Inference and Monitoring Convergence (chapter for Gilks, Richardson, and Spiegelhalter book)

Andrew Gelman
Department of Statistics
University of California
Berkeley, CA 94720*

* Thanks to Jiming Jiang for the figures, Xiao-Li Meng for helpful comments, and the American Lung Association for financial support.

0.1 Difficulties of inference from Markov chain simulation

Markov chain simulation is a powerful tool—so easy to apply, in fact, that there is the risk of serious error, including:

1. Inappropriate modeling: the assumed model may not be realistic from a substantive standpoint or may not fit the data.
2. Errors in calculation or programming: the stationary distribution of the simulation process may not be the same as the desired target distribution, or the algorithm, as programmed, may not converge to any proper distribution.
3. Slow convergence: the simulation can remain for many iterations in a region heavily influenced by the starting distribution. If the iterations are used to summarize the target distribution, they can yield falsely-precise inference.

The first two errors can occur with other statistical methods (such as maximum likelihood), but the complexity of Markov chain simulation makes mistakes more common. In particular, it is possible to program a method of computation such as the Gibbs sampler or Metropolis' algorithm that only depends on local properties of the model without ever understanding the large-scale features of the joint distribution. For a discussion of this issue in the context of probability models for images, see Besag (1986).

Slow convergence is a problem with deterministic algorithms as well; consider, for example, the literature about the convergence of EM and related algorithms (e.g., Meng and Pedlow, 1992). In deterministic algorithms, the two most useful ways of measuring convergence are (a) monitoring individual summaries, such as the increasing likelihood in the EM and ECM algorithms or the symmetry of the covariance matrix in the SEM algorithm (Dempster, Laird, and Rubin, 1977; Meng and Rubin, 1993; Meng and Rubin, 1991), and (b) replicating the algorithm with different starting points and checking that they converge to the same point (or, if not, noting multiple solutions). We apply both general approaches to Markov chain simulation, but we must overcome the difficulties that (a) the algorithm is stochastic, so we cannot expect any summary statistic to increase or decrease monotonically, and (b) convergence is to a distribution, rather than a point.

This chapter presents an overview of methods for addressing two practical tasks: monitoring convergence of the simulation and summarizing inference about the target distribution using the output from the simulations. The material in Sections 3–5 is presented in more detail, with an example, in Gelman and Rubin (1992b). The final section of this chapter introduces and provides references to various methods in the recent statistical literature for using inference from the simulation to improve the efficiency of the Markov chain algorithm.

The practical task in "monitoring convergence" is to estimate how much the inference based on the Markov chain simulations differs from the desired target distribution. Our basic method, inspired by the analysis of variance, is to form an overestimate and an underestimate of the variance of the target distribution, with the property that the estimates will be roughly equal at convergence but not before.

0.2 The risk of undiagnosed slow convergence

The fundamental problem of inference from simulation is that, for any simulation, there will be areas of the target distribution that have not been covered by a finite Markov chain. As the simulation progresses, the ergodic property of the Markov chain causes it eventually to cover all the target distribution, but in the short term the simulations cannot, in general, tell us about areas where they has not been. Incidentally, this is a general problem whenever convergence is slow, even in a distribution that has a single mode. It has happened several times in our experience that a single sequence of Markov chain simulation has appeared to have "converged," even though evidence from replications makes it clear that the movement of the simulation is just too slow to detect.

In our own experience of applying Markov chain simulation to probability models and Bayesian posterior distributions, we have commonly noticed poor convergence by examining multiple independent simulations. In many of these settings, any single one of the simulated sequences would have appeared to have converged perfectly if examined alone; some of these examples have been published as Figures 1–3 of Gelman and Rubin (1992a)—note the title of that article—and Figure 4 of Gelman and Rubin (1992b). In these examples, quantitative methods of diagnosing lack of convergence from a single sequence (e.g., Hastings, 1970, Raftery and Lewis, 1992, Geyer, 1992) all fail, because the simulations are moving so slowly, or are "stuck" in separate places in the target distribution. For this article we present yet another example, from our current applied research.

Figure 0.1 displays an example of slow convergence from a Markov chain simulation for a hierarchical Bayesian model for a pharmacokinetics problem (see Bois et al., 1994, for details). The simulations were done using a Metropolis-approximate Gibbs sampler (as in Section 4.4 of Gelman, 1992); due to the complexity of the model, each iteration was expensive in computer time, and it was desirable to keep the simulation runs as short as possible. Figures 1a and 1b display time series plots for a single parameter in the posterior distribution in two independent simulations, each of length 1000. The simulations were run in parallel simultaneously on two workstations in a network. It is clear from the separation of the two sequences that, after 1000 iterations, the simulations are still far from convergence. However, either sequence alone looks perfectly well behaved.

