An NCME Instructional Module on

Estimating Item Response Theory Models Using Markov Chain Monte Carlo Methods

Jee-Seon Kim and Daniel M. Bolt, *University of Wisconsin, Madison*

The purpose of this ITEMS module is to provide an introduction to Markov chain Monte Carlo (MCMC) estimation for item response models. A brief description of Bayesian inference is followed by an overview of the various facets of MCMC algorithms, including discussion of prior specification, sampling procedures, and methods for evaluating chain convergence. Model comparison and fit issues in the context of MCMC are also considered. Finally, an illustration is provided in which a two-parameter logistic (2PL) model is fit to item response data from a university mathematics placement test through MCMC using the WINBUGS 1.4 software. While MCMC procedures are often complex and can be easily misused, it is suggested that they offer an attractive methodology for experimentation with new and potentially complex IRT models, as are frequently needed in real-world applications in educational measurement.

Keywords: Bayesian estimation, goodness-of-fit, item response theory models, Markov chain Monte Carlo, model comparison

Estimating Item Response Theory Models Using Markov chain Monte Carlo

It has become increasingly common in educational measurement to use Markov chain Monte Carlo (MCMC) techniques for estimating item response models (see e.g., Beguin & Glas, 2001; Bolt & Lall, 2003; Bradlow, Wainer, & Wang, 1999; De la Torre, Stark, & Chernyshenko, 2006; Fox & Glas, 2001; Johnson & Sinharay, 2005; Patz & Junker, 1999a). MCMC offers many advantages, including its relative ease of implementation and the availability of free software for its use. Many researchers have found MCMC to provide a framework within which to experiment with new models needed for specialized measurement applications before going to the more challenging process of implementing maximum likelihood procedures, for example. MCMC also represents an estimation strategy that is firmly rooted in a perspective of Bayesian inference, which makes it appealing for IRT applications oriented around this perspective (see, e.g., Glas & Meijer, 2003; McLeod, Lewis, and Thissen, 2003; Zwick, Thayer & Lewis, 2000).

Despite these advantages, the MCMC methodology presents a number of unique challenges. MCMC algorithms can be quite sophisticated, and their proper use requires careful attention to several facets of implementation. Because of its complexity, MCMC can also be easily misused, and in many cases it may be difficult for users to ascertain whether the results of an MCMC analysis can be taken with confidence, especially with more challenging models. One of the other primary drawbacks of MCMC is its heavy computational demand. The sampling procedures that underlie the MCMC methodology generally require a very large number

Jee-Seon Kim and Daniel Bolt are both Associate Professors in the Department of Educational Psychology at the University of Wisconsin-Madison, 1025 West Johnson Street, Madison, WI 53705 (jeeseonkim@wisc.edu, dmbolt@wisc.edu). Dr. Kim's research interests include multilevel analysis, longitudinal data analysis, IRT models, and test equating. Dr. Bolt's research interests include item response theory and related applications.

Series Information

ITEMS is a series of units designed to facilitate instruction in educational measurement. These units are published by the National Council on Measurement in Education. This module may be photocopied without permission if reproduced in its entirety and used for instructional purposes. Information regarding the development of new ITEMS modules should be addressed to: Dr. Mark Gierl, Canada Research Chair in Educational Measurement and Director, Centre for Research in Applied Measurement and Evaluation, Department of Educational Psychology, 6–110 Education North, University of Alberta, Edmonton, Alberta, CANADA T6G 2G5.

of iterations before model parameters can be reliably estimated. It is not uncommon for a single estimation run to take several hours, or even a day or more, for more complex models or when analyzing large amounts of data.

The purpose of this module is to provide an introduction to MCMC methods and to illustrate their application in estimating item response models. In that context, we also discuss the WINBUGS software (Spiegelhalter, Thomas, Best, & Lunn, 2003; downloadable at http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/contents.shtml), a computer program that allows for easy implementation of various MCMC procedures. WINBUGS can be used to fit a host of different statistical models and has built in graphical and analytical tools that help the user monitor the estimation process and evaluate results. Admittedly, the topic of MCMC is a difficult one to present in an introductory way in a single paper; as a result, we will frequently refer to some useful references addressing different aspects of this topic to fill in details. In addition, the reader may find it helpful to have a software package like WINBUGS at their disposal in experimenting with certain aspects of MCMC. The WINBUGS package comes with several example models and data sets, including some specific to item response theory, that can help in learning the methodology. Clearly one of the best ways to learn MCMC techniques is to experiment with the procedure using models and datasets that are already familiar. Prior to discussing the specifics of MCMC, however, we review some basic principles of the Bayesian theoretical framework, as this ultimately provides the foundation for the MCMC methodology.

Brief Overview of Bayesian Inference

Any attempt to understand MCMC requires familiarity with the core principles of Bayesian inference. The centerpiece of this framework is Bayes' theorem. (Readers previously unfamiliar with this topic may find it useful to study Congdon (2001) or similar sources before attempting to work MCMC methods.) Bayes' theorem is often portrayed in terms of the probabilities of discrete events, say an event "A" and an event "B." For example, in a medical context, event A could represent the presence (or absence) of a particular type of disease, say the measles, and event B the outcome of a test for the disease returning a result "positive" or "negative." From Bayes' theorem, we know that

$$P(A \mid B) = P(B \mid A)P(A) \Big/ \left[\sum_{A} P(B \mid A)P(A) \right], \qquad (1)$$

where we refer to P(A | B) as the *posterior probability* of A given B (e.g., the probability of having the measles given the result of the test); P(A) as the *prior probability* of A (e.g., the prior probability of having measles), and P(B | A) as the *conditional probability* of B given A (e.g., the probability of a particular test result given the presence or absence of the measles).

The summation in the denominator of the right hand side of (1) represents an accumulation across all possible outcomes of event A (e.g., has the measles, doesn't have the measles), and thus can also be taken as the probability of B, P(B) (i.e., the overall probability of a particular test result). Ultimately, Bayes' theorem provides a representation of the conditional probability of one event given another (i.e., A given B, or the probability of measles given the test result) in terms of the opposite conditional probability (i.e., B given A, or the probability of a particular test result given the presence or absence of the measles). For example, suppose it is known that the probability of a positive test result given that one truly has the measles is .95, the probability of a false positive is .02, and the overall probability of having the measles among those tested is .10. Using Bayes' theorem (as shown in equation 1) we can determine that the probability of having the measles given the positive test is $(.95)(.10)/[.95(.10) + .02(.90)] = .84$.

When fitting an item response model to item response data, practitioners are faced with a situation that reflects Bayes' theorem. Usually a primary goal in fitting an IRT model is to obtain information about parameters of the item response model (e.g., item difficulty, item discrimination, examinee ability) from the item response data. In terms of Bayes' theorem, this information is reflected in the relative likelihood(s) of particular parameter values for the model ("event A") given the observed item response data ("event B"). The chosen IRT model (e.g., two-parameter logistic model; 2PL) provides a basis for describing the opposite conditional probability, namely the probability of the item response data (B) given the model parameters (A).

Although at a conceptual level this idea fits, Bayes' theorem as portrayed in (1) is inappropriate for most item response theory applications as item and ability parameters are generally continuous values, not discrete events. Consequently we cannot consider the "probability" of their occurrence as we can for finite discrete outcomes. It will therefore be more convenient to portray Bayes' theorem in the form of continuous *probability density functions*, which represent the relative likelihood of each outcome. Familiar examples of probability density functions include the normal density function, which takes the form of the familiar bell-shape curve. In the IRT context, Bayes' theorem can be written with respect to probability density functions as:

$$f(\Omega \mid X) = f(X \mid \Omega) * f(\Omega) \Big/ \left[\int_{\Omega} f(X \mid \Omega) f(\Omega)\, d\Omega \right] \qquad (2)$$

where X denotes all of the item response data (i.e., the correct/incorrect scores of each examinee to each item), and Ω all of the unknown parameters, which in IRT generally consist of item and person parameters. The use of "f(·)" in place of "P(·)" and "∫" in place of "∑" in (2) compared to (1) accounts for the continuous nature of the parameter values. One other fundamental difference between (1) and (2) is that Ω represents many parameters (i.e., multiple "events"), and X many observed outcomes, implying that the quantities in (2) should be thought of as multivariate outcomes rather than univariate ones. Consequently, we will view Ω as a set of hypothetical outcomes for all of the item and examinee parameters in the model.

