Psicológica (2017), 38, 25-55.

Bayesian estimation of multidimensional item response models. A comparison of analytic and simulation algorithms

Manuel Martin-Fernandez[1] & Javier Revuelta[2]

[1] *Universitat de Valencia. Valencia, Spain.*

[2] *Universidad Autónoma de Madrid. Madrid, Spain.*

This study compares the performance of two estimation algorithms of new usage, the Metropolis-Hastings Robins-Monro (MHRM) and the Hamiltonian MCMC (HMC), with two consolidated algorithms in the psychometric literature, the marginal likelihood via EM algorithm (MML-EM) and the Markov chain Monte Carlo (MCMC), in the estimation of multidimensional item response models of various levels of complexity. This paper evaluates the performance of parameter recovery via three simulation studies from a Bayesian approach. The first simulation uses a very simple unidimensional model to evaluate the effect of diffuse and concentrated prior distributions on recovery. The second study compares the MHRM algorithm with MML-EM and MCMC in the estimation of an item-response model with a moderate number of correlated dimensions. The third simulation evaluates the performance of the MHRM, HMC, MML-EM and MCMC algorithms in the estimation of an item response model in a high-dimensional latent space. The results showed that MML-EM loses precision with high-dimensional models whereas the other three algorithms recover the true parameters with similar precision. Apart from this, the main differences between algorithms are: 1) estimation time is much shorter for MHRM than for the other algorithms, 2) MHRM achieves the best precision in all conditions and is less affected by prior distributions, and 3) prior distributions for the slopes in the MCMC and HMC algorithms should be carefully defined in order to avoid problems of factor orientation. In summary, the new algorithms seem to overcome the difficulties of the traditional ones by converging faster and producing accurate results.

---

[1] **Acknowledgments.** We would like to thank the two anonymous reviewers for their insightful comments, which improved greatly the quality of this manuscript. This research has been founded with the project PSI2012-31958 of the Spanish Dirección General de Investigación Científica y Técnica, Ministerio de Economía y Competitividad. **Corresponding author**: Manuel Martin-Fernandez. E-mail: manuel.martinf@inv.uam.es

1. Introduction

Multidimensional item response models apply to the investigation of the latent factors underlying psychological tests and questionnaires composed of dichotomously scored items, or items with few response categories. These models are equivalent to a categorical factor analysis and thus are informative about the number and composition of latent factors as well as the relations between them (McDonald, 1985).

The estimation of models with five or more factors is a usual demand of exploratory and confirmatory analyses. However, the selection of a reliable and fast estimation algorithm is an open problem in the practical application of multidimensional item response models. A number of alternatives exist, from limited information algorithms based on tetrachoric correlations (Christofferson, 1975) and marginal/EM estimation (Bock & Aitkin, 1981), to Bayesian MCMC estimation (Gelman, Carlin, Stern & Rubin, 1995). Nevertheless, while these algorithms perform well in low dimensional models, they can easily run into difficulties in high dimensional latent spaces. This problem is often referred to in the literature as *the curse of dimensionality* (Cai, 2010a), because the complexity of the integration problem involved in estimation has an exponential growth rate in relation to the number of factors.

Apart from the technical difficulties of integration over the latent space, complex models may have weakly identifiable parameters, which are those parameters that are identified from a purely algebraic analysis of the model structure, but the sample contains little information to estimate them. These parameters present difficulties of convergence and estimation of standard errors. Even more so, the presence of weakly identifiable parameters may transmit uncertainty in the estimation of the other parameters and impede the reliable estimation of the whole model. Estimation problems originated by weak identification can be alleviated or eliminated by imposing Bayesian priors on item parameters. Because high dimensional models will be used in this paper, inference will be performed in the Bayesian framework.

There are two broad classes of Bayesian estimation algorithms: analytic and simulation. On the one hand, analytic algorithms are based on the explicit mathematical derivation of estimation equations, which involve a Gauss-Hermite numerical integration procedure and a Newton-Raphson algorithm to find the roots of the estimation equations (Schilling & Bock, 2005). On the other hand, simulation algorithms consist of taking samples of parameters from the posterior distribution using a Markov chain Monte Carlo algorithm implemented via Gibbs sampling (Gilks, Richardson &

Spiegelhalter, 1996). Simulation algorithms avoid the use of derivatives and are mathematically simpler at the cost of an increased computational burden. There are also some algorithms that are a hybrid of these two classes, such as the stochastic EM algorithm (Diebolt & Ip, 1996; Wirth & Edwards, 2007).

The purpose of this study is to gather information about the performance of two recent Bayesian estimation algorithms in comparison to two algorithms consolidated in the psychometric literature. The recent algorithms were introduced to overcome the difficulties of the traditional ones, although few studies comparing performance have been published yet. There is still little evidence about the supposed benefits of the new methods because of their novelty. The four algorithms considered in this paper are:

1. Marginal likelihood (MML-EM; Bock & Aitkin, 1981), which is based on an analytic differentiation of the log-likelihood. The MML-EM proceeds iteratively in two steps: in the first step, the algorithm computes the distribution of the factors conditional on the item responses; in the second step item parameters are estimated while keeping fixed the conditional distribution obtained in the first step. The computation of the conditional distribution of the factors involves the marginal distribution of item responses, which is approximated by a number of methods such as static Gauss-Hermite quadrature, adaptive Gauss-Hermite or Monte Carlo simulation (Schilling and Bock, 2005).
2. Bayesian simulations via Markov chain Monte Carlo (MCMC; Gilks et. al., 1996). MCMC algorithms take samples from a target posterior distribution. The algorithm creates several Markov chains in parallel whose stationary distribution is the posterior distribution of interest. Once the chain has converged to the stationary distribution, the samples from the chains will behave approximately like samples from the posterior distribution.
3. Metropolis-Hastings Robbins-Monro (MHRM; Cai, 2010a, 2010b). The MHRM is a hybrid algorithm based on marginal likelihood in which samples from the conditional distribution of the factors are combined via stochastic approximation. It produces maximum-likelihood and modal or expected a-posteriori point-estimate solutions for multidimensional item response models, avoiding the Gauss-Hermite numerical integration procedure.
4. Hamiltonian MCMC (HMC; Neal, 2011). The HMC method speeds up convergence of MCMC simulations by applying the Hamiltonian dynamics (Neal, 2011). However this decrement in the estimation time requires, first, to compute the gradient of the log-posterior and, second,

to specify the number of steps and the steps' size to run a Hamiltonian system. Hoffman and Gelman (2014) propose the No-U-Turn sampler (NUTS) to make the HMC procedure decisions more convenient for the user.

The two classical algorithms are MML-EM with Gauss-Hermite numerical integration and MCMC via Gibbs-sampling, whereas MHRM and HMC are recent evolutions of them aimed at reducing the computational burden and speeding up convergence. Apart from the analytic versus simulation issue, these algorithms differ in the estimates they provide. MML-EM and MHRM maximize the posterior distribution of item parameters and provide the modal a-posteriori estimate (MAP). However, MCMC and the HMC provide a simulation approximation to the full posterior distribution of item parameters, which can be summarized in the expected a-posteriori estimate (EAP).

The model used in this paper applies to dichotomous data. The probability that individual *j* gives a positive response to item *i* is given by

$$P_{ij} = \frac{\exp(d_i + a_{i1}\theta_{j1} + \cdots + a_{iD}\theta_{jD})}{1 + \exp(d_i + a_{i1}\theta_{j1} + \cdots + a_{iD}\theta_{jD})}, \quad (1)$$

where D is the number of factors, θj_1, …, θj_D are person parameters, d_i is the item intercept, and a_{i1}, …, a_{iD} are the item slopes. Apart from these parameters, the model includes the factor variances, σ_j^2, and covariances, σ_{jk}. Not all of these parameters can be estimated simultaneously, and some of them have to be fixed to constant values for the others to be identifiable. For example, one item slope, a_{ij}, has to be fixed to a constant value for the factor variance of the corresponding factor, σ_j^2, to be identifiable. The model in Equation (1) is equivalent for all practical purposes to a factor analysis of dichotomous variables.

Let **X** be the matrix of observed responses, MML-EM and MHRM provide a point estimate by maximizing the posterior distribution:

$$f(\boldsymbol{a},\boldsymbol{d},\boldsymbol{\sigma} \mid \mathbf{X}) \propto f(\boldsymbol{a},\boldsymbol{d},\boldsymbol{\sigma}) \int P(\mathbf{X} \mid \boldsymbol{a},\boldsymbol{d},\boldsymbol{s},\boldsymbol{\theta}) f(\boldsymbol{\theta}) d\boldsymbol{\theta}, \quad (2)$$

where $P(\mathbf{X} \mid \boldsymbol{a},\boldsymbol{d},\boldsymbol{s},\boldsymbol{\theta}) = \prod_i \prod_j P_{ij}^{x_{ij}} (1 - P_{ij})^{1 - x_{ij}}$ is the probability of the observed data conditional on all parameters, $f(\boldsymbol{a},\boldsymbol{d},\boldsymbol{\sigma})$ is the prior distribution and $f(\boldsymbol{\theta})$ is a multivariate normal density function. The purpose of MCMC and HMC is to take samples from $f(\boldsymbol{a},\boldsymbol{d},\boldsymbol{\sigma} \mid \mathbf{X})$; these

samples can be summarized using central tendency and dispersion measures to obtain Bayesian point estimates and associated standard errors.