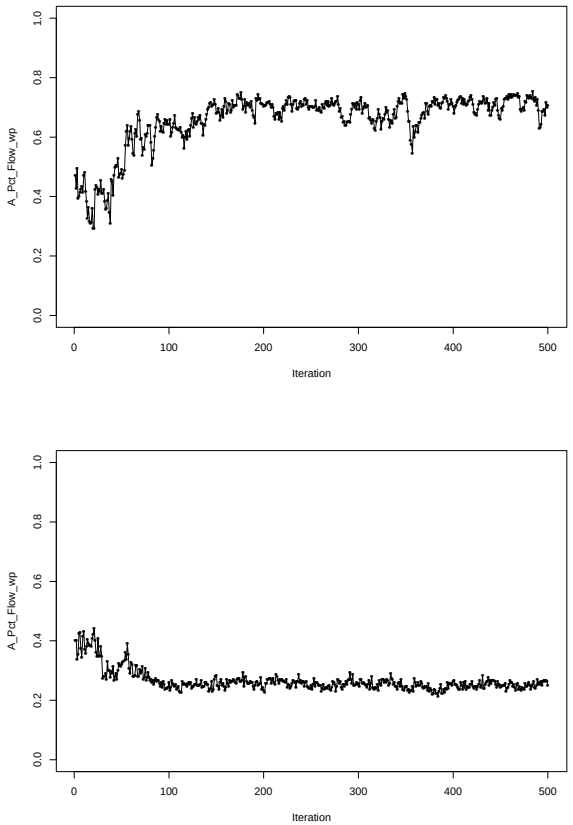

Figure 0.1. *Time series of the value of a single parameter from two parallel simulations of length 1000 from iterative simulation of a complicated multiparameter model. Lack of convergence is evident by comparing the simulations but cannot be detected from either sequence alone.*

Interestingly, we do not yet know whether the slow convergence exhibited in Figure 0.1 is due to an inappropriate model, programming mistakes, or just slow movement of the Markov chain. As is common in applied statistics, we have repeatedly altered our model in the last several months of research on this problem, as we have gained understanding about the relation between the model, the data, and our prior information. The simulations from a previous version of the model had reached approximate convergence before 1000 iterations, which leads us to suspect an error of some sort in the simulation leading to Figure 1. (One of the strengths of Markov chain simulation has been that it has allowed us to change our model with

only minor alterations in the computer program.) Several times in this research, we have noted poor convergence by comparing parallel sequences, and each time we have investigated and found a substantive flaw in the model or a programming error, or else we have had to alter our Markov chain simulation to run more efficiently. Another approach would be to run the simulation indefinitely and wait until the lines begun in Figures 1a and 1b overlap; because of the slowness in computing the simulation draws at each step, we would prefer to avoid this approach. Also, our model is still in a preliminary stage, and so any investment made now in computational efficiency (or in debugging!) can be expected to pay off repeatedly in computations with future versions of the model.

0.3 Designing the simulations to make inference more reliable: multiple sequences and overdispersed starting points

Our general approach to monitoring convergence of Markov chain simulations is based on plots such as Figure 0.1 above. We have always found it useful to simulate at least two parallel sequences, typically four or more. If the computations are implemented on a network of workstations or a parallel machine, it makes sense to run as many parallel simulations as there are free workstations or machine processors. The recommendation to always simulate multiple sequences is not new in the iterative simulation literature (e.g., Fosdick, 1959) but is somewhat controversial (see the discussion of Gelman and Rubin, 1992b, and Geyer, 1992). In our experience with Bayesian posterior simulation, however, we have found that the added information obtained from replication (as in Figures 0.1) outweighs any additional costs required in multiple simulations.

It is desirable to choose starting points that are widely dispersed in the target distribution. Overdispersed starting points are an important design feature because starting far apart can make lack of convergence apparent (as in Figure 0.1), and also for the purposes of inference, to ensure that all major regions of the target distribution are represented in the simulations. For many problems, especially with discrete or bounded parameter spaces, it is possible to pick several starting points that are far apart by inspecting the parameter space and the form of the distribution. For example, the proportion in a two-component mixture model can be started at values of 0.1 and 0.9 in two parallel sequences.