The left hand side of (2), referred to as the *joint posterior density* (of the model parameters given the data), is used to determine estimates of the model parameters. To evaluate it requires knowledge about the quantities on the right hand side. The quantity f(X | Ω), which expresses the likelihood of the item response data given all of the model parameters, is defined by the item response model (e.g., 2PL) along with its associated assumptions of local independence. The quantity $f(\Omega)$ is the *prior density* of the model parameters, and can be thought of as indicating the relative likelihoods of particular parameter values "prior to" data collection. The quantity in the denominator is now written in terms of integration (as opposed to discrete summation) of the conditional distribution of the data given parameters over the parameter space, and is a constant for a fixed data set. It is often referred to as a *normalizing constant* as its value generally makes $f(\Omega \mid X)$ a proper density. Because this value is typically unknown (and often not easy to determine), we sometimes write:

$$f(\Omega \mid X) \propto f(X \mid \Omega) * f(\Omega) \qquad (3)$$

to indicate that the joint posterior density is "proportional to" the product of the quantities on the right hand side. This proportionality relationship is often the basis for sampling procedures that underlie MCMC, as it makes it possible to evaluate (and sample with respect to) the relative likelihoods of different sets of parameter values even if the exact density of the posterior density cannot be determined.

The ultimate goal of MCMC is to reproduce the $f(\Omega \mid X)$ distribution. Although this distribution cannot often be determined analytically, as in our earlier example, it is often possible to sample observations with respect to it (for details, see Spiegelhalter, Thomas, Best, and Gilks, 1995). By sampling enough observations, it becomes possible to determine characteristics of the distribution, such as its mean and variance, that can be the basis for model parameter estimates. The precise mechanism by which sampling is best conducted varies depending on the known features of $f(\Omega \mid X)$; consequently, there are various different types of sampling algorithms considered within MCMC. Once an appropriate sampling procedure is determined, aspects of the posterior relevant for determining parameter estimates, such as the mean and standard deviation, become possible through computing corresponding characteristics of the generated sample, such as its mean and standard deviation.

A fundamental difference between MCMC and other popular estimation techniques such as maximum likelihood (ML) estimation is the emphasis in Bayesian inference on estimating distributions, as opposed to point estimates, when describing model parameters. On the one hand, this allows a potentially richer description of the parameter estimate distribution than is usually provided in ML estimation. However, it is not uncommon to see MCMC methods used in a frequentist fashion, where point estimates (e.g., mean of posterior) and standard errors (e.g., standard deviation of posterior) are reported (see Rupp, Dey, and Zumbo, 2004 for more discussion of these issues).

Implementation of MCMC with IRT Models

Despite various approaches that can be applied in implementation of MCMC methods, a common set of considerations apply. Below, we outline some issues in the selection of priors, sampling procedures, diagnostics for evaluating chain convergence, and model comparison and fit issues under MCMC.

Specification of priors

IRT practitioners frequently use priors in ML estimation, especially when using IRT estimation programs such as BILOG (Mislevy & Bock, 1989) or MULTILOG (Thissen, 1991). Although not strictly needed in ML, priors allow known information about the characteristics of items to be incorporated into the estimation process, and can also be useful in addressing problems in estimation, such as when the data provide very little information about certain parameters (e.g., the "guessing" parameter in the three parameter logistic model). Unlike in ML however, in MCMC specification of priors of some sort is always necessary for all item and examinee parameters in IRT models, as the prior densities are needed to define the posterior densities. Because of the fundamental role of priors in MCMC, there are several important considerations in selecting them. One concerns the distributional family chosen for the prior. Where possible, it is usually desirable to select priors that are *conjugate priors.* By definition, conjugate priors are priors that return posterior distributions from the same family of distributions as the prior. This is appealing because it implies the distributional form of the posterior is known, which makes sampling from it much easier. As seen in Equation (2), the prior density interacts with the specified model in determining the form of the posterior. Thus, the existence of conjugate priors also depends on the type of model chosen.

As a more general statistical modeling example, suppose a sample of observations (say $X = 5, 2, 4, 4, 9, 2, 3, 6$) were assumed to come from a normal distribution with unknown mean (μ) and known variance of 1. The specification of a normal prior for the unknown mean (μ), say Normal(5, 2), would return a posterior for the mean that was also normal, as the normal prior is a conjugate prior for the mean of a normally distributed variable. Specifically, based on the example data, the posterior distribution $f(\mu \mid X)$ can be shown to be Normal with an approximate mean of 4.4 and variance of .12 using Equation (2). Conjugate priors are thus desirable in that they result in posterior distributions of a known functional form, and thus make sampling in MCMC more computationally efficient. Any prior other than a normal prior would in this case be a nonconjugate prior, thus rendering a posterior of unknown distributional form. Various tables can be consulted (see e.g., Spiegelhalter et al., 1995, p. 21) for determining conjugate priors for the parameters of a given model.

A second issue in selecting priors concerns the chosen strength of the priors. The influence of the priors can often be controlled through the parameters specified for the prior distribution, referred to as *hyperparameters*. In the above example, the specified variance of the prior distribution for the population mean (a hyperparameter) could be reduced from 2 to .2 in order to increase the strength of the prior. Note that by reducing the variance of the prior, we are indicating a higher level of confidence in the likely values of the parameter. The posterior density naturally then also changes, and is now a normal density with mean of approximately 4.6, and a variance of approximately .08. In effect, the stronger prior has reduced the influence of the data (sample mean of $X = 4.375$) relative to the prior in determining the distributional characteristics of the posterior distribution.

Figure 1 provides an illustration of the two example prior densities and the resulting posterior densities. The greater strength of the prior with smaller variance is seen from its sharper peak. The posterior densities, both normal, have means at the estimates of μ mentioned above.

It is important to note that the hyperparameters need not be assigned specific values. In fact, the hyperparameters can

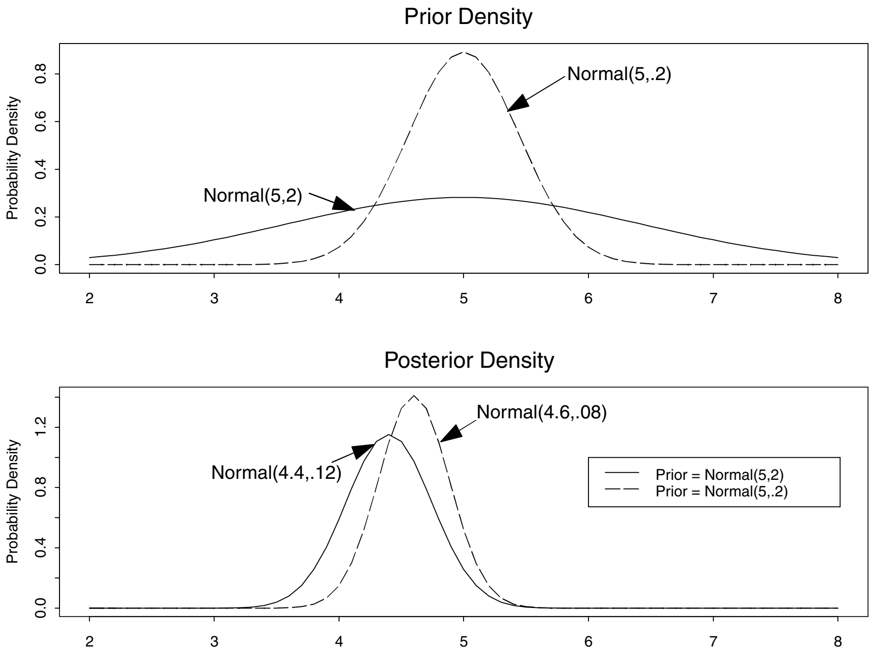

FIGURE 1. Illustration of Prior and Posterior Density Functions for Unknown Mean (μ) of Normal Distribution.

themselves be assigned priors in the form of *hyperpriors*. For example, in an IRT model, we may assign a normal prior to the difficulty parameters of the items, but regard the mean of this prior as unknown, with some specified prior distribution. The use of hyperpriors provides a way in which the strength of priors can be reduced. The use of hyperpriors introduces a hierarchical structure to the model that is sometimes portrayed in graphical form. For details, see Baker and Kim (2004, pp. 303–305).