In this research, the four algorithms are applied in a Bayesian context with prior distributions for all parameters. In this context, the label MML-EM means that the likelihood function involved in the posterior distribution in Equation (2) is a marginal likelihood: $\int P(\mathbf{X}|\boldsymbol{a},\boldsymbol{d},\boldsymbol{s},\boldsymbol{\theta})f(\boldsymbol{\theta})d\boldsymbol{\theta}$.

The four algorithms were applied to models from low to high complexity to test them in a number of conditions of realistic complexity. To this end, the paper reports three simulation studies. The first simulation study investigates the effect of the prior distributions in a unidimensional model. The second simulation study is based on models of an intermediate number of factors and many different types of parameters, whereas a model with many factors is used in the third simulation study.

2. Simulation study one. Estimation of a unidimensional model

The first study evaluated the effect of type of prior distribution on the parameter recovery of a relatively simple unidimensional item-response model. In particular, we fitted the two parameter logistic model (Birnbaum, 1968), which is equivalent to Equation (1) when $D = 1$, to a sample test of 15 items. The model was estimated using two analytic algorithms (MML-EM and MHRM).

Table 1 shows the true item parameters of the model. The a_j and d_j parameters were generated at random. The a-parameters (slopes or scale parameters) were obtained from a lognormal distribution (with $\mu = 0$ and $\sigma^2 = 0.5$, yielding a distribution with an expected value of 1.13 and a variance of 0.36). True d-parameters (intercepts) and person parameters (factor scores) were obtained from a standard normal distribution. We selected these distributions to obtain true parameter values comparable to those found in real applications.

Both MML-EM and MHRM algorithms were implemented using the statistical software R (R Core Team, 2015) and version 1.13 of the *mirt* package (Chalmers, 2012). Default estimation options were left for both algorithms (31 quadrature points used per dimension, 500 max EM cycles, convergence occurred when all parameters were less than |.0001| across cycles; 2000 iterations for the MHRM and a burn-in period of 150 iterations).

The version 1.13 of the *mirt* package only allowed the use of the normal and the lognormal prior distributions. However, if no prior was selected and lower and upper bounds are set for the parameters in the MML

algorithm, the resulting estimate is equivalent to a Bayesian estimate with a uniform prior. So we utilize the normal, lognormal and uniform prior to estimate the model.

Table 1. True parameters values for the first simulation study

Item	d_i	a_i
1	-0.41	0.39
2	0.75	0.53
3	1.25	0.75
4	0.31	0.82
5	1.02	1.35
5	0.65	1.19
6	0.25	1.79
7	-0.49	0.83
8	1.31	0.84
9	0.29	1.96
10	0.33	0.8
11	-0.35	1.03
12	0.2	0.53
13	-0.64	2.76
14	-0.68	1.11
15	-0.41	0.39

Person parameters followed a standard normal distribution, which is a commonplace in the IRT context (Curtis, 2010). Nonetheless, the prior distribution for the item parameters varied from scale to intercept parameters. For the scale parameters it was not usual to have negative estimated values in unidimensional models, and thus the prior distribution should be defined in the positive real line only. Not so for the intercept parameters, for which only the uniform or the normal distributions could be a feasible choice.

This leaded to three different configurations of priors using the *mirt* package:

1) Flat little informative distributions:

$$\begin{aligned} a &\sim \text{uniform}\ (0,\ 5) \\ d &\sim \text{uniform}(-5,\ 5) \end{aligned} \tag{3}$$

The prior for the slopes had an expected value of 2.5 and a standard deviation of 1.44, and the prior for the intercepts had expected value of 0, a variance of 2.5 and a standard deviation of 2.88.

2) Item parameters were assumed to follow a normal distribution, being able to take positive and negative values

$$\begin{aligned} a &\sim \text{normal}\ (0,\ 3) \\ d &\sim \text{normal}\ (0,\ 3) \end{aligned} \tag{4}$$

The standard deviation of the normal (0, 3) was 1.73, rendering a relatively flat uninformative distribution.

3) Slopes were assumed to follow a lognormal distribution, taking only positive values, and intercept parameters a normal distribution, taking either positive or negative values

$$\begin{aligned} a &\sim \text{lognormal}\ (0,\ 0.5) \\ d &\sim \text{normal}(0,\ 3) \end{aligned} \tag{5}$$

The lognormal (0, 0.5) distribution has a median of 1, an expectation of 1.13, and a standard deviation of 0.6, which are reasonable values for the a prior; we have fixed the median to 1 instead of the expectation because the lognormal has a remarkable asymmetry and setting the expectation equal to 1 would resulted in a distribution with a thick right tail.

Note that the prior distributions in Equation (5) that were used for estimation were the same as the prior used to generate item parameters. Thus, in this condition the priors are *correct*, whereas the priors in (3) and (4) differed from the generating distributions and might introduce bias in estimation.

The conditions of the simulation study were: estimation algorithm (MML-EM vs. MHRM); the prior distributions (Equation (3), (4) or (5)); and sample size (500 vs. 1000 simulees). This left a total of 12 simulation conditions (two algorithms × three priors × two sample sizes). One hundred samples were generated for each condition. Item parameters and true person parameters remained constant across replications, and the response matrices varied from one replication to another, but not across conditions. The two estimation algorithms were applied to the same response matrices to ensure that differences in performance were due solely to the estimation algorithm and not to sampling error in the response matrices or the values of θ.

Parameter recovery was evaluated by the absolute mean bias and the root mean squared error (RMSE) between point estimates and the true parameter values. Correlations between true and estimated parameters were also computed. The results appear in Tables 2 and 3.

Table 2. Parameter recovery statistics for sample size $N = 500$

		MML-EM			MHRM		
		uniform	a-normal	a-lognormal	uniform	a-normal	a-lognormal
Bias	a	0.138	0.133	0.135	0.138	0.134	0.142
	d	0.098	0.097	0.096	0.099	0.098	0.100
	θ	0.378	0.377	0.377	0.379	0.379	0.379
RMSE	a	0.178	0.170	0.172	0.181	0.179	0.183
	d	0.123	0.122	0.121	0.124	0.123	0.125
	θ	0.459	0.458	0.459	0.460	0.459	0.461
r	a	0.999	1.000	1.000	0.999	1.000	1.000
	d	0.999	0.999	0.999	0.999	0.999	0.999
	θ	0.988	0.988	0.988	0.988	0.988	0.988
	Mean T	0.26	0.51	0.45	0.25	12.84	12.66
	SD T	0.04	0.06	0.06	0.03	0.34	0.47
	min. T	0.20	0.32	0.37	0.20	11.61	12.12
	max. T	0.44	0.83	0.79	0.44	13.82	15.90

Note: Variable T is the elapsed time in seconds.

Parameter recovery was similar regardless of the estimation method and prior configuration. Even in the small sample conditions, the likelihood function dominated the prior and determined the value of the point estimate. As expected, the increase in sample size resulted in a more accurate estimation of the item parameters, reducing the bias and the dispersion of the estimation. Regarding the factor scores, absolute biases, RMSE and correlations remained almost equal irrespective of sample size, as the key element to increase precision was not the sample size but the test length.

The most important discrepancy between the conditions was estimation time. The MML-EM estimation proved to be faster in the two implementations than Bayesian estimation, converging in barely 0.25 seconds for small samples sizes and 0.30 seconds for large sample sizes. When priors were added to the model, time was duplicated in the MML-EM algorithm. However, using priors with the MHRM algorithm increased

estimation time up to only 12 seconds in the small sample side, and up to 22 in the large sample size.

Table 3. Parameter recovery statistics for sample size $N = 1000$

		MML-EM			MHRM		
		uniform	a-normal	a-lognormal	uniform	a-normal	a-lognormal
Bias	a	0.097	0.096	0.098	0.093	0.093	0.099
	d	0.089	0.088	0.088	0.086	0.085	0.092
	θ	0.374	0.374	0.374	0.374	0.374	0.376
RMSE	a	0.124	0.122	0.124	0.119	0.119	0.125
	d	0.107	0.107	0.106	0.106	0.105	0.106
	θ	0.460	0.460	0.460	0.461	0.461	0.463
r	a	1.000	1.000	1.000	1.000	1.000	1.000
	d	0.998	0.998	0.998	0.998	0.998	0.998
	θ	0.988	0.988	0.988	0.988	0.988	0.988
	Mean T	0.31	0.62	0.55	0.31	22.78	22.64
	SD T	0.04	0.08	0.07	0.04	0.97	1.14
	min. T	0.24	0.42	0.40	0.25	20.91	21.31
	max. T	0.43	0.84	0.74	0.44	25.50	28.24

Note: Variable T is the elapsed time in seconds.