In more complicated situations, more work may be be needed to find a range of dispersed starting values. In practice, we have found that any additional effort spent on approximating the target density is useful for understanding the problem and for debugging: after the Markov chain simulations have been completed, the final estimates can be compared to the earlier approximations. In complicated applied statistical problems, it is standard practice to gradually improve models as more information be-

comes available, and the estimates from each model can be used to obtain starting points for the computation in the next stage.

Before running the Markov chain simulations, it is important to have a rough idea of the extent of the target distribution. In many problems, initial estimates can be obtained using the data and a simpler model; for example, approximating a hierarchical generalized linear model by a linear regression or nonhierarchical generalized linear model computation. In other problems, including the example used for Figure 0.1, the prior distribution is informative and can be used to construct rough bounds on the parameters of the model. In problems without strong prior distributions, it is often useful to locate the mode or modes of the target distribution using some deterministic algorithm such as stepwise ascent, EM, or ECM (Dempster, Laird, and Rubin, 1977; Meng and Rubin, 1993). (Once the Markov chain simulation algorithm has been programmed, it is often easily altered to find modes, by replacing random jumps with deterministic steps to move to higher points in the target density.) It is also useful to estimate roughly the scale of the target distribution near the modes, which can often be done by computing the second derivative matrix of the log-posterior density at each mode. For continuous-parameter problems, starting points for parallel Markov chain simulations can be drawn from an approximate Student-t mixture distribution based on the posterior modes, possibly corrected by importance resampling; see Gelman and Rubin (1992b) for details. If the target distribution is multimodal, or suspected to be multimodal, it is a good idea to start at least one sequence at each mode. If the number of modes is large, the simulation algorithm should be designed to frequently jump between modes. As we have seen, preliminary estimation is not always easy, but the effort generally pays off in greater understanding of the model and confidence in the results.

0.4 Monitoring convergence using simulation output

Our recommended general approach to monitoring convergence is based on detecting when the Markov chains have "forgotten" their starting points by comparing several sequences drawn from different starting points and checking that they are indistinguishable. There are many possible ways to compare parallel sequences, the most obvious approach being to look at time series plots such as Figures 1a and 1b overlaid and see if the two sequences can be distinguished. Here we outline a more quantitative approach based on the analysis of variance: approximate convergence is diagnosed when the variance "between" the different sequences is no larger than the variance "within" each individual sequence.

A more general formulation of the method presented here is to identify "convergence" with the condition that empirical distribution of simulations obtained separately from each sequence is approximately the same as the

distribution obtained by mixing all the sequences together. Before the parallel sequences have converged, the collected simulations from each single sequence will be much less variable than the simulations collected from all the sequences combined; consider Figure 0.1, for example.

The approach we have found most convenient is based on separately monitoring the convergence of all scalar summaries of interest from the target distribution. For example, we may be interested in all the parameters in the distribution and various predictive quantities. We will defer until the end of this section a discussion of what scalar summaries should be monitored; for the purpose of defining the method, we shall consider a single summary at a time, and label it ψ. We shall assume m parallel simulations, each of length n.

For each scalar summary of interest, we would like a numerical equivalent of the comparison in Figure 0.1 that states, "the two sequences are much farther apart than we could expect just based on their internal variability." For each scalar summary ψ, we label the m parallel sequences of length n as (ψ_{ij}), $j = 1, \ldots, n; i = 1, \ldots, m$, and we compute the following two quantities—the between-sequence variance B and the within-sequence variances W:

$$B = \frac{n}{m-1}\sum_{i=1}^{m}(\overline{\psi}_{i\cdot} - \overline{\psi}_{\cdot\cdot})^2, \quad \text{where } \overline{\psi}_{i\cdot} = \frac{1}{n}\sum_{j=1}^{n}\psi_{ij}, \quad \overline{\psi}_{\cdot\cdot} = \frac{1}{m}\sum_{j=1}^{n}\overline{\psi}_{i\cdot}$$

$$W = \frac{1}{m}\sum_{i=1}^{m}s_i^2, \quad \text{where } s_i^2 = \frac{1}{n-1}\sum_{j=1}^{n}(\psi_{ij} - \overline{\psi}i\cdot)^2.$$

The between-sequence variance B contains a factor of n because it is based on the variance of the within-sequence means, $\overline{\psi}_{i\cdot}$, each of which is an average of n values ψ_{ij}.