To allow the data to provide as much information as possible regarding the posterior density, highly noninformative priors can be assigned to model parameters by specifying a very large variance, for example. Noninformative priors ultimately reduce the influence of the prior in defining the posterior distribution. Under some conditions, it may be possible to assign an *improper* prior, essentially a prior that assumes all values are equally likely, implying no known prior information about the parameter. However, such priors cannot be implemented within WINBUGS.

Finally, it should be noted that even when conjugate priors do not exist for a given IRT model, it is often possible to specify priors of a form that will render posterior densities that have known properties that can be used to make MCMC sampling more efficient (see, for example, Speigelhalter et al., 1995, p.13). We return to this issue in more detail shortly when discussing the WINBUGS program.

Sampling Procedures

Once the data have been collected, an IRT model has been chosen, and priors have been specified for all model parameters, sampling becomes possible. As noted earlier, the objective of MCMC is to define a mechanism by which observations can be sampled from the joint posterior density of model parameters shown in (2). This makes the iterative process conducted under MCMC considerably different from that conducted in ML, as the sampled values do not "converge" to a point estimate of the model parameters, but rather a distribution that represents the posterior distribution of the model parameters.

Although we will not go into much detail on the specific algorithms that have been proposed for this purpose, it is worth noting that the algorithms frequently differ in terms of their efficiency and rates of convergence, and since computational time is one of the primary drawbacks of MCMC, the chosen sampling mechanism is an important consideration. Details on some different approaches common in IRT can be found in Patz and Junker (1999b), and for more general statistical models in Gilks, Richardson and Spiegelhalter (1996). We briefly describe two of the more common procedures below.

(a) Gibbs sampler

The *Gibbs sampler* provides a mechanism by which sampling can be performed with respect to smaller numbers of parameters, often one at a time. In this latter case, the Gibbs sampler samples with respect to *univariate conditional distributions* of the model parameters. Unlike the full joint posterior distribution, the conditional distributions, denoted $f(\Omega_k \mid X, \Omega_{-k})$ represent the posterior distribution of a single model parameter Ω_k conditional upon the data X and all other model parameters Ω_{-k}. For example, under an IRT model, a univariate conditional distribution could correspond to the difficulty parameter of a single item given the data and all other item and ability parameters. Under Gibbs sampling, each parameter is then sampled individually with respect to its conditional distribution, regarding all the other parameters as known. In this respect, the Gibbs sampler can be viewed as a type of "divide and conquer" strategy (Patz & Junker, 1999b), as each parameter is updated one at a time.

An important consideration in implementing the Gibbs sampler involves choosing priors that result in conditional distributions that can make sampling efficient. As noted, when conjugate priors are chosen, the posterior distributions have a known form, and samples can often be generated directly from that distribution. However, in IRT models, direct Gibbs sampling has only been implemented for normal ogive IRT models, and even then requires use of a process referred to as *data augmentation*. Details are provided in Albert (1992), Baker (1998), and Bradlow et al. (1999).

Due to its use of known conditional distributions for sampling, implementation of Gibbs sampling is often straightforward as standard statistical software can be used to generate the samples. Details on the generation of these samples are provided shortly.

(b) Metropolis Hastings

Alternative procedures to Gibbs sampling are needed when the conditional distributions are not of a known distributional form. This is generally the case when logistic IRT models (e.g., two-parameter or three-parameter logistic models) are estimated. Patz and Junker (1999b) discuss the use of Metropolis Hastings sampling in such contexts. The Metropolis Hastings method makes use of the proportionality relationship established in Equation (3). In effect, the samples are indirectly taken from the joint posterior by generating candidate observations from proposal distributions. For example, the proposal distribution might be a normal distribution, with mean determined by current state of the parameter in the chain, and a specified variance. These candidate observations are then chosen as a new state for the chain in proportion to their relative likelihood (based on Equation 3) compared against the current state of sampled parameter values in the Markov chain. If the candidate state is rejected, the previous state is retained as the new state. Key issues in implementing a Metropolis Hastings strategy revolve around construction of the proposal distributions, as the frequency with which the candidate observations are retained affects the efficiency of the algorithm (see Patz and Junker, 1999b for further discussion of this issue). For example, the value of the constant variance chosen in the above example will likely have a substantial influence on how frequently the proposal distributions return values that are accepted as new states for the chain.

Monitoring the Markov chain

Once a sampling mechanism has been determined, observations are randomly and repeatedly sampled in an iterative fashion producing a series of observations that represent states in a Markov chain. To begin the sampling process, it is necessary to specify an initial set of values for the model parameters. These values can be randomly generated, or in some other way be systematically defined, such as educated guesses as to point estimates of the model parameters. Regardless of the method used, we refer to this first state as the *starting state* of the chain. However, the sequence of values produced in this chain will not be independent; new states will likely be affected by previous states, as the conditional distributions of parameters are defined at least in part by the values of the previous state. This is particularly true under Metropolis Hastings which, as noted, has the potential to retain the previous state as its current state. As a result, there will typically be a positive correlation between parameter values sampled at successive states in the chain. This not only makes it inappropriate to view a small number of states as a random sample from the posterior, but also makes the initial sampled states of questionable value, as they will likely be influenced by the starting state. It is therefore common to dismiss a number of the initial states (referred to as "burn-in" states) and to estimate the posterior only from observations sampled after the burn-in period. For example, with IRT models, it has been common to dismiss the first 500 or so states as burn-in iterations.

Another critical issue in monitoring the simulated states of the Markov chain involves evaluating chain convergence. The sequence of states for the Markov chain should theoretically converge to a stationary distribution such that the sampled observations can be viewed as a sample from the posterior distribution of the model parameters. The rate at which this convergence occurs can vary depending on several factors. First, high correlations between adjacent states imply a slow rate of convergence, thus requiring a very large number of iterations before the sampled states can be viewed as a sample from the posterior. Second, the sampling algorithm used can also affect the rate of convergence. For example, under Metropolis Hastings, if the candidate states generated from proposal distributions are rarely selected, longer chains will be needed. Other causes of nonconvergence may relate to identification problems with the model.

Detecting convergence is an important part of determining whether an MCMC run has been successful. Lack of convergence can sometimes be apparent from an inspection of the history of the chain. The sampling histories of the chains are often graphically depicted as in Figure 2. Figures 2(a) and (b) provide example illustrations of two chains, one of which shows a high likelihood of convergence (a), the other of which demonstrates nonconvergence (b). Various diagnostic criteria/indices can also be applied to observations from the chain to evaluate the likelihood of convergence. Several are implemented within the computer program CODA v0.3 (Best, Cowles, & Vines, 1996) that can be run under statistical programs such as R or Splus, and is used in conjuction with WINBUGS output. One example of a convergence diagnostic is Geweke's (1992) criterion. Under Geweke's approach, a z score is computed from the sampled states for each parameter. The z-score for a given parameter is defined by taking the difference between the mean of the first 10% of states, and the mean of the last 50% of states, and dividing by their pooled standard deviation. Z-values within a range of non-significance (e.g., $-1.96 \leq z \leq 1.96$) can be taken as evidence of convergence. A similar approach by Raftery and Lewis (1992a) considers the number of samples needed to estimate quantiles of the posterior with sufficient precision. The Raftery and Lewis criterion returns an index, I, indicating the increase in the number of sampled states needed to reach convergence due to autocorrelations in the chain; values of $I > 5.0$ indicate problems with convergence.

Still another strategy for evaluating convergence is to simulate multiple chains, each based on a different starting state. If the chains converge to the same stationary distribution, as is often reflected by a large overlap in their sampling histories, there is a strong likelihood of convergence. As with a single chain, diagnostic indices can also be applied to multiple chains to evaluate convergence. An example is the Gelman and Rubin (1992) criterion. The Gelman and Rubin test is based on a comparison of the variances within and between chains for each parameter. From these quantities, a variance ratio statistic $\sqrt{\hat{R}}$ is then computed for each parameter, where $\sqrt{\hat{R}} \approx 1$ is taken to imply convergence.