Results show that in simple models there were no real differences between the MML-EM estimation implemented by default in the *mirt* packages and MHRM, except for a few seconds of computation time. However, MHMR was developed for conditions of high dimensional models, and these results leave open the question of what happens when the model complexity grows. Is adding more information with the priors beneficial for the precision and the convergence of the estimation in such models? In the next two studies we explored the answer to this question.

3. Simulation study two. Estimation of a model with a complex parameterization

The purpose of the second study was to evaluate recovery for all kinds of structural parameters, intercepts, slopes, factor variances, and factor covariances. Recovery of incidental parameters, factor scores, were be evaluated as well. Two consolidated estimation algorithms (MML-EM and

MCMC) were compared to an algorithm of new usage (MHRM). Bayesian estimation allows testing more flexible factor structures, enhancing the possibility of estimate parameters that usually are constrained to zero (Muthén & Aspharauhov, 2012). Given that in IRT applications dimensions tend to be highly correlated (Sinharay, 2010), in this study we explored the estimation of a model with five correlated factors and 25 manifest variables. The theoretical model is based in the factor structure purposed by Golay, Reverte, Rossier, Favez & Lecerf (2013), in which manifest variables are allowed to load on different factors.

Table 4 shows the true item parameter values. For identification purposes, the first item in each set of five items loaded on a single dimension, and the rest of the items loaded on two different factors. In this way, there was a unidimensional item for each factor, and the other items were two-dimensional. Scale parameters of the unidimensional items were fixed to 1 for the factor variances to be identifiable. True scale parameters of the two-dimensional items were randomly generated from a lognormal (0, 0.5) distribution and intercepts were generated from a normal (0, 1). Factor scores were generated at random from a multivariate normal ($\mathbf{0}$, $\mathbf{\Sigma}$) distribution, where $\mathbf{\Sigma}$ contains ones in the diagonal and 0.5 as true covariance values. These kinds of models are typically applied to measure dimensions that share some aspects of the items, like coping strategies (Zuckerman & Gagne, 2003).

The three algorithms, MML-EM, MHRM, and MCMC, were implemented using the statistical software R (R Core Team, 2015) by means of the version 1.13 of the *mirt* package for MML-EM and MHRM (Chalmers, 2012), and *rjags* for MCMC by Gibbs-sampling (Plummer, 2015). Default options were set for the *mirt* algorithms (7 quadrature points used per dimension, 500 max EM cycles, convergence occurred when all parameters were less than |.0001| across cycles; 2000 iterations for the MHRM and a burn-in period of 150 iterations). The MCMC algorithm uses four chains per parameter, a burn-in period of 5000 samples, and 10000 samples per chain are kept after the burn-in period.

The three estimation algorithms were implemented using the same prior distributions. This imposes some restrictions on the design of the simulation because the number of prior distributions implemented in *mirt* was more limited than in *rjags*. As in the first study, we fitted the models twice using different priors:

- *Uninformative priors*. The uniform distribution was the prior for the item parameters so that Bayesian point estimates were equivalent to maximum likelihood estimates; we refer to these priors as *uninformative*.

Table 4. True parameters values for the second simulation study

d_i	a_{i1}	a_{i2}	a_{i3}	a_{i4}	a_{i5}
-0.93	**1**				
-0.70		**1**			
0.41			**1**		
0.48				**1**	
0.12					**1**
0.79	1.25	1.05			
-1.95	0.83	0.36			
0.09	0.70	1.15			
-2.21	0.89	0.49			
1.91		2.34	1.30		
-0.68		0.78	1.03		
2.19		0.99	0.57		
-0.96		1.10	1.85		
1.33			0.94	0.47	
-0.28			0.57	0.71	
-0.92			1.20	2.01	
0.45			0.42	0.59	
-0.91				1.23	1.08
1.13				0.82	1.38
2.08				1.51	1.43
0.81				0.32	1.58
0.27	0.87				0.83
-2.02	1.11				0.53
-0.36	1.40				0.83
1.19	1.16				1.06

Note: Boldface indicates fixed parameters. Empty cells represent structural zeros

- *Informative priors*. We employed the lognormal and the normal distributions for the slopes and the intercepts, respectively. We chose the lognormal because there were no substantive differences in the previous study compared with the normal as slope prior, and also because it has been frequently used in the IRT literature; for example, Patz and Junker (1999) used the lognormal to implement the Bayesian MCMC.

  Therefore, the prior distributions for item parameters are:

$$
\begin{aligned}
&a \sim \text{uniform}\ (0,\ 5)\\
&d \sim \text{uniform}\ (-5,\ 5)\\
&\qquad\text{or}\\
&a \sim \text{lognormal}\ (0,\ 0.5)\\
&d \sim \text{normal}\ (0,\ 3)
\end{aligned}
\tag{6}
$$

A multivariate normal ($\mathbf{0}$, $\mathbf{\Sigma}$) was used for person parameters; and the variance-covariance matrix, $\mathbf{\Sigma}$, follows an inverse Wishart (υ) distribution, which is a common choice for Bayesian estimation because the inverse Wishart prior is conjugate to the normal distribution of $\boldsymbol{\theta}$ (Gelman, Carlin, Stern & Rubin, 1995). Thus:

$$
\begin{aligned}
&\theta_i \mid \sum \sim \text{multivariate normal}\ (0,\ \sum)\\
&\sum \sim \text{inverse wishart}\ (\upsilon\mathbf{I},\ \upsilon)
\end{aligned}
\tag{7}
$$

The Wishart parameter, υ, should be equal to or higher than the number of dimensions of the model (Gelman et al., 1995). Setting $\upsilon = 5$, the expected value of the inverse Wishart distribution is an identity matrix, and the variance-covariance matrix of $\mathbf{\Sigma}$ has 0.4 on the diagonal and 0.2 outside the diagonal (Ntzoufras, 2009).

The *mirt* package did not admit the inverse Wishart prior or any other distribution for $\mathbf{\Sigma}$, which was equivalent to assuming that the prior for $\mathbf{\Sigma}$ is a uniform one. So, this matrix was estimated with no specific prior in *mirt*. Thus, we applied MML-EM and MHRM using *mirt* with a uniform prior for $\mathbf{\Sigma}$, and MCMC using *rjags* with the inverse-Wishart prior. The purpose of fixing the slopes of the unidimensional items was to estimate all the elements of $\mathbf{\Sigma}$ matrix with the three algorithms.

Parameter recovery for the three estimation algorithms was studied in different conditions. We manipulated: the sample size (500 vs. 1000 simulees), and the prior distributions for the item parameters (ML estimation vs. Bayesian estimation). This rendered a total of twelve conditions (three algorithms × two sample sizes × two prior distributions).

As in the first study, 100 simulated samples were computed for each sample size, keeping the same vector of true θ values for all the samples. Again, the response matrices varied from one replication to another, but not across conditions. The estimation algorithms were applied to the same simulated samples of responses so that the differences between estimation methods are not contaminated by sampling error.

In order to improve the comparability of the solutions given by each algorithm, the same procedure was followed to obtain the person parameter estimates. Due to the difficulties of computing a MAP estimator for the MCMC method, the EAP estimator was employed in the simulation and the analytic algorithms. Thus, all parameters were estimated by EAP and their recovery was assessed with the absolute mean bias, the RMSE and the correlations between point estimates and the true parameter values. The results appear in Tables 5 and 6.

Table 5. Parameter recovery statistics for the uninformative priors

		N = 500			N = 1000		
		MCMC	MHRM	MML-EM	MCMC	MHRM	MML-EM
Bias	a	0.134	0.101	0.177	0.079	0.081	0.191
	d	0.059	0.065	0.079	0.045	0.046	0.053
	θ	0.340	0.376	0.354	0.339	0.342	0.380
	Σ	0.054	0.053	0.403	0.050	0.077	0.397
RMSE	a	0.328	0.380	0.792	0.249	0.245	0.499
	d	0.151	0.159	0.185	0.106	0.108	0.131
	θ	0.543	0.649	0.764	0.536	0.626	0.898
	Σ	0.135	0.168	0.621	0.122	0.149	0.599
r	a	0.975	0.956	0.910	0.993	0.966	0.915
	d	0.997	0.997	0.996	0.999	0.999	0.999
	θ	0.918	0.886	0.900	0.919	0.904	0.882
	Σ	0.972	0.960	0.903	0.978	0.974	0.881
	Mean T	0:43:33	0:00:33	0:15:06	2:12:05	0:00:52	0:12:49
	SD T	0:04:01	0:00:01	0:05:27	0:01:12	0:00:05	0:06:56
	min. T	17:11:30	0:00:30	0:03:20	2:07:54	0:00:49	0:02:42
	max. T	1:16:53	0:00:38	0:31:20	2:18:25	0:01:29	0:33:42

Note: Variable T is the elapsed time with format hh:mm:ss

The main result from tables 5 and 6 was the impact of the prior on the estimation precision. In particular, using informative priors for the item parameters improved the accuracy of the estimation of the slopes and the person parameters among the three methods, especially when the sample size was small. The differences between informative and uninformative priors in absolute bias and RMSE were lower in the large sample condition,

but recovery was still better for the informative prior. Something similar happened with the recovery of the intercepts; there were almost no differences between priors with the large sample size. However, when the sample size was small the RMSE remained lower in the informative prior conditions, although the absolute bias was similar for both types of priors.

