1993), "Numerical Experience With a Class of Algorithms for Nonlinear Optimization Using Inexact Function and Gradient Information," *SIAM Journal on Scientific Computing,* 14, 368–388.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977), "Maximum Likelihood From Incomplete Data via the EM Algorithm" (with discussion), *Journal of the Royal Statistical Society,* Ser. B, 39, 1–38.

Gelfand, A. E., and Carlin, B. P. (in press), "Maximum Likelihood Estimation for Constrained or Missing Data Models," *Canadian Journal of Statistics.*

Gelfand, A. E., and Smith, A. F. M. (1990), "Sampling-based Approaches to Calculating Marginal Densities," *Journal of the American Statistical Association,* 85, 398–409.

Gelman, A. (1994), "Bayesian Model-Building by Pure Thought: Some Principles and Examples," submitted for publication in *Statistica Sinica.*

Gelman, A., Roberts, G., and Gilks, W. (in press), "Efficient Metropolis Jumping Rules," *Bayesian Statistics,* 5.

Gelman, A., and Rubin, D. B. (1992a), "A Single Series From the Gibbs Sampler Provides a False Sense of Security," in *Bayesian Statistics 4*, ed. J. Bernardo, Oxford University Press, pp. 625–631.

——— (1992b), "Inference From Iterative Simulation Using Multiple Sequences" (with discussion), *Statistical Science,* 7, 457–511.

Geman, S., and McClure, D. E. (1987), "Statistical Methods for Tomographic Image Reconstruction," *Proceedings of the 46th Session of the ISI, Bulletin of the ISI,* 52, 5–21.

Geyer, C. J. (in press), "On the Convergence of Monte Carlo Maximum Likelihood Calculations," *Journal of the Royal Statistical Society,* Ser. B.

——— (in press), "Approximation of Functions and Optimization," in *Practical Markov Chain Monte Carlo*, eds. W. Gilks, S. Richardson, and D. Spiegelhalter, New York: Chapman and Hall.

Geyer, C. J., and Thompson, E. A. (1992), "Constrained Monte Carlo Maximum Likelihood for Dependent Data" (with discussion), *Journal of the Royal Statistical Society*, Ser. B, 54, 657–699.

Gilks, W. R., Clayton, D. G., Spiegelhalter, D. J., Best, N. G., McNeil, A. J., Sharples, L. D., and Kirby, A. J. (1993), "Modelling Complexity: Applications of Gibbs Sampling in Medicine," *Journal of the Royal Statistical Society,* Ser. B, 55, 39–102.

Hammersley, J. M., and Handscomb, D. C. (1964), *Monte Carlo Methods*, London: Chapman and Hall.

Liang, K. Y., and Zeger, S. L. (1986), "Longitudinal Data Analysis Using Generalized Linear Models," *Biometrika,* 73, 153–160.

Kong, A. (1992), "A Note on Importance Sampling Using Standardized Weights," Technical Report #348, University of Chicago, Dept. of Statistics.

Marquardt, D. W. (1963), "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *Journal of the Society for Industrial and Applied Mathematics,* 11, 431–441.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953), "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics,* 21, 1087–1092.

Moyeed, R. A., and Baddeley, A. J. (1991), "Stochastic Approximation of the MLE for a Spatial Point Pattern," *Scandinavian Journal of Statistics,* 18, 39–50.

Penttinen, A. (1984), "Modelling Interaction in Spatial Point Patterns: Parameter Estimation by the Maximum Likelihood Method," *Jyvaskyla Studies in Computer Science, Economics and Statistics,* 7.

Ripley, B. D. (1988), *Statistical Inference for Spatial Processes*, Cambridge, U.K.: Cambridge University Press.

Robbins, H., and Monro, S. (1951), "A Stochastic Approximation Method," *The Annals of Mathematical Statistics,* 22, 400–407.

Ruppert, D. (1985), "A Newton-Raphson Version of the Multivariate Robbins-Monro Procedure," *The Annals of Statistics,* 13, 236–245.

Tanner, M. A. (1991), *Tools for Statistical Inference: Observed Data and Data Augmentation Methods*, New York: Springer-Verlag.

Thompson, E. A., and Guo, S. W. (1991), "Evaluation of Likelihood Ratios for Complex Genetic Models," *IMA Journal of Mathematics Applied in Medicine and Biology,* 8, 149–169.

Wei, G. C., and Tanner, M. A. (1990), "A Monte Carlo Implementation of the EM Algorithm and the Poor Man's Data Augmentation Algorithms," *Journal of the American Statistical Association,* 85, 699–704.