Taken together, criteria such as those mentioned above lend credibility to the results from a Markov chain Monte

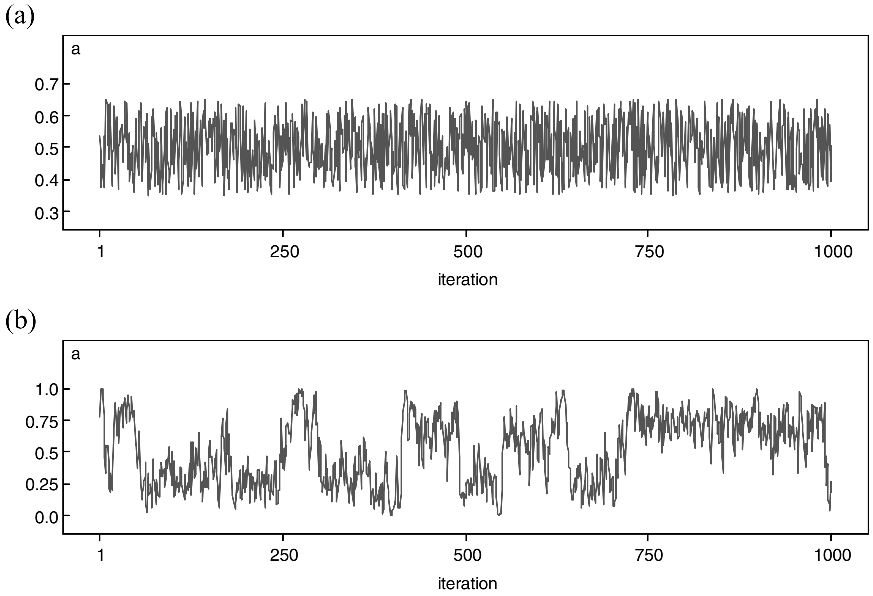

FIGURE 2. Examples of Sampling Histories Associated with Markov Chains Displaying Evidence of Convergence (a) and Nonconvergence (b).

Carlo analysis as a basis for approximating the joint posterior. Various other criteria also exist for evaluating convergence, as described in Best, Cowles and Vines (1996). It is important to note, however, that even satisfaction of the above criteria should not be taken as a guarantee of convergence. For additional information on the issue of convergence under MCMC within psychometric models, see Sinharay (2004).

Constructing posterior distributions

Once sufficient evidence of convergence has been obtained, the simulated chain can be used to construct the marginal posterior distributions that are the basis for model parameter estimates. Several additional issues are considered in this process. The first concerns the number of burn-in states to dismiss. Raftery and Lewis (1992b) recommend basing this decision in part on the estimated autocorrelations (i.e., the correlations between adjacent states) in the chain, which can be estimated using CODA. The length of the burn-in should naturally be at least as large as the distance between samples needed to achieve an autocorrelation of 0. However, because the actual burn-in usually involves a relatively small number of iterations ($<1\%$ of the total), the effect of some inaccuracy is generally of minimal significance.

A second consideration involves the possibility of *thinning* the chain. For example, rather than including all states of the chain, it is possible to choose only every fifth or tenth state, for example, if substantial autocorrelations are present. When thinning is not used, CODA also has procedures that can appropriately adjust the standard errors of the parameter estimates to account for the autocorrelations.

Finally, in determining how many sampled states of the chain are necessary, it is important to recognize the influence of Monte Carlo standard errors on the results (Patz and Junker, 1999b). Because the posterior distributions are constructed from samples, they are imprecise due to sampling error. Consequently, when computing moments of the posterior distributions, such as their means, error in the estimates can be attributed not only to the standard error of the point estimate (as reflected by the standard deviation of the posterior), but also to sampling error, referred to as Monte Carlo error. As a rule of thumb, the simulation should be run until the Monte Carlo error for each parameter of interest is less than about 5% of the sample standard deviation (Spiegelhalter et al., 2003). The Monte Carlo error can always be reduced by lengthening the chain. More on the distinction between these two forms of error will be described in the real data example.

Evaluating model fit

Because of the emphasis in Bayesian inference on posterior distributions as opposed to point estimates of model parameters, MCMC methods are also generally associated with different procedures for evaluating model fit than when using ML methods. One general strategy involves the use of *posterior predictive checks*. Sinharay (2005) and Sinharay, Johnson, and Stern (2006) provide good illustrations of ways in which posterior predictive checks can be used with item response models. A *posterior predictive distribution* refers to the distribution for a replicate set of observations (e.g., a new item response data set) conditional on the distribution of model parameters given the observed data. Such replicate observations can be easily generated within WINBUGS even in the process of estimating the model parameters. From these generated replicate observations, a discrepancy statistic is chosen that can be evaluated both for the observed item response data as well as each of the replicate data sets. The discrepancy statistic for the actual data is compared against the distribution of discrepancy statistics across the replicate data sets to evaluate model fit. For example, if the discrepancy statistic for the real data exceeds (in magnitude) a large percentage of the discrepancy statistics observed for the replicated datasets (say 95%), the model is said not to fit (at $\alpha = .05$). Different types of deviance statistics can be chosen depending on the aspect of model misfit of greatest concern. For example, if local dependence among item pairs is of concern, a deviance statistic such as an item-pair odds ratio can be used to identify item pairs that fail to satisfy this condition. Other deviance statistics might attend to features of the test score distribution, or item proportion correction statistics. We consider an application of posterior predictive checks using odds ratios in a real data example introduced shortly.

Model comparison

Beyond studies of absolute model fit, other approaches can be used for *model comparison*. Unlike statistical tests of model fit, these criteria identify which of one or more models provides a better fit to the data, without evaluating the degree of fit in an absolute sense. We consider two possibilities, the *Pseudo-Bayes Factor* criterion, and the *Deviance Information Criterion* (DIC). To introduce the former index, it is first necessary to define the *Bayes factor*, a fundamental concept for model comparison under Bayesian inference (see Raftery, 1996). A Bayes Factor (BF) is an index for comparing models that is defined as the ratio of the marginal likelihoods of the data under each model. In other words,

$$\mathrm{BF} = \frac{\mathrm{Likelihood(Data|Model1)}}{\mathrm{Likelihood(Data \mid Model2)}} \tag{4}$$

where the preferred model is the model returning the higher likelihood. Based on (4), Model 1 is preferred when BF>1, and Model 2 is preferred when BF<1. The relative magnitude of BF also can be used in evaluating the relative weight of evidence in support of either model, with BF >12 implying strong evidence in favor of Model 1, and BF>150 implying very strong evidence in favor of Model 1 (Jeffreys, 1961), for example.

In practice, it is common to approximate this comparison using the conditional predictive ordinate (CPO), a so-called pseudo-Bayes factor comparison (Geisser & Eddy, 1979; Gelfand, Dey & Chang, 1992). The CPO can be computed at the level of an individual item response as

$$\mathrm{CPO}^{-1} = \frac{1}{\mathrm{T}} \sum_{1}^{\mathrm{T}} 1/\mathrm{f}(\mathrm{x} \mid \Omega_t) \tag{5}$$

where T is the total number of sampled states in a chain, and $\mathrm{f}(\mathrm{x} \mid \Omega_t)$ is the likelihood of the observed item response ($\mathrm{x} = 0$ or 1) based on the sampled parameter values at state t.

In IRT modeling, a separate CPO index can be computed for each item response. A summary of the index values across item responses can be computed by taking the log of the product of the CPOs. The preferred model is the one returning the higher log product. In the WINBUGS program, computation of the CPO is straightforward, as it only requires tracing the inverse probability of each observed item response over the MCMC sampled states, and then tabulating the average of their logs when the chain is finished.

A second index for model comparison is the Deviance Information Criterion (DIC; Spiegelhalter, Best, Carlin & van der Linde, 2002). The DIC is an index similar to the Akaike Information Criterion (AIC; Akaike, 1973) and the Bayesian Information Criterion (BIC; Schwartz, 1978) often used under ML in that it weighs both model fit and model complexity in identifying the preferred model. The DIC is based on the posterior distribution of the deviance (i.e., −2 x log likelihood) and a term representing "the effective number of parameters" that accounts for the expected decrease in deviance attributable to the added parameters of the more complex model. Estimation of the DIC index can be requested within the WINBUGS program. As with AIC and BIC, the smaller the value of DIC, the better the model. For an illustration of model fit comparison involving IRT models, the reader is referred to Sahu (2002).

Practical Illustration: Fit of 2PL model to University Math Placement Data using WINBUGS 1.4

To provide an example illustration of the above procedures, we consider an item response dataset from a 36-item mathematics placement test. The test is administered each fall to entering freshmen in the University of Wisconsin system to assist with course placement decisions. All items are five option multiple choice items. In this illustration we consider the application of a two-parameter logistic model (2PL), although other models, such as a three-parameter logistic (3PL), may also be appropriate.