Table 6. Parameter recovery statistics for the informative priors

		N = 500			N = 1000		
		MCMC	MHRM	MML-EM	MCMC	MHRM	MML-EM
Bias	a	0.090	0.081	0.135	0.066	0.098	0.204
	d	0.054	0.062	0.052	0.046	0.061	0.046
	θ	0.327	0.375	0.474	0.327	0.344	0.476
	Σ	0.085	0.091	0.203	0.058	0.081	0.233
RMSE	a	0.233	0.274	0.298	0.194	0.207	0.304
	d	0.130	0.142	0.135	0.099	0.112	0.101
	θ	0.550	0.644	0.661	0.541	0.623	0.654
	Σ	0.157	0.179	0.210	0.122	0.139	0.236
r	a	0.983	0.986	0.989	0.992	0.992	0.996
	d	0.998	0.997	0.997	0.999	0.999	0.999
	θ	0.916	0.885	0.815	0.919	0.903	0.816
	Σ	0.978	0.944	0.985	0.978	0.981	0.990
	Mean T	0:26:26	0:00:32	0:01:39	1:57:52	0:00:58	0:01:36
	SD T	0:05:24	0:00:03	0:00:51	0:12:48	0:00:04	0:00:29
	min. T	0:24:39	0:00:28	0:00:41	0:11:52	0:00:45	0:00:59
	max. T	1:18:07	0:00:51	0:05:04	2:29:02	0:01:09	0:03:34

Note: Variable *T* is the elapsed time with format hh:mm:ss

Contrary to item parameters, the variance-covariance matrix was estimated with better precision (i.e. lower absolute bias and RMSE) with uninformative priors when the sample size was small for the MCMC and MHRM algorithms. These discrepancies disappeared when the sample size was increased. This did not occur with the MML-EM algorithm, whose estimation was far worse with uninformative priors. The correlations between true and estimated parameters followed a similar pattern; correlations tended to be higher when the lognormal and the normal

distributions are utilized as priors for the item parameters. Thus, the estimates were in general better with informative priors; but this is achieved at the cost of forcing a-parameters to be positive.

In general, the MCMC and MHRM algorithms rendered similar results, whereas MML-EM provided less precise estimates. The MCMC and MHRM solutions showed smaller bias, lower RMSE and higher correlations between the true and estimated parameters than the MML-EM algorithm for both types of priors. One exception was the recovery of the intercept parameters, where the MML-EM performed as well as the other two algorithms.

The greater discrepancy among the algorithms occurred in the recovery of the person parameters. The bias and the RMSE of the analytic algorithms seemed remarkably high in comparison with MCMC estimates. Interestingly, these differences appear with both sample sizes, and resulted even more pronounced in MML-EM conditions.

Looking closer at the MCMC and the MHRM results, the MCMC solution offered a slightly less biased estimation than the one reached with the MHRM algorithm. Although the point estimate of the MHRM solution for the slopes was less biased than the point estimate of the MCMC, the dispersion of the MCMC estimates was lower, yielding lower RMSE values for these parameters in the small sample size conditions. Nonetheless, these differences softened up in the larger sample size, obtaining similar absolute biases, RMSE and correlations for the item parameters and the matrix $\boldsymbol{\Sigma}$.

Another important concern for practical applications is estimation time, which compromises the number and the structure of the models than can be applied to a real data sample for the purposes of judging their relative merit. As expected, estimation time had a direct relation with the computational load of the algorithm (see Tables 5 and 6). The MHRM was the fastest algorithm, converging on average in 32-58 seconds when using informative priors, and converging in 33-52 seconds with uninformative priors. The MML-EM was very much affected by the type of prior, with the informative prior it converges in about one and a half minute, whereas estimation lasted about fifteen minutes on average with uninformative priors. Finally, MCMC took between 26 minutes and around two hours, which was not surprising given that this method is computationally more intensive. In general, more time was needed as the sample size increased, although the effect was more prominent as the computational burden of the algorithm increases.

To sum up, using informative priors was beneficial for the three estimation algorithms. Besides this, the MHRM method –an improvement

of MML-EM designed to overcome those situations where computational complexity of the Gauss-Hermite numerical integration procedure involved in MML-EM is cumbersome– achieved its objective of reducing computational time while providing precise estimates.

These results were not conclusive because a model with five factors can still be handled by a numerical integration procedure, but it was unclear if it can be applied to models with a higher number of dimensions that are frequently found in practical applications. The supposed benefits of the new methods have not yet been extensively tested in high dimensional models. A third simulation study was carried out to obtain more information regarding high dimensional models where numerical quadrature integration methods could be unfeasible.

4. Simulation study three. Estimation of a highly dimensional model

The aim of the third simulation study was to evaluate the performance of four Bayesian estimation algorithms with a complex model of high dimensionality. The parameterization of the model was simpler than in the second simulation study, in the sense that no factor variances or covariances were estimated, but the number of factors was higher. The simulation was based on the hierarchical factor model described by Yung, Thissen, and McLeod (1999) to use a realistic factorial structure. We have chosen this model to increase the ecological validity of the simulation study because it was based on the bi-factor model, which has a long tradition in psychometrics, and has proven to be a reliable alternative to the classical second-order models in health and behavioral sciences (Chen, West & Sousa, 2006; Patrick, Hicks, Nichol, & Krueger, 2007; Reise, 2012).

In this study, the model included 18 variables and 10 factors. Each variable measured three factors: one general factor that was shared by all variables and two group factors. However, we have not used the same values for the slopes as Yung et al. True parameter values for the present study appear in Table 7; as can be seen, intercept parameters were fixed to 0, and scale parameters were set to 1. The reason was that we wanted to evaluate the impact of the factor structure on the recovery of parameters. As all true parameters were fixed to the same value, differences in recovery between parameters will not be attributed to their true value but to factor structure and sampling error.

Four algorithms were compared: MML-EM, MHRM, MCMC, and HMC (Neal, 2011). By MCMC, we refer to the traditional MCMC based on Gibb-sampling, although HMC was also an MCMC algorithm that

substitutes Hamiltonian dynamics for Gibbs-sampling. HMC was implemented using the R package *rstan* (Stan Development Team, 2014a, 2014b). The HMC algorithm was applied in this simulation example because it is a recent improvement of MCMC aimed at increasing computational speed and efficiency, which are demanding requisites for a model of high complexity. The typically long estimation time and waste of computational resources associated with Gibbs-sampling-based-MCMC might render this estimation method unfeasible as model complexity grows. We have used the same chain length and burn-in period values for the MHRM and MCMC as in the previous studies.

Table 7. True parameter values for the second simulation study

d	a_{i1}	a_{i2}	a_{i3}	a_{i4}	a_{i5}	a_{i6}	a_{i7}	a_{i8}	a_{i9}	a_{i10}
0	1	1			1					
0	1	1			1					
0	1	1			1					
0	1	1				1				
0	1	1				1				
0	1	1				1				
0	1		1				1			
0	1		1				1			
0	1		1				1			
0	1		1					1		
0	1		1					1		
0	1		1					1		
0	1			1					1	
0	1			1					1	
0	1			1					1	
0	1			1						1
0	1			1						1
0	1			1						1

Note: Empty cells represent structural zeros

The MML-EM algorithm was implemented using the *bfactor* function in the *mirt* package. This function is applicable when the factorial structure contains both general factors shared by all items and specific factors shared by group of items, as in this example. The advantage of the *bfactor* function was that it introduced simplifications in the numerical integration procedure involved in MML-EM and demanded fewer computational resources than the traditional Gauss-Hermite integration procedure. In particular, the *bfactor* function reduces the integration over the 10 factors of the model to integration over five dimensions (four dimensions in the second-tier plus

one specific factor). Minor changes were performed to the *mirt* default options in this third study (7 quadrature points used per dimension, 500 max EM cycles, convergence occurred when all parameters were less than |.0001| across cycles). The number of quadrature points were increased to 10 because in pilot runs we observed that estimation was imprecise using the lower number of points that *bfactor* sets by default.