From the two variance components, we construct two estimates of the variance of ψ in the target distribution. First,

$$\widehat{\text{var}}(\psi) = \frac{n-1}{n}W + \frac{1}{n}B$$

is an estimate of the variance that is *unbiased* under stationarity (that is, if the starting points of the simulations were actually drawn from the target distribution), but is an *overestimate* under the more realistic assumption that the starting points are overdispersed. We call $\widehat{\text{var}}(\psi)$ a "conservative" estimate of the variance of ψ under overdispersion.

Meanwhile, for any finite n, the "within" variance W should *underestimate* the variance of ψ because the individual sequences have not had time to range over all of the target distribution and, as a result, will have less variability; in the limit as $n \to \infty$, both $\widehat{\text{var}}(\psi)$ and W approach $\text{var}(\psi)$, but from opposite directions.

We can now monitor the convergence of the Markov chain by estimating

the factor by which the conservative estimate of the distribution of ψ might be reduced; that is, the ratio between the estimated upper and lower bounds for the standard deviation of ψ, which we call the "estimated potential scale reduction,"

$$\sqrt{\hat{R}} = \sqrt{\frac{\widehat{\text{var}}(\psi)}{W}}.$$

(This is $\hat{R}$ rather than R because the numerator and denominator are merely *estimated* upper and lower bounds on the variance.) As the simulation converges, the potential scale reduction declines to 1, meaning that the parallel Markov chains are essentially overlapping. If the potential scale reduction is high, then we have reason to believe that proceeding with further simulations may improve our inference about the target distribution.

For example, the estimate $\widehat{\text{var}}(\psi)$ derived from the two simulations of Figures 0.1 would just about cover the range of both sequences, and is about 2.5^2, while the average within variance W measures just the variance within each sequence and is about 0.5^2. The estimated potential scale reduction $\hat{R}$ is about 5 for this example, indicating poor convergence and a potential for confidence intervals for ψ to shrink by as much as a factor of 5 once convergence is eventually reached.

In general, if $\widehat{R}$ is not near 1 for all scalar summaries of interest, it is probably a good idea to continue the simulation runs (and perhaps alter the simulation algorithm itself to make the simulations more efficient, as we discuss in Section 6 of this chapter). In practice, we generally run the simulations until the values of $\widehat{R}$ are all less than 1.1 or 1.2. Using this method, we never have to actually look at graphs such as Figure 0.1; the potential scale reductions are all computed automatically.

There is still the question of what scalar summaries to monitor, although the above approach simplifies the problem in practice by making monitoring so easy that we can, and have, monitored over a hundred summaries for a single problem and just scanned for values of $\widehat{R}$ greater than 1.2 as indicating poor convergence. We have no problem monitoring all parameters and hyperparameters of a model and also examining predictive simulations of interest and other summaries of interest such as the ratio between two variance components. Tables 2 and 3 of Gelman and Rubin (1992b) provide an example of monitoring several summaries at once. In addition, the method could be generalized to monitor convergence of vector summaries, in which case B, W, and $\widehat{\text{var}}(\psi)$ become matrices whose eigenvalues can be compared to estimate the potential reduction in the scale of vector inferences.

Another issue to consider is sampling variability of the quantities W and B; we do not want to falsely declare convergence when $\widehat{R}$ just happens to be near 1 in a short simulation run. In practice, sampling variability of the convergence monitoring statistics is not a serious concern, because, regard-

less of convergence, one will almost always run the simulations long enough to get a fairly good estimate of the variance in the target distribution. In addition, if several scalar summaries are being monitored, it is extremely unlikely that they will all appear to have converged by "luck," especially if the number of parallel simulations m is fairly large (at least 10, say). For theoretical completeness, however, it is possible to correct the above estimates for sampling variability, leading to a slightly different estimate of R; details appear in Gelman and Rubin (1992b), and a computer program ("itsim") in the S language is available on Statlib or from the author.

A potentially useful improvement for monitoring convergence is to create an underestimate of $\text{var}(\psi)$ that is more efficient than W, by making use of the autocorrelated time-series structure of the iterations within each series. Hastings (1970) discusses this approach, and Geyer (1992) reviews some more recent theoretical results in this area; both these references attempt to estimate $\text{var}(\psi)$ from a single Markov chain sequence, which is a hopeless task in many practical applications (see Section 2 of this chapter), but can be useful as improved *under*estimates for use in place of W in the formula for $\hat{R}$.