Under the 2PL, the probability of a correct response for examinee i to item j is modeled as

$$\mathrm{Prob}(\mathrm{X_{ij}} = 1) = \frac{\exp[\mathrm{a_j}(\theta_i - \mathrm{b_j})]}{1 + \exp[\mathrm{a_j}(\theta_i - \mathrm{b_j})]}, \tag{6}$$

where $\mathrm{X_{ij}}$ denotes the item response (0 = incorrect; 1 = correct), θ_i is an examinee ability parameter, and item parameters $\mathrm{a_j}$ indicate the item discrimination and $\mathrm{b_j}$ the item difficulty. A random sample of 1000 examinees was used for estimation of the model. Consistent with common IRT practice, we focus here on the estimation of the item parameters, where the ability parameters are viewed as nuisance parameters (Patz & Junker, 1999b). However, it is important to note that ability parameters could also be estimated in a similar fashion.

Specification of priors

To estimate the 2PL using MCMC, priors must first be specified for all item and person parameters. We assume θ_i ~Normal (0,1) for all persons i, and $\mathrm{a_j}$ ~LogNormal(0,.5) and $\mathrm{b_j}$ ~Normal (0,2) for all items j. These are similar to priors commonly used in BILOG. Less or more informative priors and/or priors of different distributional forms could naturally also have been chosen. Each item response is assumed to be a Bernoulli outcome with probability of correct response determined by the 2PL model and corresponding person and item parameters.

Sampling procedure

As noted, the program WINBUGS has the potential to implement several different sampling algorithms depending on which method is most efficient for a given application. The appendix displays WINBUGS code for the current analysis. The model and priors selected are not conjugate priors and thus do not lead to conditional distributions that would permit direct Gibbs sampling. In addition, the characteristics of the conditional distributions vary depending on the type of parameter. Consequently, different forms of sampling will be implemented for different parameter types. As normal priors were assumed for both the ability (θ) and difficulty (b) parameters, each of these parameter types results in a conditional distribution that is log concave, allowing WINBUGS 1.4 program to implement an *adaptive rejection sampling* (ARS) algorithm. (For details on this procedure, see Gilks & Wild, 1992). However, the conditional distributions for the item discrimination parameters (a) do not possess this property. Nevertheless, a more efficient algorithm than Metropolis Hastings is also available due to the restricted range of values for these parameters (due to the log-normal prior, they must always be positive). WINBUGS implements a *slice sampling algorithm* in which observations are sampled uniformly from the domain of its conditional probability density. Details of this procedure are provided by Neal (2003). It should be noted that the selection of each of these algorithms occurs through a process internal to WINBUGS, and so need not be specified by the user.

Despite the use of different methods of sampling for different parameters in the model, both approaches seek to produce chains that provide observations that reproduce the joint posterior distribution of the model parameters. Consequently, the general process by which the chain is monitored and estimates determined is not affected by these different sampling appraoches.

Monitoring the chain
Using a randomly generated starting state, A Markov chain for the 2PL was run out to 10,000 states for five different chains. Convergence was examined both through visual inspection of the sampling histories of the chains as well as computation of convergence diagnostics. Figure 3 illustrates the state histories for the discrimination and difficulty parameters of items 1 and 2 from chain 1, both of which appear to display relatively quick convergence to a stationary distribution (Similar results were observed for the other items and chains). Similarly, an overlay of the sampling histories of parameters for the five chains (not shown here) further supported convergence, as between-chain variability relative to

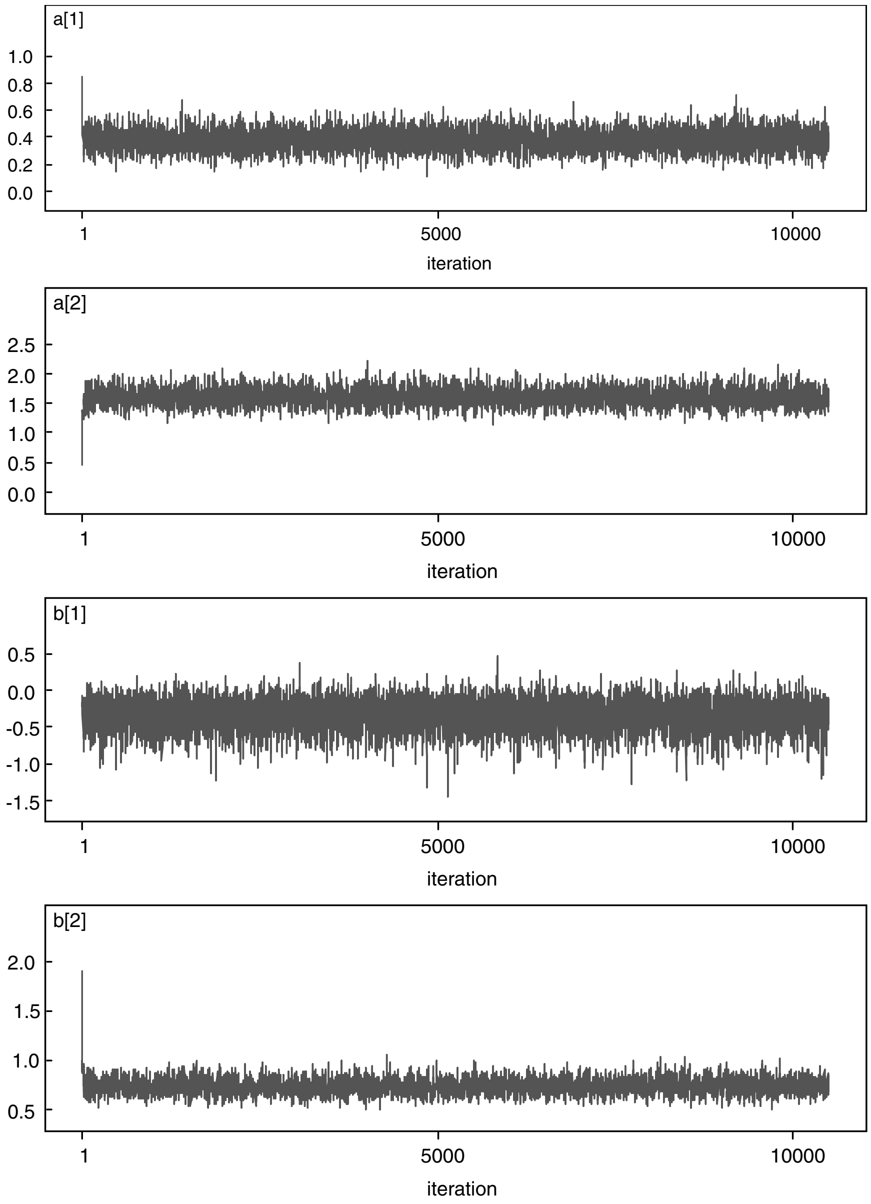

FIGURE 3. Chains for Discrimination and Difficulty Parameters, Items 1 and 2 of Math Placement Data.

within-chain variability appeared minimal. Each chain took approximately 3 hours to complete.

Further confirmation of convergence was obtained from convergence diagnostics. The Geweke (1992) criterion returned z-scores between ±2 for all but one of the discrimination parameters (item 20: -2.03), and for the difficulty parameters was always between ±2. Similarly, the Raftery and Lewis (1992a) diagnostic returned I indices less than 5 for all discrimination and difficulty parameters, with a maximum value of 3.16 for the difficulty parameters (Item 22), and a maximum value of 3.34 for the discrimination parameters (Item 10). Lastly, the Gelman and Rubin (1992) criterion returned point estimates of $\sqrt{\hat{R}} = 1.0$ for each of the 72 difficulty and discrimination parameters, suggesting high similarity across chains. Overall, it therefore appears there is good evidence for the convergence of the chain.

Inspecting posterior distributions

Based on the Raftery and Lewis diagnostics and autocorrelation estimates, a conservative burn-in of 500 was used for all parameters. The Raftery and Lewis (1992b) diagnostic returned a maximum burn-in recommendation of 21 (for the a parameter of item 10), while the largest autocorrelation for samples 10 states apart was.16 (for the b parameter of item 36) and essentially 0 for all samples 50 states apart. Due to the overall length of the chain, the use of 500 as burn-in, although conservative, comes at little cost.