The four algorithms were applied in two conditions, with high informative and with low informative prior distributions. The high informative prior distributions were:

$$\begin{aligned} a &\sim \text{lognormal}\ (0,\ 0.5) \\ d &\sim \text{normal}\ (0,\ 3) \end{aligned} \tag{8}$$

As before, the lognormal distribution for a had a median of 1, an expectation of 1.13, and a standard deviation of 0.6. The normal prior for the d parameter was a relatively flat uninformative distribution. The low informative prior distributions were:

$$\begin{aligned} a &\sim \text{uniform}\ (0,\ 5) \\ d &\sim \text{uniform}\ (-5,\ 5) \end{aligned} \tag{9}$$

Note that the prior distributions in (8) and (9) were not high and low informative in a general sense, but in comparison with one another. The prior for θ was standard normal in all conditions.

The four algorithms were examined with sample sizes of 500 and 1000 simulees, resulting in 16 conditions (four algorithms × two samples sizes × two set of prior distributions). Once again, 100 simulated samples were computed for each sample size, keeping the item and person parameters constant across replications, and varying only the response matrices. The four estimation algorithms were applied to each of the simulated samples.

Recovery of parameters was analyzed by computing the root mean squared error (RMSE) between Bayesian point estimates and the true parameter values. Correlations between true and estimated parameters were computed for θ (the correlation for a and d would be 0, as all true parameters take the same value).

Tables 8 and 9 contain the correlations and RMSE of the factor scores. The recovery of person parameters was a little bit worse with MML than with the other three methods, mainly with the uniform prior. The most prominent effect was that recovery of factor scores was better as the number of items per factor increased. In this manner, the correlations between the true and the estimated factor scores for the general factor, measured by 18

items, were higher than for the other factor, measured by six or three items. The RMSE values followed this same tendency, presenting lower values for the general factor than for the other ones.

Table 8. Correlations between true and estimated theta

Lognormal	1	.707	.705	.710	.710	.721	.714	.721	.714
Normal	2	.535	.543	.515	.534	.542	.551	.542	.551
	3	.551	.525	.532	.517	.554	.532	.554	.532
	4	.542	.510	.498	.496	.550	.519	.551	.519
	5	.461	.479	.445	.469	.460	.478	.461	.478
	6	.473	.489	.428	.478	.473	.490	.473	.490
	7	.501	.468	.468	.463	.501	.469	.501	.469
	8	.509	.492	.487	.475	.510	.493	.510	.493
	9	.466	.466	.456	.460	.467	.465	.467	.465
	10	.454	.477	.401	.468	.465	.481	.465	.481
Uniform	1	.699	.710	.710	.710	.732	.720	.709	.712
	2	.495	.543	.514	.535	.553	.559	.529	.545
	3	.531	.523	.532	.517	.565	.541	.548	.525
	4	.478	.493	.499	.496	.562	.530	.532	.505
	5	.439	.471	.447	.469	.467	.483	.451	.468
	6	.437	.481	.431	.478	.479	.497	.463	.481
	7	.467	.458	.470	.463	.508	.476	.491	.456
	8	.477	.483	.487	.475	.518	.499	.500	.483
	9	.434	.458	.456	.460	.472	.471	.457	.457
	10	.429	.473	.411	.468	.471	.487	.456	.472

Note: The column Prior refers to the prior distribution for scale and intercept parameters, which can be either ($a \sim$ lognormal(0, 0.5), $d \sim$ normal(0, 3)) or ($a \sim$ uniform(-5, 5), $d \sim$ uniform(-5, 5)).

Table 10 summarizes recovery of intercept parameters and estimation time. The estimation algorithm and the prior distribution had little effect on the accuracy of the recovery of the intercepts. The precision of the estimates of the intercept parameters increased when the sample size was incremented, although recovery seemed adequate even for the $N = 500$ condition.

With respect to estimation time, simulation algorithms were clearly slower than the analytic ones. MCMC was by far the slowest algorithm, followed by HMC. These are expected results since MCMC is computationally intensive. The results for MML were quite interesting because it had remarkable differences between the runs with high and low informative prior distributions. This was due to difficulties of convergence for MML in the conditions with uniform priors. MHRM had the best results from all fronts; it was the fastest algorithm and was not affected by sample size and the type of prior distributions.

Table 9. Root mean square error of theta

		mirt (MML-EM)		mirt (MHRM)		Jags (MCMC)		Stan (HMC)	
Prior	Dim.	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$
Lognormal	1	.623	.639	.622	.634	.615	.638	.615	.638
Normal	2	.715	.723	.733	.732	.721	.728	.720	.728
	3	.720	.699	.736	.707	.728	.706	.728	.706
	4	.727	.738	.757	.747	.729	.739	.729	.739
	5	.760	.759	.778	.762	.769	.763	.768	.763
	6	.759	.752	.776	.761	.765	.757	.765	.757
	7	.764	.766	.783	.774	.771	.771	.771	.771
	8	.792	.791	.807	.797	.798	.795	.798	.795
	9	.760	.764	.779	.775	.769	.773	.769	.773
	10	.779	.765	.790	.760	.773	.763	.773	.763
Uniform	1	.678	.635	.622	.634	.606	.634	.629	.641
	2	.793	.725	.733	.732	.711	.720	.733	.736
	3	.787	.702	.736	.707	.716	.698	.738	.715
	4	.824	.746	.756	.747	.719	.732	.743	.750
	5	.831	.762	.778	.762	.751	.752	.776	.769
	6	.834	.758	.775	.761	.749	.747	.773	.763
	7	.841	.773	.782	.774	.757	.758	.778	.780
	8	.871	.796	.808	.797	.784	.785	.805	.800
	9	.838	.773	.779	.775	.754	.761	.777	.778
	10	.843	.763	.786	.760	.758	.750	.781	.770

Note: The column Prior refers to the prior distribution for scale and intercept parameters, which can be either ($a \sim$ lognormal(0, 0.5), $d \sim$ normal(0, 3)) or ($a \sim$ uniform(-5, 5), $d \sim$ uniform(-5, 5)).

Results for the scale parameters appear in tables 11 and 12. The sample size and the number of items per dimension affected the recovery of the slopes. In general, the RMSE presented lower values as the sample size and the number of items per dimension increased. The effect of the type of prior distribution depended on the estimation method.

The most relevant effect on slopes was the poor performance of MML in the condition with uniform prior and 500 simulees. As indicated before, we have observed in pilot runs that performance of MML was poor using the default number of quadrature points of the *bfactor* function (7 points per dimension for integration over five dimensions). Because of this, we increased this number to 10 quadrature points per dimension. However, recovery for MML still did not compare with the other methods, and convergence was slow. In the conditions with 1000 simulees, parameter recovery for the MML was not as good as for the other methods, but the difference was smaller than with 500 simulees.

Table 10. Recovery of intercept parameters (RMSE) and elapsed time

		mirt (MML-EM)		mirt (MHRM)		Jags (MCMC)		Stan (HMC)	
Prior	Parameter	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$
Normal	d_1	.106	.076	.113	.078	.112	.077	.112	.077
	d_2	.107	.076	.115	.077	.117	.076	.117	.076
	d_3	.109	.072	.110	.079	.115	.075	.116	.076
	d_4	.144	.078	.139	.077	.142	.079	.142	.079
	d_5	.137	.083	.135	.081	.136	.081	.136	.082
	d_6	.146	.092	.139	.096	.144	.092	.144	.092
	d_7	.103	.115	.105	.109	.103	.116	.102	.117
	d_8	.116	.119	.117	.111	.121	.117	.123	.178
	d_9	.119	.109	.130	.102	.121	.109	.121	.109
	d_{10}	.121	.111	.137	.111	.136	.109	.136	.109
	d_{11}	.122	.123	.131	.126	.127	.126	.127	.126
	d_{12}	.101	.114	.109	.117	.106	.112	.106	.111
	d_{13}	.106	.078	.120	.081	.117	.081	.116	.081
	d_{14}	.118	.075	.117	.079	.115	.079	.116	.079
	d_{15}	.112	.079	.125	.083	.123	.078	.123	.078
	d_{16}	.131	.076	.129	.071	.122	.076	.122	.076
	d_{17}	.129	.085	.134	.085	.123	.090	.123	.090
	d_{18}	.132	.077	.130	.074	.126	.081	.125	.081
	Mean T	0:2:41	0:4:16	0:3:20	0:3:33	0:21:51	1:40:45	0:8:24	0:21:27
	Std. T	0:0:38	0:1:14	0:0:8	0:0:6	0:0:28	0:1:49	0:0:25	0:2:51
	min. T	0:1:44	0:2:50	0:3:1	0:3:19	0:21:29	1:37:15	0:8:1	0:18:30
	max. T	0:4:51	0:9:43	0:3:39	0:3:53	0:23:23	1:44:20	0:11:14	0:26:35
Uniform	d_1	.118	.076	.113	.078	.099	.071	.121	.082
	d_2	.117	.075	.114	.077	.103	.070	.128	.082
	d_3	.103	.075	.112	.078	.102	.070	.124	.080
	d_4	.169	.077	.139	.077	.124	.072	.157	.083
	d_5	.152	.078	.133	.082	.121	.075	.148	.087
	d_6	.182	.081	.139	.096	.126	.083	.158	.099
	d_7	.107	.092	.105	.109	.089	.106	.113	.125
	d_8	.185	.125	.119	.111	.104	.108	.133	.126
	d_9	.102	.116	.128	.102	.105	.102	.133	.115
	d_{10}	.170	.107	.135	.111	.116	.100	.147	.119
	d_{11}	.156	.109	.130	.126	.110	.114	.137	.136
	d_{12}	.115	.125	.109	.117	.092	.101	.113	.120
	d_{13}	.182	.111	.120	.081	.101	.076	.129	.087
	d_{14}	.122	.079	.117	.079	.103	.074	.127	.086
	d_{15}	.160	.079	.125	.083	.110	.071	.135	.084
	d_{16}	.160	.077	.128	.071	.109	.069	.132	.081
	d_{17}	.170	.090	.132	.085	.109	.082	.136	.096
	d_{18}	.133	.081	.130	.074	.113	.074	.137	.087
	Mean T	0:37:36	0:23:15	0:3:19	0:3:28	0:25:58	1:32:43	0:13:39	0:33:1
	Std. T	0:29:59	0:19:7	0:0:9	0:0:6	0:0:9	0:1:16	0:0:31	0:2:13
	min. T	0:6:59	0:9:42	0:2:59	0:3:15	0:25:42	1:29:22	0:12:53	0:30:22
	max. T	2:7:48	1:49:25	0:3:52	0:3:43	0:26:30	1:37:36	0:16:11	0:44:55