In addition, several methods have been proposed in recent years to use the Markov chain transition probabilities, which are known in most applications of Markov chain simulation, to more efficiently diagnose lack of convergence. At convergence, the simulations in any sequence should look just as "likely" backward as forward and the joint distribution of successive simulations in a sequence should be symmetric. Cui et al. (1992) construct a scalar summary based on these principles that can diagnose poor convergence in cases where summaries based only on the simulation output (and not the transition probabilities) fail. Liu, Liu, and Rubin (1992) and Roberts (1993) construct somewhat similar scalar summaries using information from multiple sequences. All these methods are most effective when used in addition to the more basic analysis of variance approach for monitoring scalar summaries of interest.

0.5 Inference about the target distribution

Our main practical concern in Bayesian inference is to make reliable inferences about the target distribution; for example, claimed 95% regions that include *at least* 95% of the mass of the target distribution, with exact coverage as the length of the Markov chain simulations approach infinity.

The simplest and most generally useful idea in inference is to use the empirical distribution of the simulated draws, as in multiple imputation (Rubin, 1987), with the iterations from all the parallel simulations mixed in together. If θ is the vector variable from which N values have been simulated, this means computing any moments of the posterior distribution using sample moments of the N draws of θ, estimating 95% posterior in-

tervals of any scalar summary ψ by the 2.5% and 97.5% order statistics of the N simulated values ψ, and so forth. This approach is generally reliable if based on multiple sequences with overdispersed starting points. Intervals obtained before convergence should be overdispersed and conservative; once approximate convergence has been reached, the intervals and other summaries of the target distribution should be accurate, up to the granularity of the finite number of simulation draws. If the early parts of the simulated sequences have been discarded in monitoring convergence, they should also be discarded for the purposes of inference.

It has sometimes been suggested that inferences should be based on every k-th iteration of each sequence, with k set to some value high enough that successive draws of θ are approximately independent. This strategy can be useful when the set of simulated values is so large that reducing the number of simulations by a factor of k gives important savings in storage and computation time. Except for storage and the cost of handling the simulations, however, there is no advantage in discarding intermediate simulation draws, even if highly correlated. The step of mixing the simulations from all m sequences and then choosing at random destroys any serial dependence in the simulated sequences, and even correlated draws add some information. A quantitative treatment of these issues is given by Geyer (1992), for the case of estimating the mean of a scalar summary using simulation draws.

Suppose, as is common, we are interested in the distribution of a scalar summary, ψ, for a multivariate target distribution, $p(\theta)$. If we know the mathematical form of the conditional density of ψ given the other components of θ, then we can obtain a better estimate of the density of ψ by averaging the conditional densities over the simulated values of θ:

$$\hat{p}(\psi) = \frac{1}{N} \sum_{i=1}^{N} p(\psi | \theta_i(-\psi)),$$

where the notation $\theta_i(-\psi)$ represents all the components of θ_i except for ψ. The application of this method to Markov chain simulation, specifically the Gibbs sampler, is due to Tanner and Wong (1987) and Gelfand and Smith (1990), with a theoretical proof of its effectiveness by Liu, Kong, and Wong (1994).

0.6 Some current research topics on inference from iterative simulation

An interesting area of current research combines the ideas of inference and efficiency. It is possible to improve the convergence monitoring process in various ways to more effectively use the information in the Markov chain simulation. Most obviously, we note that the early part of a simulation is often far from convergence, and we can crudely create simulations that are

closer to convergence by simply discarding the early parts of the simulated sequences. In our applications, we have followed the simple but effective approach of discarding the first half of each simulated sequence and apply the above procedure to the remainder.

Finally, it is possible, both in theory and practice, to use inference about the target distribution to improve the efficiency of the simulation algorithm. Many different ideas apply to this problem. A Gibbs sampler is generally most efficient when the jumps are along the principal components of the target distribution; inference from early simulations can be used to reparameterize (Hills and Smith, 1992). In a Metropolis algorithm, theory from the normal distribution suggests that the most efficient jumping kernel is shaped like the target distribution scaled by a factor of about $2.4/\sqrt{d}$, where d is the dimension of the target distribution (Gelman, Gilks, and Roberts, 1994; also see Muller, 1993). The scale and shape of the target distribution can again be estimated from early simulation draws, and the simulations can be adaptively altered as additional information arrives. Inference from multiple simulated sequences is useful here, so that the early estimates of the target distribution are conservatively spread. Other related approaches suggested by normal distribution theory for the Metropolis algorithm involve adaptively altering Metropolis jumps so that the frequency of acceptances is in the range of 1/4 to 1/2, or optimizing the average distance jumped (Gelman, Gilks, and Roberts, 1994). These approaches have not yet reached the stage of automatic implementation; as Gelfand and Sahu (1993) demonstrate, transition rules that are continually adaptively altered have the potential for converging to the wrong distribution. We anticipate that the interaction between methods of inference, monitoring convergence, and improvements in efficiency will ultimately lead to more automatic, reliable, and efficient iterative simulation algorithms.