Figure 4 illustrates the resulting marginal posterior distributions from the 10,000 iterations of chain 1 for the difficulty and discrimination parameters of items 1 and 2. The unimodal, symmetric shape of the posteriors suggests that the posterior mean likely provides a good point estimate of the model parameters. The standard deviations of the marginal posterior distributions provide standard errors for these estimates. Table 2 shows the final estimates of model parameters for the 36 items, as well as the corresponding ML estimates computed using the program BILOG and the same priors. The MCMC columns also provide an estimate of the Monte Carlo error based on the number of iterations of each chain. As can be seen from Table 1, the MCMC and MML estimates are virtually identical across items. For a detailed review of studies comparing ML versus MCMC-based point estimation of IRT model parameters, see Rupp, Dey and Zumbo (2004).

Model fit and comparison

To examine issues related to model comparison and fit, a posterior predictive check was applied using an odds ratio (OR) discrepancy statistic. As noted earlier, such a statistic permits inspection of local independence at the item pair level. For a given pair of items, the index is defined as

$$OR = \frac{n_{11}n_{00}}{n_{10}n_{01}}, \tag{7}$$

where the "n"s denote the number of examinees obtaining a given sequence of scores for a given item pair, and the subscripts identify the pattern (e.g., n_{10} indicates the number of examinees answering the first item in the pair correctly and the second item incorrectly). As this test statistic is defined for any item pair, there exist a total of $36\times35/2 = 630$ different OR statistics that can be studied using this posterior predictive check. In the current analysis, 24 of the statistics displayed a p-value less than .05, 15, of which produce values more positive than expected. For example, one such item pair was items 3 and 6, where the observed $OR = 2.52$, while a 95% confidence interval (CI) based on replicate data had lower and upper bounds of 1.36 and 2.44, respectively. For items 16 and 20, the $OR = 1.37$, with 95% $CI = 1.38, 2.50$. So overall in terms of local independence, the fit of the 2PL seems to be reasonably good, with significant pairs generally falling just outside the region of nonsignificance. As noted earlier, there are many other aspects of fit that could be evaluated using different discrepancy statistics.

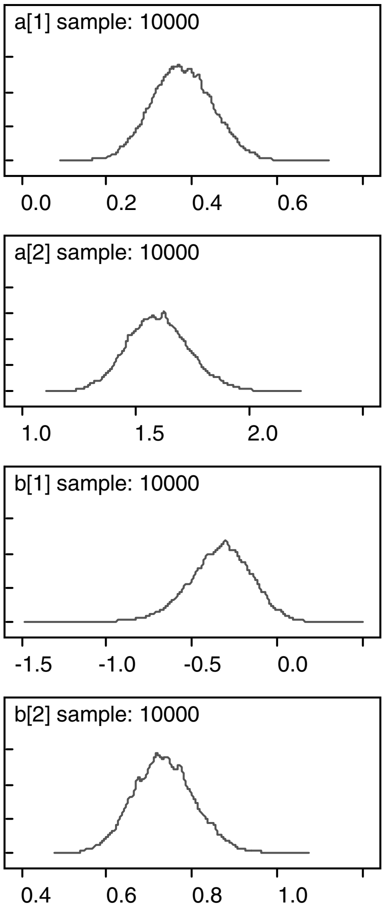

FIGURE 4. Marginal Posterior Density for Item Discrimination and Difficulty Parameters, Items 1 and 2, Math Placement Data

To evaluate the 2PL in comparison to the one-parameter logistic model (1PL), the 1PL model was also fit to the same data, essentially following the same process used above in estimating the 2PL. The 1PL models the probability of correct response as

$$\mathrm{Prob}(X_{ij} = 1) = \frac{\exp[a(\theta_i - b_j)]}{1 + \exp[a(\theta_i - b_j)]}, \tag{8}$$

with the primary difference from the 2PL being the application of a common discrimination parameter across all items. In estimating the model, the same priors as for the 2PL were imposed for the difficulty parameters and the one discrimination parameter.

As in the 2PL, an "a" parameter was estimated, but was assumed equal across all items.

Table 1. Markov Chain Monte Carlo (MCMC) and Marginal Maximum Likelihood (MML) 2PL Model Parameter Estimates, Math Placement Data

	MCMC Estimates						MML Estimates			
Item	a	se	mcse	b	se	mcse	a	se	b	se
1	.38	.07	.001	-.34	.19	.002	.41	.06	-.30	.16
2	1.60	.14	.002	.74	.07	.001	1.57	.13	.75	.07
3	.88	.10	.002	-.62	.10	.002	.87	.09	-.61	.09
4	.54	.08	.001	.24	.13	.002	.56	.07	.23	.12
5	.40	.07	.001	1.16	.27	.006	.42	.07	1.09	.23
6	.92	.09	.001	.19	.08	.001	.92	.09	.20	.08
7	.99	.10	.002	.18	.08	.001	.98	.09	.19	.07
8	.70	.09	.001	-.31	.11	.001	.71	.08	-.29	.10
9	1.34	.12	.002	-.27	.06	.001	1.32	.12	-.26	.06
10	.52	.08	.002	2.11	.32	.009	.52	.08	2.10	.31
11	.89	.10	.003	1.94	.20	.005	.87	.10	1.97	.20
12	.72	.09	.002	1.05	.15	.003	.72	.08	1.06	.14
13	.73	.09	.001	-.71	.12	.002	.74	.09	-.69	.11
14	.94	.09	.001	.64	.10	.002	.93	.09	.65	.09
15	.97	.10	.001	.30	.08	.001	.97	.09	.31	.08
16	.66	.08	.001	.60	.13	.002	.67	.08	.59	.12
17	.46	.07	.001	-.34	.16	.002	.49	.07	-.31	.14
18	.81	.10	.003	2.07	.24	.007	.80	.09	2.07	.22
19	.60	.08	.001	.60	.14	.003	.61	.08	.59	.13
20	1.38	.12	.002	.38	.07	.001	1.36	.11	.40	.06
21	.53	.08	.002	1.68	.26	.007	.54	.07	1.65	.24
22	.59	.08	.002	1.54	.24	.006	.60	.08	1.52	.21
23	.90	.10	.002	-.99	.12	.002	.89	.10	-.98	.11
24	.51	.07	.001	.82	.18	.003	.53	.07	.80	.16
25	.69	.08	.002	1.10	.16	.003	.69	.08	1.10	.14
26	.82	.09	.001	.52	.10	.002	.82	.08	.53	.10
27	.96	.10	.002	.86	.11	.002	.95	.09	.87	.10
28	1.11	.10	.001	.29	.08	.001	1.10	.10	.30	.07
29	1.00	.10	.002	-.82	.10	.002	1.00	.10	-.81	.09
30	.73	.09	.001	.60	.12	.002	.74	.08	.60	.11
31	.95	.10	.002	1.17	.13	.002	.94	.09	1.18	.12
32	.86	.10	.002	1.66	.18	.004	.85	.09	1.68	.18
33	.75	.09	.001	-.42	.10	.002	.76	.08	−.40	.10
34	1.33	.12	.002	.16	.07	.001	1.31	.12	.17	.06
35	.70	.08	.001	.53	.12	.002	.71	.08	.53	.11
36	1.06	.10	.002	1.24	.12	.003	1.04	.10	1.25	.12

a = item discrimination estimate.
b = item difficulty estimate.
se = standard error.
mcse = Markov chain standard error.

Both the DIC and Pseudo-Bayes factor criteria were evaluated for the 2PL and 1PL. In both cases, the 2PL is shown to be preferred to the 1PL. For the DIC index, the 2PL returned DIC = 42,578 (the effective number of parameters, $_{p}D$ = 901) while the 1PL returned DIC = 42,921 ($_{p}D$ = 847). In terms of the Pseudo-Bayes factor criterion, the 1PL returned a log(CPO) = −21,463 across item responses, while the 2PL returned a log(CPO) = −21,294. Consequently, the criteria agree in the selection of the 2PL over the 1PL model for these data.