Note: Variable T is the elapsed time with format hh:mm:ss

Table 11. Recovery of scale parameters (RMSE) with lognormal prior

	mirt (MML-EM)		mirt (MHRM)		Jags (MCMC)		Stan (HMC)	
Par.	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$
$a_{1,1}$	.199	.166	.216	.149	.175	.143	.175	.144
$a_{2,1}$	.205	.152	.226	.147	.181	.135	.180	.135
$a_{3,1}$	.212	.149	.219	.154	.187	.128	.187	.127
$a_{4,1}$	.187	.152	.185	.158	.180	.152	.181	.152
$a_{5,1}$	.208	.178	.217	.162	.191	.163	.192	.162
$a_{6,1}$	.189	.148	.188	.160	.195	.151	.194	.150
$a_{7,1}$	.201	.135	.191	.152	.190	.136	.191	.137
$a_{8,1}$	.212	.158	.176	.162	.206	.140	.207	.141
$a_{9,1}$	.222	.180	.218	.151	.201	.155	.204	.154
$a_{10,1}$	.242	.160	.226	.148	.185	.140	.185	.140
$a_{11,1}$	.254	.150	.253	.146	.206	.140	.205	.142
$a_{12,1}$	.245	.155	.239	.159	.193	.142	.193	.142
$a_{13,1}$	.215	.166	.215	.159	.195	.144	.193	.143
$a_{14,1}$	.211	.138	.195	.153	.202	.145	.202	.144
$a_{15,1}$	.206	.146	.209	.162	.182	.138	.181	.139
$a_{16,1}$	.194	.169	.201	.145	.170	.134	.171	.136
$a_{17,1}$	.190	.175	.201	.173	.170	.164	.171	.165
$a_{18,1}$	.167	.148	.184	.143	.169	.134	.169	.134
$a_{1,2}$	.234	.211	.328	.233	.210	.172	.211	.169
$a_{2,2}$	.245	.192	.320	.248	.222	.162	.229	.164
$a_{3,2}$	.224	.227	.341	.248	.198	.194	.196	.192
$a_{4,2}$	.200	.263	.327	.237	.167	.173	.168	.172
$a_{5,2}$	.224	.216	.341	.237	.198	.142	.197	.142
$a_{6,2}$	.218	.288	.368	.243	.199	.194	.194	.189
$a_{5,3}$	.274	.214	.414	.228	.209	.162	.206	.162
$a_{6,3}$	.270	.204	.374	.234	.232	.177	.231	.174
$a_{7,3}$	.310	.237	.415	.265	.267	.198	.262	.196
$a_{8,3}$	.273	.248	.312	.273	.228	.179	.228	.179
$a_{9,3}$	.348	.250	.450	.252	.297	.149	.300	.155
$a_{10,3}$	.309	.237	.482	.290	.293	.186	.297	.182
$a_{11,4}$	.198	.237	.366	.313	.180	.196	.181	.202
$a_{12,4}$	.205	.232	.376	.314	.206	.204	.206	.203
$a_{13,4}$	.250	.233	.362	.302	.237	.186	.237	.185
$a_{14,4}$	.230	.191	.398	.249	.196	.173	.193	.170
$a_{15,4}$	.213	.212	.423	.254	.199	.186	.195	.182
$a_{16,4}$	.190	.200	.395	.260	.191	.188	.189	.184

Table 11 (continued). Recovery of scale parameters (RMSE) with lognormal prior

	mirt (MML-EM)		mirt (MHRM)		Jags (MCMC)		Stan (HMC)	
Par.	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	Par.	$n = 500$	$n = 1000$	$n = 500$
$a_{1,5}$	.218	.256	.456	.299	.242	.203	.243	.207
$a_{2,5}$	.226	.247	.469	.270	.264	.205	.255	.210
$a_{3,5}$	.213	.265	.418	.314	.244	.217	.241	.220
$a_{4,6}$	.222	.245	.443	.360	.251	.210	.248	.217
$a_{5,6}$	.209	.254	.491	.320	.234	.215	.230	.216
$a_{6,6}$	.213	.241	.452	.303	.239	.193	.237	.196
$a_{7,7}$	.239	.242	.470	.297	.261	.254	.269	.254
$a_{8,7}$	.232	.214	.482	.302	.265	.193	.266	.192
$a_{9,7}$	.227	.212	.483	.284	.281	.196	.280	.191
$a_{10,8}$	.250	.257	.504	.337	.277	.238	.278	.232
$a_{11,8}$	.298	.248	.525	.267	.314	.217	.317	.222
$a_{12,8}$	.254	.232	.470	.376	.266	.206	.267	.202
$a_{13,9}$	.217	.197	.496	.285	.262	.192	.255	.197
$a_{14,9}$	.205	.208	.372	.285	.227	.214	.229	.213
$a_{15,9}$	.188	.225	.500	.282	.215	.222	.213	.223
$a_{16,10}$	.783	.725	.489	.281	.200	.214	.194	.211
$a_{17,10}$	.222	.294	.492	.300	.221	.229	.222	.230
$a_{18,10}$	.199	.281	.520	.305	.229	.231	.224	.228
Mean	.237	.218	.353	.239	.218	.179	.218	.179

To sum up, MHRM, MCMC and HMC recovered factor scores and item parameters with similar precision, whereas MML was clearly inferior. MCMC and HMC were slower than MML and MHRM but this is an expected result from the definition of these algorithms. Simulation algorithms proved to be as precise as the analytic ones. Estimation by simulation seemed to be unnecessary if the purpose is just to obtain a point estimate (possibly supplemented with standard error) because of the longer estimation time. However, simulations have the advantage of providing a whole sample of parameter estimates and not just a point estimate, and these samples can be used as other quantities such as diagnostic statistics for posterior predictive assessment of model fit.

Table 12. Recovery of scale parameters (RMSE) with uniform prior

	mirt (MML-EM)		mirt (MHRM)		Jags (MCMC)		Stan (HMC)	
Par.	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$
$a_{1,1}$	.434	.165	.214	.149	.264	.180	.240	.183
$a_{2,1}$	.374	.170	.223	.147	.264	.162	.250	.199
$a_{3,1}$	.323	.156	.219	.155	.260	.170	.262	.176
$a_{4,1}$	.477	.183	.182	.157	.189	.127	.326	.240
$a_{5,1}$	.509	.181	.211	.162	.189	.129	.329	.253
$a_{6,1}$	.547	.170	.189	.160	.180	.119	.335	.246
$a_{7,1}$	.439	.287	.193	.152	.186	.153	.294	.209
$a_{8,1}$	.492	.159	.173	.162	.186	.154	.315	.208
$a_{9,1}$	.331	.175	.217	.151	.201	.155	.312	.232
$a_{10,1}$	.314	.168	.226	.148	.273	.141	.222	.231
$a_{11,1}$	.521	.192	.252	.146	.291	.141	.259	.227
$a_{12,1}$	.265	.163	.235	.159	.279	.146	.237	.223
$a_{13,1}$	.607	.170	.217	.159	.212	.150	.342	.232
$a_{14,1}$	.638	.206	.197	.153	.197	.133	.362	.272
$a_{15,1}$	.540	.285	.206	.161	.211	.139	.320	.255
$a_{16,1}$	.380	.312	.197	.145	.215	.138	.291	.229
$a_{17,1}$	.561	.214	.195	.173	.192	.154	.325	.263
$a_{18,1}$	.622	.322	.186	.142	.203	.142	.325	.254
$a_{1,2}$	.422	.195	.334	.233	.222	.167	.341	.268
$a_{2,2}$	.646	.207	.339	.248	.188	.167	.370	.245
$a_{3,2}$	.634	.222	.342	.247	.199	.155	.320	.292
$a_{4,2}$	.479	.214	.327	.237	.197	.147	.303	.261
$a_{5,2}$	.355	.176	.331	.238	.224	.154	.315	.225
$a_{6,2}$	.650	.223	.353	.243	.198	.138	.347	.313
$a_{5,3}$	.583	.251	.418	.228	.175	.173	.379	.233
$a_{6,3}$	.558	.205	.370	.234	.184	.173	.389	.262
$a_{7,3}$	.342	.225	.405	.265	.172	.178	.433	.261
$a_{8,3}$	.426	.205	.323	.273	.171	.155	.374	.280
$a_{9,3}$	.579	.194	.462	.251	.165	.143	.468	.284
$a_{10,3}$	.346	.220	.466	.290	.172	.151	.436	.308
$a_{11,4}$	.482	.296	.360	.313	.234	.209	.286	.289
$a_{12,4}$	.480	.332	.379	.313	.226	.215	.343	.287
$a_{13,4}$	.516	.281	.362	.302	.223	.205	.376	.267
$a_{14,4}$	.644	.238	.382	.249	.219	.185	.315	.259
$a_{15,4}$	.811	.268	.426	.254	.230	.184	.321	.277
$a_{16,4}$	.675	.281	.388	.259	.218	.190	.306	.277