References

Bois, F. Y., Gelman, A., Jiang, J., and Maszle, D. R. (1993). A toxicokinetic analysis of tetrachloroethylene metabolism in humans. Technical report.

Besag, J. (1986). On the statistical analysis of dirty pictures (with discussion). *Journal of the Royal Statistical Society B* **48**, 259–302.

Cui, L., Tanner, M. A., Sinha, D., and Hall, W. J. (1992). Monitoring convergence of the Gibbs sampler: further experience with the Gibbs stopper. Comment on Gelman and Rubin (1992b) and Geyer (1992). *Statistical Science* **7**, 483–486.

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 B* **39**, 1–38.

Fosdick, L. D. (1959). Calculation of order parameters in a binary alloy by

the Monte Carlo method. *Physical Review* **116**, 565–573.

Gelfand, A., and Sahu, S. K. (1993). On Markov chain Monte Carlo acceleration. Technical report, Department of Statistics, University of Connecticut.

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. (1992). Iterative and non-iterative simulation algorithms. *Computing Science and Statistics* **24**, 433–438.

Gelman, A., Gilks, W. R., and Roberts, G. O. (1994). Efficient Metropolis jumping rules. Technical report, Department of Statistics, University of California, Berkeley.

Gelman, A., and Rubin, D. B. (1992a). A single sequence from the Gibbs sampler gives a false sense of security. In *Bayesian Statistics 4*, ed. J. M. Bernardo et al., 625–631. New York: Oxford University Press.

Gelman, A., and Rubin, D. B. (1992b). Inference from iterative simulation using multiple sequences (with discussion). *Statistical Science* **7**, 457–511.

Geyer, C. J. (1992). Practical Markov chain Monte Carlo (with discussion). *Statistical Science* **7**, 473–511.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**, 97–109.

Hills, S. E., and Smith, A. F. M. (1992). Parameterization issues in Bayesian inference (with discussion). In *Bayesian Statistics 4*, ed. J. Bernardo, Oxford University Press, 227–246.

Liu, C., Liu, J., and Rubin, D. B. (1992). A variational control variable for assessing the convergence of the Gibbs sampler. Proceedings of the Statistical Computing Section, American Statistical Association, 74–78.

Liu, J., Kong, A., and Wong, W. H. (1994). Correlation Structure and Convergence Rate of the Gibbs Sampler with Applications to the Comparisons of Estimators and Augmentation Schemes. *Biometrika*, to appear.

Meng, X. L., and Pedlow, S. (1992). EM: a bibliographic review with missing articles. *Proceedings of the Statistical Computing Section, American Statistical Association*, 24–27.

Meng, X. L., and Rubin, D. B. (1991). Using EM to obtain asymptotic variance-covariance matrices: the SEM algorithm. *Journal of the American Statistical Association* **86**, 899–909.

Meng, X. L., and Rubin, D. B. (1993). Maximum likelihood estimation via the ECM algorithm: a general framework. *Biometrika* **80**, 267–278.

Muller, P. (1993). A generic approach to posterior integration and Gibbs sampling. *Journal of the American Statistical Association*, to appear.

Raftery, A. E., and Lewis, S. M. (1992). How many iterations in the Gibbs

sampler? In *Bayesian Statistics 4*, ed. J. M. Bernardo et al., 763–773. New York: Oxford University Press.

Roberts, G. O. (1994). Methods for estimating L^2 convergence of Markov chain Monte Carlo. In the Arnold Zellner honorary volume, to appear.

Rubin, D. B. (1987). *Multiple Imputation for Nonresponse in Surveys*. New York: Wiley.

Tanner, M. A., and Wong, W. H. (1987). The calculation of posterior distributions by data augmentation (with discussion). *Journal of the American Statistical Association* **82**, 528–550.