Concluding Comments

Models such as the 2PL can usually be easily fit using MCMC methods. As noted in the introduction, however, the MCMC methodology perhaps finds its greatest appeal in its ability to accommodate more complex IRT models for which ML-based IRT estimation software (like BILOG) is unavailable. Such applications often present new challenges, however; we mention a couple of common problems from our experience. One is the occurrence of sampling "traps." A trap occurs when a state in the chain is encountered from which the possible domain from which a new state can be sampled is so restricted that no eligible sample is produced, even after a large number of attempts. The occurrence of a trap causes WINBUGS to terminate, and often requires restarting the chain from a new starting state. Other challenges are related to the frequent need for stricter identifiability constraints than are applied when using other estimation procedures. For example, use of MCMC with multidimensional and/or mixture IRT models can lead to dimension/class "identity switching" over the course of a simulated chain. Identity switching refers to the situation where one dimension or class comes to represent another, making the estimation of dimension-specific parameters problematic. For example, in a two-dimensional item response model, it can happen that at some point in the chain, what was originally ability dimension 1 assumes the role of ability dimension 2, thus creating an inconsistency in how the item discrimination values should be interpreted over the course of the chain.

For these and other reasons, practitioners new to the MCMC methodology may find it useful to experiment with simple models in the context of MCMC before attempting more complex ones. A "model-building" strategy that begins with a simplified version of the IRT model of interest may, where possible, provide a better starting point for implementing MCMC procedures to ensure identifiability of individual parameters. Readers are encouraged also to examine a variety of different MCMC applications, including those outside of IRT, for a better understanding of the methodology, as well as potential challenges in implementing MCMC. For example, the uses of MCMC methods for de-convolving mixtures (Robert, 1997) or in fitting hierarchical models (Clayton, 1997) have been frequently studied, and in ways that would likely be informative for IRT models generalized to accommodate similar types of structure.

Finally, we note that while this paper has focused on the WINBUGS software for estimating MCMC models, a variety of different packages can also be used. Essentially any statistical software that can be used to simulate data from specified distributions can be used. Patz and Junker (1999b) provide code for implementing MCMC procedures with IRT models using the program S-plus. Such applications often help considerably in reducing the computational time in conducting MCMC, but demand a little more in terms of programming on the part of the user.

Self-test

1. Suppose that a drug screening test used to disqualify riders in a long-distance bicycle race is 95% accurate in detecting users of the illegal drug, and is 99% accurate in detecting nonusers of the drug. Suppose it is further known that 3% of the cyclists use the drug. Following Bayes' theorem, identify

   (a) the prior distribution for use of the drug
   (b) the posterior probability that a cyclist is actually a user if the test is positive
   (c) the posterior probability that a cyclist is not a user if the test is negative

2. Which of the potential priors for the difficulty parameters in an IRT model: N(−1, 1), N(−1, 20) or N(−1,.2) would likely have the greatest influence on the final item difficulty estimates?

3. True/False questions

   (a) In MCMC estimation, the Markov chain for a given parameter should eventually converge to a single point value.
   (b) A primary advantage of MCMC methods with IRT models is that they are easier to implement than ML methods, although generally take a long time to run.
   (c) The distributional form selected for the prior(s) matters little in determining an appropriate MCMC sampling algorithm.
   (d) If an MCMC algorithm fails to converge for a given model and data set, there must be something wrong with how the sampling procedure was programmed.
   (e) A fundamental difference between Bayesian approaches (like MCMC) and frequentist approaches (like ML) to model estimation is the emphasis in Bayesian estimation on estimating posterior distributions rather than point estimates of model parameters.
   (f) It is possible to study different aspects of IRT model fit through the use of different discrepancy statistics using posterior predictive checks.
   (g) If after a single MCMC runs a second MCMC chain were run using a different starting value and produced a different sampling history than the original chain, the original chain has not converged.

4. In evaluating the fit of the 2PL model to his item response data set, test practitioner Phil approximates the Bayes factor for the 2PL versus 1PL to be 1340. Is Phil correct in assuming that the 2PL therefore provides a close fit to his item response data?
5. Suppose that after 1,000 iterations an MCMC run has converged. If the number of MCMC iterations were then increased by 100 times to 100,000, which should decrease, the Monte Carlo standard error (MCSE) or the point estimate standard error (SE)? By how much is it expected to decrease?
6. Suppose test practitioner Janet estimates a model using MCMC and ML, and finds noticeable differences in the final point estimates of the model despite having used the same priors in both estimation runs. She rechecks the MCMC algorithm, and is confident of convergence. What might explain the difference in the results?

Answers to Self-Test

1. (a) Because the distribution is only defined over a domain of two events, the prior can be expressed as P(user) =.03; P(nonuser) = .97.
   (b) Using Bayes' theorem, P(user | test = positive) = P(test = positive | user) P(user)/[P(test = positive | user) P(user)+ P(test = positive | nonuser) - P(nonuser)] = .95 ×.03 / (.95×.03 +.01×.97) = .75
   (c) Using Bayes theorem, P(nonuser | test = negative) = P(test = negative | nonuser) P(nonuser)/[P(test = negative | user) P(user)+ P(test = negative | nonuser)−P(nonuser)] = .99 ×.97 / (.05×.03 +.99×.97) >.99
2. The N(−1,.2) prior would have the strongest effect, as it has the smallest variance. It would generally also produce parameter estimates with the smallest standard errors.
3. (a) F, the chain should converge to a stationary distribution of values, not a single value.

(b) T, many MCMC algorithms take only minutes to program, but when used in practice can take hours or more to run.

(c) F, the form of the prior defines characteristics of the posterior that can often permit more efficient MCMC sampling techniques.

(d) F, model misspecification and/or a lack of information in the data could also be responsible.

(e) T, in MCMC we estimate marginal posterior distributions for the model parameters; in ML we determine point estimates of the model parameters.

(f) T, posterior predictive checks provide a very flexible set of techniques for investigating any form of misfit that can be captured by a discrepancy statistic.

(g) F, the two chains should converge to a common distribution, but their sampling histories will likely be different.

4. No, the Bayes factor criterion, and approximations to it, provide a criterion for model comparison, not the evaluation of model fit in an absolute sense. Phil would be correct in claiming the 2PL provides a better relative fit than the 1PL, but that does not necessarily mean that the 2PL model provides a close fit to the data.
5. The Monte Carlo standard error (MCSE) should decrease, not the point estimate standard error (SE). Only the MCSE is systematically affected by the size of the sample (i.e., the number of MCMC iterations). Because the MCSE reflects the standard error of the sample mean, we can use the standard error of the mean formula in approximating the expected reduction in the MCSE: $\sigma/\sqrt{1{,}000} = \sqrt{100} \times \sigma/\sqrt{100{,}000}$, implying a 10 times smaller MCSE.
6. When point estimates are derived from an MCMC run, they are generally based on a characteristic of the posterior distributions of the model parameters, often the posterior mean. The posterior mean may not be the value producing the maximum likelihood, especially when the posterior distribution is asymmetric. So the estimates likely differ because they reflect different characteristics of the posterior distributions.

References

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In B.N. Petrov & F. Csáki (Eds.), *Proceedings, 2nd International Symposium on Information Theory* (pp. 267–281). Budapest: Akadémiai Kiadó.

Albert, J.H. (1992). Bayesian estimation of normal ogive item response curves using Gibbs sampling. *Journal of Educational Statistics*, *17*, 251–269.

Baker, F.B. (1998). An investigation of the item parameter recovery characteristics of a Gibbs sampling approach. *Applied Psychological Measurement*, *22*, 153–169.

Baker, F.B., & Kim, S-H. (2004). *Item response theory: Parameter estimation techniques* (*2nd ed.*). New York: Marcel Dekker.

Beguin, A.A., & Glas, C.A.W. (2001). MCMC estimation and some model-fit analysis of multidimensional IRT models. *Psychometrika*, *66*, 541–562.

Best, N., Cowles, M.K., & Vines, K. (1996). *CODA*: Convergence diagnosis and output analysis software for gibbs sampling output, version 0.30*. Cambridge, UK: MRC Biostatistics Unit.

Bolt, D.M., & Lall, V.F. (2003). Estimation of compensatory and noncompensatory multidimensional item response models using Markov chain Monte Carlo. *Applied Psychological Measurement*, *29*, 395–414.

Bradlow, E.T., Wainer, H., & Wang, X. (1999). A Bayesian random effects model for testlets. *Psychometrika*, *64*, 153–168.

Clayton, D.G. (1997). Generalized linear mixed models. In W.R. Gilks, S. Richardson, & D.J. Spiegelhalter (Eds.), *Markov chain Monte Carlo in practice* (pp. 275–302). London: Chapman & Hall.

Congdon, P. (2001). *Bayesian statistical modeling*. Chichester, England: John Wiley & Sons.