Table 12 (continued). Recovery of scale parameters (RMSE) with uniform prior

	mirt (MML-EM)		mirt (MHRM)		Jags (MCMC)		Stan (HMC)	
Par.	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$
$a_{1,5}$	.857	.283	.466	.299	.232	.189	.430	.342
$a_{2,5}$	.781	.340	.461	.270	.223	.183	.427	.387
$a_{3,5}$	.609	.314	.418	.314	.250	.176	.402	.370
$a_{4,6}$	.865	.338	.461	.359	.231	.161	.445	.354
$a_{5,6}$	.975	.291	.475	.321	.241	.160	.414	.378
$a_{6,6}$	1.154	.271	.439	.300	.249	.168	.415	.336
$a_{7,7}$	.901	.668	.466	.298	.210	.193	.445	.444
$a_{8,7}$	.908	.278	.466	.302	.221	.191	.442	.365
$a_{9,7}$	.714	.260	.483	.284	.215	.213	.466	.348
$a_{10,8}$	.796	.326	.497	.337	.197	.174	.452	.397
$a_{11,8}$	1.361	.345	.521	.268	.196	.161	.486	.351
$a_{12,8}$	.394	.277	.465	.376	.198	.163	.434	.355
$a_{13,9}$	1.189	.263	.496	.286	.264	.194	.437	.306
$a_{14,9}$	.897	.352	.369	.285	.268	.177	.410	.366
$a_{15,9}$	1.080	.535	.503	.282	.269	.176	.386	.357
$a_{16,10}$	.646	.610	.468	.282	.258	.167	.367	.330
$a_{17,10}$	.998	.301	.485	.302	.255	.168	.412	.376
$a_{18,10}$	.960	.497	.492	.306	.253	.166	.414	.404
Mean	.631	.268	.351	.239	.219	.165	.360	.289

5. The problem of factor orientation

The problem of factor orientation in multidimensional item response theory and factor analysis is that fit remains unchanged when both the scale parameters and the factor scores are multiplied by -1. This property has important implications for Bayesian MCMC estimation because several chains of parameters run in parallel, and the different chains may be oriented in different directions. Even more so, one single chain may change its orientation along the MCMC simulations. If the chains have different orientations, MCMC will not converge to the posterior distribution irrespective of the chain's length.

We have run 1000 iterations of the MCMC algorithm using the Stan software with four chains in the model for the third simulation study. The prior distribution for the scale parameters was uniform (-5, 5). Figure 1 shows the trace plot and the density plot of a_{21}. Two of the chains converged to a positive estimate for a_{21}, whereas the factor was oriented in the opposite direction for the other two chains, and the estimate of a_{21} was negative. In fact, the problem was more serious than the figure suggests. If some chains are oriented in one direction and the others in the opposite, this could be detected a-posteriori and all chains rescaled in the same direction.

However, it is harder to detect the problem and recode the results when factor orientation changes within the same chain, so that there are parts of the chains in which the factor is oriented one way and other parts in which orientation is reversed.

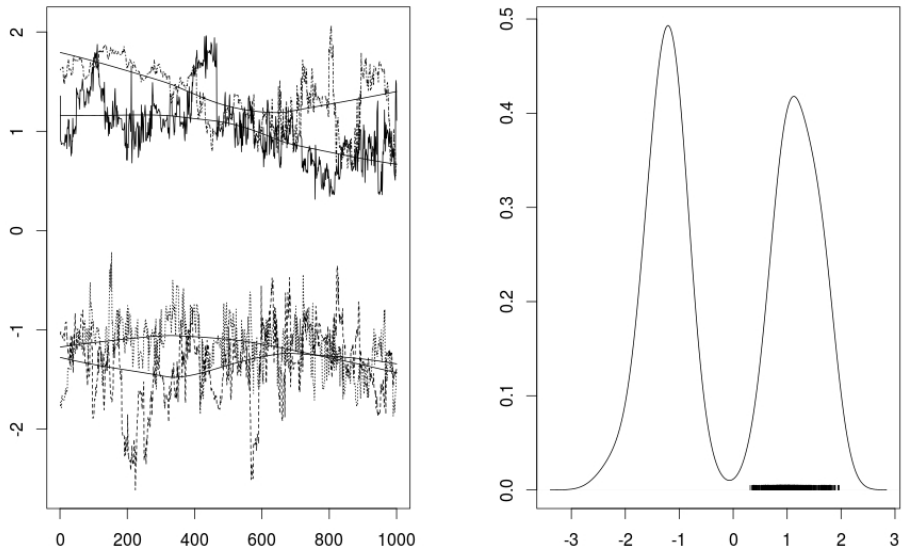

Figure 1. Trace plot and kernel density estimate of the realized values of the scale parameter a_{21}. The true parameter value is 1. The sign of the parameter is reversed across chains depending on the factor orientation.

As these problems will not be corrected by increasing the length of the chains, they have to be resolved a-priori by imposing appropriate constraints on parameters before running the chains. Apparently, a simple way to ensure that all chains have the same orientation is to set one slope to 1 (say $a_{11} = 1$) so that the fixed scale parameter sets the orientation of the factor; then, the factor variance should be set free for the total number of estimated parameters to remain unaltered. In theory, this should be enough to fix factor orientation, but we have found in practice that this constraint is too mild, and extensive computation time and chain lengths are necessary to avoid factor orientation problems with this method.

The solution we propose is to use a prior distribution that imposes that scale parameters take positive values; for example, the log-normal one for our example, or a less informative one, such as a uniform distribution in the positive real line. It can be argued that this is not a realistic assumption for practical applications in which the sign of scale parameters cannot be determined beforehand. In that case, MCMC can be run in two stages. The first step would be to run a pilot estimation with a single chain and, from the output, determine a set of scale parameters that clearly converge toward positive or negative values. The second step would be to run the definitive MCMC simulation with several chains using ad-hoc prior distributions that match the results of the pilot run. The scale parameters that converged to values away from zero in the first simulation would have prior distributions defined in the positive or negative real line for the second run. The scale parameters that converged to values close to zero in the first run would have prior distributions defined in the (complete) real line for the second run.

6. Conclusions

The purpose of this paper was to gather information about the performance of new estimation methods (MHRM and HMC) in comparison with the most habitual algorithms (MML-EM and MCMC) under latent structures of different complexity. The results showed that the four estimation methods perform similarly in recovering the parameters of models up to five factors, whereas MML-EM had problems recovering models with more dimensions.

As expected, less biased and more accurate estimates were found as the sample size and the number of items that measure each dimension increase. Recovery of intercept parameters was precise even in the conditions with 500 simulees. However, estimation of scale parameters is more demanding and is also influenced by the number of items per factors, since scale parameters on a poorly defined factor can hardly be estimated with precision. The same pattern of results was found for the estimation of factor scores.

The four estimation methods can be classified in two groups, simulation (MCMC and HMC) and analytic (MMLE-EM and MHRM). Simulation methods provide samples from the posterior distribution of item parameters, and analytic methods provide a point estimate. Moreover, HMC and MHRM can be seen as recent improvements over the more traditional methods, MCMC and MMLE-EM, on which they are based.

Besides the precision of the estimation, estimation time was a crucial criterion when choosing one estimation algorithm in practical applications, as a large number of different models are typically estimated and compared. In this respect, MHRM was by far the fastest estimation method, and within the simulation methods, HMC was clearly better than MCMC. Regarding the simulation methods, both MCMC and HMC algorithms yielded almost identical solutions, working as well as the analytic methods in models of different complexity and providing more accurate estimates of the item parameters in complex models. However, HMC converged faster than MCMC, and even faster than the EM algorithm in the small sample conditions. HMC substantively reduced estimation time, taking from 1-2 hours to 20-50 minutes, depending on the sample size. Thus, the newer methods, MHRM and HCM, constitute clear improvements over the traditional ones. These results are congruent with those obtained by Han and Paek (2014), who did not find significant differences between MML-EM, MHRM, and MCMC –among other methods and software– in conditions of low and medium model complexity.

This paper has also researched the effect of the prior distributions on recovery. There were negligible differences in recovery between low and high informative prior distribution when using HMC, small differences when using MCMC and HMC and large differences when using MML-EM in combination with small samples. It should be taken into account that many current studies with real data use smaller samples sizes than the conditions with 500 simulees in this study. Hence, it is expected that the differences between conditions will be more prominent in real applications. Thus flat priors can be used in real applications to represent high uncertainty about parameter values as long as one of the other estimation methods is used instead of MML-EM.

One important problem regarding Bayesian simulation methods is the factor orientation problem, which may impede convergence of the MCMC chains and render biased parameter estimates. This problem was addressed in this paper by using prior distributions defined only in the positive real line. Prior distributions that allow positive and negative values for scale parameters –such as the normal distribution– can entail convergence problems between different chains, resulting in a bimodal posterior distribution, with the two peaks representing the positive and negative orientations of the factor. The mean of the bimodal posterior, which is the simulated EAP estimate, will be close to zero, failing to recover the true value of the scale parameter. The simulations show that this problem can be solved by using a prior distribution defined only in the positive real line. However, this solution assumes that the sign of the scale parameter is

known beforehand, which is an unrealistic condition for practical applications. The sign of the scale parameters can be determined in real data analysis by conducting a pilot run for the estimation algorithm, although this is clearly a topic for further research.

Recent methods supersede the traditional ones regarding estimation time and accuracy. Results showed that simulation methods posed no real advantage for obtaining a point estimate. However, it is important to take into account that the purpose of the MCMC and HMC techniques was to obtain an approximation to the posterior distributions of the parameters, not only to obtain a point-wise estimator like the analytic methods. The utility of Bayesian simulation methods could be achieved in connection with more complex models (hierarchical factor structures, random item parameters, etc.), to compute posterior variances and probability intervals, or for the broader purpose of simulating the distribution of goodness-of-fit statistics, which are not always computable without simulations. On balance, MHRM seems the best alternative for the purposes of point estimation and to overcome the *curse of dimensionality* implicit in multidimensional item response and categorical factorial models.

RESUMEN

Estimación Bayesiana de modelos multidimensionales de respuesta al ítem. Una comparación de algoritmos analíticos y de simulación. El presente estudio compara el rendimiento de dos algoritmos de estimación de reciente implementación, Metropolis-Hastings Robins-Monro (MHRM) y Hamiltonian MCMC (HMC), con dos algoritmos consolidados en la literatura psicométrica, máxima verosimilitud marginal a través del algoritmo EM (MML-EM) y las cadenas de Markov de Monte Carlo (MCMC), en la estimación de modelos multidimensionales de respuesta al ítem de diferente complejidad. Para evaluar la recuperación de parámetros se plantearon tres estudios de simulación desde un acercamiento Bayesiano. El primer estudio utiliza un modelo unidimensional sencillo para evaluar el efecto de distribuciones previas informativas y no informativas. El segundo estudio compara el algoritmo MHRM con MML-EM y MCMC en la estimación de un modelo de respuesta al ítem con un número moderado de dimensiones correlacionadas. El tercer estudio evalúa el desempeño de los algoritmos MHRM, HMC, MML-EM y MCMC en la estimación de un modelo de respuesta al ítem de alta dimensionalidad. Los resultados ponen de manifiesto que MML-EM pierde precisión con modelos de alta dimensionalidad mientras que los otros tres algoritmos recuperan los parámetros verdaderos con una precisión similar. Además, las principales diferencias encontradas entre los algoritmos fueron: 1) MHRM tarda mucho menos en estimar el modelo que el resto de algoritmos; 2) MHRM se muestra más preciso y menos afectado por las distribuciones previas en sus estimaciones; y 3) las distribuciones previas para los parámetros a en los

algoritmos MCMC y HMC deben definirse con cuidado para evitar problemas de orientación de los factores. En resumen, los nuevos algoritmos parecen superar las dificultades de los tradicionales, convergiendo más rápido y obteniendo resultados similares.

REFERENCES

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. *Statistical theories of mental test scores*, 395-479.

Bock, R. D., & Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: An application of the EM algorithm. *Psychometrika, 46,* 443–459.

Cai, L. (2010a). High-dimensional exploratory item factor analysis by a Metropolis-Hastings Robbins-Monro algorithm. *Psychometrika*, 75, 33-57.

Cai, L. (2010b). Metropolis-Hastings Robbins-Monro algorithm for confirmatory item factor analysis. *Journal of Educational and Behavioral Statistics, 35*, 307–335.

Chalmers, P. (2012). mirt: A multidimensional item response theory package for the R environment. *Journal of Statistical Software*, *48(6)*, 1-29.

Chen, F. F., West, S. G., & Sousa, K. H. (2006). A comparison of bifactor and second-order models of quality of life. *Multivariate Behavioral Research*, *41*(2), 189-225.

Christofferson, A. (1975). Factor analysis of dichotomized variables. *Psychometrika, 40*, 5-32.

Curtis, S. M. (2010). BUGS code for item response theory. *Journal of Statistical Software*, *36*(1), 1-34.

Diebolt, J. & Ip, E. H. S. (1996). Stochastic EM: Method and application. In W. R. Gilks, S. Richardson & D. J. Spiegelhalter (Eds.), *Markov chain Monte Carlo in practice*. London: Chapman & Hall.

Gelman, A., Carlin, J. B., Stern, H. S. & Rubin, D. B. (1995). *Bayesian data analysis*. New York: Chapman and Hall.

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

Golay, P., Reverte, I., Rossier, J., Favez, N., & Lecerf, T. (2013). Further insights on the French WISC–IV factor structure through Bayesian structural equation modeling. *Psychological assessment, 25*(2), 496-408.

Han, K. C. T., & Paek, I. (2014). A review of commercial software packages for multidimensional IRT modeling. *Applied Psychological Measurement*, *38*(8), 1-13.

Hoffman, M. D., & Gelman, A. (2014). The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *The Journal of Machine Learning Research*, *15*(1), 1593-1623.

McDonald, R. P. (1985). *Factor analysis and related methods*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Muthén, B., & Asparouhov, T. (2012). Bayesian structural equation modeling: a more flexible representation of substantive theory. *Psychological methods*, *17*(3), 313.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In S. Brooks, A. Gelman, G. L. Jones & X. L. Meng (Eds.), *Handbook of Markov chain Monte Carlo*. Boca Raon, FL. CRC Press.

Ntzoufras, I. (2009). Bayesian modelling using WinBugs. New York: Wiley.

Patrick, C. J., Hicks, B. M., Nichol, P. E., & Krueger, R. F. (2007). A bifactor approach to modeling the structure of the psychopathy checklist-revised. *Journal of personality disorders*, *21*(2), 118.

Patz, R. J., & Junker, B. W. (1999). A straightforward approach to Markov chain Monte Carlo methods for item response models. *Journal of Educational and Behavioral Statistics*, *24*(2), 146–178.

Plummer, M. (2015). *rjags: Bayesian graphical models using MCMC*. Author. Retrieved from http://cran.r-project.org/web/packages/rjags.

R Core Team (2015). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing.

Reise, S. P. (2012). The rediscovery of bifactor measurement models. *Multivariate Behavioral Research*, *47*(5), 667-696.

Schilling, S., & Bock, R. D. (2005). High-dimensional maximum marginal likelihood item factor analysis by adaptive quadrature. *Psychometrika*, *70*(3), 533–555.

Sinharay, S. (2010). How often do subscores have added value? Results from operational and simulated data. *Journal of Educational Measurement*, *47*, 150-174.

Stan Development Team. (2014-a). *Stan modeling language users guide and reference manual*, Version 2.5.0.

Stan Development Team. (2014-b). *Stan: A C++ Library for probability and sampling, Version 2.5*. Retrieved from: http://mc-stan.org.

Wirth, R. J., & Edwards, M. C. (2007). Item factor analysis: Current approaches and future directions. *Psychological Methods, 12*(1), 58–79.

Yung, Y.-F., Thissen, D. & McLeod, L. D. (1999). On the relationship between the higher-order factor model and the hierarchical factor model. *Psychometrika, 64*, 113-128.

Zuckerman, M., & Gagne, M. (2003). The COPE revised: Proposing a 5-factor model of coping strategies. *Journal of Research in Personality*, *37*(3), 169-204.

(Manuscript received: 10 August 2015; accepted: 9 May 2016)