De la Torre, J., Stark, S., & Chernyshenko, O. (2006). Markov Chain Monte Carlo estimation of item parameters for the generalized graded unfolding model. *Applied Psychological Measurement*, *30*, 216–232.

Fox, J.-P., & Glas, C.A.W. (2001). Bayesian estimation of multilevel IRT models using Gibbs sampling. *Psychometrika*, *66*, 271–288.

Geisser, S., & Eddy, W. (1979). A predictive approach to model selection. *Journal of the American Statistical Association*, *74*, 153–160.

Gelfand, A.E., Dey, D.K., & Chang, H. (1992). Model determination using predictive distributions with implementation via sampling methods. In J.M. Bernardo, J.O. Berger, A.P. Dawid, & A.F.M. Smith (Eds.), *Bayesian statistics* (Vol. 4, pp. 147–167). Oxford, UK: Oxford University Press.

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

Geweke, J. (1992) Evaluating the accuracy of sampling-based approaches to calculating posterior moments. In J.M. Bernardo, J.O. Berger, A.P. Dawid, & A.F.M. Smith (Eds.), *Bayesian statistics* (Vol. 4, pp. 169–193). Oxford, UK: Oxford University Press.

Gilks, W.R., & Wild, P. (1992). Adaptive rejection sampling for Gibbs sampling. *Applied Statistics*, *41*, 337–348.

Gilks, W.R., Richardson, S., & Spiegelhalter, D.J. (1996). *Markov chain Monte Carlo in practice*. Boca Raton, FL: Chapman & Hall.

Glas, C. A. W., & Meijer, R. R. (2003). A Bayesian approach to person fit analysis in item response theory models. *Applied Psychological Measurement*, *27*, 217–233.

Jeffreys, H. (1961). *Theory of probability*. London: Oxford University Press.

Johnson, M.S., & Sinharay, S. (2005). Calibration of polytomous item families using Bayesian hierarchical modeling. *Applied Psychological Measurement*, *29*, 369–400.

McLeod, L., Lewis, C, & Thissen, D. (2003). A Bayesian method for the detection of item preknowledge in computerized adaptive testing. *Applied Psychological Measurement*, *27*, 121–137.

Mislevy, R.J., & Bock, R.D. (1989). *BILOG 3: Item analysis and test scoring with binary logistic models*. Mooresville, IN: Scientific Software.

Neal, R.M. (2003). Slice sampling. *Annals of Statistics*, *31*, 705–767.

Patz, R., & Junker, B. (1999a). Applications and extensions of MCMC in IRT: Multiple item types, missing data, and rated responses. *Journal of Educational and Behavioral Statistics*, *24*, 342–366.

Patz, R., & Junker, B. (1999b). A straightforward approach to Markov chain Monte Carlo methods for item response models. *Journal of Educational and Behavioral Statistics*, *24*, 146–178.

Raftery, A.E. (1996). Hypothesis testing and model selection. In W.R. Gilks, S. Richardson, & D.J. Spiegelhalter (Eds.), *Markov chain Monte Carlo in practice* (pp. 115–130). London: Chapman & Hall.

Raftery, A.E., & Lewis, S.M. (1992a). How many iterations in the Gibbs sampler? In J.M. Bernardo, J.O. Berger, A.P. Dawid, & A.F.M. Smith (Eds.), *Bayesian statistics* (Vol. 4, pp. 765–776). Oxford, UK: Oxford University Press.

Raftery, A.E., & Lewis, S.M. (1992b). One long run with diagnostics: implementation strategies for Markov chain Monte Carlo. *Statistical Science*, *7*, 493–497.

Robert, C.P. (1997). Mixtures of distributions: Inference and estimation. In W.R. Gilks, S. Richardson, & D.J. Spiegelhalter (Eds.), *Markov chain Monte Carlo in practice* (pp. 441–464). London: Chapman & Hall.

Rupp, A.A., Dey, D.K., & Zumbo, B.D. (2004). To Bayes or not to Bayes, from whether to when: Applications of Bayesian methodology to modeling. *Structural Equation Modeling*, *11*, 424–451.

Sahu, S.K. (2002). Bayesian estimation and model choice in item response models. *Journal of Statistical Computation and Simulation*, *72*, 217–232.

Schwartz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, *6*, 461–464.

Sinharay, S. (2004). Experiences with Markov chain Monte Carlo convergence assessment in two psychometric examples. *Journal of Educational and Behavioral Statistics*, *29*, 461–488.

Sinharay, S. (2005). Assessing fit of unidimensional item response theory models using a Bayesian approach. *Journal of Educational Measurement*, *42*, 375–395.

Sinharay, S., Johnson, M.S., & Stern, H.S. (2006). Posterior predictive assessment of item response theory models. *Applied Psychological Measurement*, *30*, 298–321.

Spiegelhalter, D.J., Best, N.G., Carlin, B.P., & van der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society, Series B*, *64*, 583–604.

Spiegelhalter, D.J., Thomas, A., Best, N., & Gilks, W. (1995). *BUGS 0.5*: Bayesian inference using Gibbs sampling manual (version ii)*. Cambridge, UK: MRC Biostatistics Unit.

Spiegelhalter, D.J., Thomas, A., Best, N., & Lunn, D. (2003). *WINBUGS Version 1.4 User's manual* [Computer software manual]. Cambridge, UK: MRC Biostatistics Unit.

Thissen, D. (1991). *MULTILOG 6.0*. Chicago: Scientific Software.

Zwick, R., Thayer, D.T., & Lewis, C. (2000). Using loss functions for DIF detection: An empirical Bayes approach. *Journal of Educational and Behavioral Statistics*, *25*, 225–247.

Appendix: WINBUGS code for 2PL model, Math Placement Data

```
Model
        {
# Read in individual item responses
        for (i in 1:1000) {
                for (j in 1:36) {
                        x[i,j] <- response[i,j]
                          }
                    }

# Identify two-parameter logistic (2PL) model
        for (i in 1 : 1000) {
                for (j in 1 : 36) {
                        p[i, j] <- exp(a[j]*(theta[i] - b[j]))/(1+exp(a[j]*(theta[i] - b[j])))
                        x[i, j]~dbern(p[i,j])
                        mx[i,j]<-1-r[i,j]
                           }
#Specify prior for examinee parameters
                theta[i] ~ dnorm(0,1)
                     }

# Specify priors for item parameters
        for (j in 1:36){
                a[j]~dlnorm(0,2)
                b[j]~dnorm(0,.5)
               }

# Generate replicate data for posterior predictive checks
        for (i in 1001 : 2000) {
                for (j in 1 : 36) {
                        p[i, j] <- exp(a[j]*(theta[i] - b[j]))/(1+exp(a[j]*(theta[i] - b[j])))
                        r[i, j]~dbern(p[i,j])
                        mr[i,j]<-1-r[i,j]
                           }
                theta[i] ~ dnorm(0,1)
                        }
# Compute odds ratios for replicate data
        for (k in 2:36) {
                for (l in 1:(k-1)) {
                or[k,l]<-inprod(r[1001:2000,l],r[1001:2000,k])*inprod(mr[1001:2000,l],mr[1001:2000,k])/
                (inprod(mr[1001:2000,l],r[1001:2000,k])*inprod(r[1001:2000,l],mr[1001:2000,k]))
                           }
                 }


#Compute CPO for observed item responses
        for (j in 1:1000){
                for (k in 1:36){
                        prob[j,k]<-pow(p[j,k],r[j,k])*pow((1-p[j,k]),(1-r[j,k]))
                        inprob[j,k]<-pow(prob[j,k],-1)
                       }
                  }

        }
# End of model


#Listing of item response data

list(response=structure(.Data=c(
0,0,1,1,0,1,1,1,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,1,0,1,1,1,1,0,0,0,1,0,0,1,0,0,1,1,0,1,1,1,1,1,0,0,1,1,0,0,1,1,1,0,
1,0,1,1,1,1,1,1,1,0,0,1,1,1,1,1,0,0,1,0,1,1,1,0,0,1,0,1,0,1,1,1,1,1,1,1,
        .       .       .       .       .       .
        .       .       .       .       .       .
        .       .       .       .       .       .
1,0,1,0,1,0,1,1,1,1,1,0,1,0,1,0,0,1,0,1,1,0,1,1,1,1,1,0,1,1,1,0,0,1,0,1), .Dim=c(1000,36)))
```