



# MÁSTERES de la UAM

Facultad de Psicología /13-14

Máster en Metodología  
de las Ciencias del  
Comportamiento  
y la Salud



**Métodos de  
Estimación para  
Modelos Multidi-  
mensionales de la  
Teoría de Respuesta  
al Ítem. Un estudio  
de Simulación.**  
*Manuel Martín  
Fernández*



## Índice de Contenidos

Resumen	1
Introducción	2
Modelos MIRT	3
Aplicaciones de los modelos MIRT	6
Estimación MML-EM	8
Estimación Bayesiana	10
Objetivos	12
Método	13
Modelo	13
Simulación	15
Estimación de Parámetros	18
Recuperación de Parámetros	24
Resultados	25
Parámetros de los Ítems	25
Parámetros de las Personas	33
Discusión	38
Referencias	42
Anexo I – Script Openbugs	49
Anexo II – Código de R	51

## **Resumen**

El objetivo de este estudio es comparar el desempeño de dos métodos de estimación (MML-EM y MCMC) al estimar un modelo multidimensional de la teoría de respuesta al ítem de dos dimensiones en distintas condiciones. Para ello se diseñó un estudio de simulación de Monte Carlo en el que se manipularon el tamaño muestral, la longitud del test y la relación entre las dimensiones latentes y se realizaron 50 réplicas de cada condición con cada método. Los resultados muestran que apenas hay diferencias entre ambos procedimientos a la hora de recuperar los parámetros del modelo, si bien la estimación mediante MML-EM probó ser más computacionalmente más eficiente.

*Palabras clave: estimación bayesiana, teoría de respuesta al ítem multidimensional, MCMC, algoritmo EM.*

## **Abstract**

The aim of this study is to compare the performance of the MCMC and the MML-EM algorithms estimating a multidimensional item response theory model with two dimensions in several conditions by simulation of Monte Carlo. The sample size, the test length and the correlation between the latent dimensions were manipulated and 50 replicas were made for each condition with both methods. Results show that both procedures recover the parameters relatively well, but the MML-EM method proved to be computationally more efficient than the Bayesian method.

*Keywords: bayesian estimation, multidimensional item response theory, mcmc, EM algorithm.*

## **Introducción**

El empleo de varios tests o cuestionarios que evalúen distintas variables latentes de forma sucesiva es una práctica común en las ciencias sociales, en especial en sus vertientes más aplicadas donde impera recoger la mayor cantidad de información en un tiempo limitado. Sin embargo, a pesar de medirse diferentes habilidades o rasgos, es habitual obtener las puntuaciones totales de cada cuestionario de forma separada, sin tener en cuenta las posibles relaciones entre las dimensiones. Un ejemplo de esta práctica son los tests de personalidad con varios factores como el EPQS (Eysenck y Eysenck, 1975), el Big Five (Costa y McCrae, 1992), o el 16PF (Catell y Catell, 1995). En todos ellos el perfil de personalidad se obtiene sumando por separado los ítems pertenecientes a cada sub-escala y convirtiendo las puntuaciones directas a alguna métrica normativa que facilite la interpretación de las puntuaciones. Otro caso frecuente son las llamadas “baterías” de tests, donde se aplican varios tests o escalas en sucesión para calcular también aisladamente las puntuaciones totales en cada variable.

Los modelos más sencillos de la teoría de respuesta al ítem (TRI) asumen que cada escala evalúa un único rasgo latente, lo que justificaría la obtención de las puntuaciones por separado. No obstante, trabajos recientes ponen de manifiesto la importancia de tener en cuenta las relaciones entre las variables latentes a la hora de obtener las puntuaciones totales de cada rasgo o dimensión latente (de la Torre y Patz, 2005; Huang, Wang, Chen y Su, 2013; Sheng y Wickle, 2007). Según estas investigaciones, no tener en cuenta las relaciones entre los rasgos latentes puede llevar a estimar con menor precisión la habilidad de los evaluados.

## *Modelos Multidimensionales de la Teoría de Respuesta al Ítem*

Los modelos multidimensionales de la teoría de respuesta al ítem (MIRT<sup>1</sup>, Reckase 1997, 2009) suponen una buena alternativa a la hora de modelar situaciones en las que se espere la influencia de más de un único rasgo latente en la evaluación. Según Reckase (2009), existen dos tipos principales de modelos MIRT. El primero de ellos comprendería aquellas situaciones en las que la probabilidad de dar una respuesta determinada a un ítem concreto depende de más de una única dimensión o rasgo latente. A estos modelos se les denomina *modelos compensatorios*, ya que una dimensión latente baja puede ser compensada por una dimensión latente alta a la hora de dar una respuesta a un ítem. Un ejemplo clásico de este tipo de situaciones sería un ítem con un problema de matemáticas en el que dar con la respuesta correcta dependa de lo buena que sea una persona en cálculo aritmético, pero que también dependa en cierta medida de lo buena que sea esa persona en manipulación de signos algebraicos. De este modo, aquellos evaluados que reúnan un alto dominio en ambas habilidades acertarán en ítem con mayor probabilidad que aquellos con un bajo dominio en las dos habilidades matemáticas. Sin embargo, lo interesante de estos modelos es que las personas con una baja competencia en una de las dos habilidades, pueden llegar a acertar el ítem si tienen un buen dominio de la otra.

La formulación de este tipo de modelos es sencilla, de manera que tan sólo hay que ampliar el número de rasgos latentes definiendo un vector  $\theta = (\theta_1, \dots, \theta_M)'$  con un número  $M$  de dimensiones a modelar. La generalización de los modelos logísticos unidimensionales al caso multidimensional es bien conocida y se formula:

---

<sup>1</sup> Dada la ausencia de trabajos en castellano que aborden estos temas, creo prudente mantener las siglas en inglés de algunos de los términos o procedimientos que se describen en este trabajo, a pesar de “castellanizar” estos términos. En este caso: *Multidimensional Item Response Theory*.

$$P(X_{ij} = 1 | \boldsymbol{\theta}_i, \mathbf{a}_j, c, d) = c_j + (1 - c_j) \frac{\exp(\sum_{m=1}^M a_{jm} \theta_{im} + d_j)}{1 + \exp(\sum_{m=1}^M a_{jm} \theta_{im} + d_j)} \quad (1)$$

donde,

$P(X_{ij} = 1 | \boldsymbol{\theta}_i, \mathbf{a}_j, c_j, d_j)$  es la probabilidad del evaluado  $i$  de acertar el ítem  $j$ ;

$X_{ij}$  es la respuesta del evaluado  $i$  en el ítem  $j$  ( $0 =$  incorrecta,  $1 =$  correcta);

$\boldsymbol{\theta}_i$  es el vector de  $m$  habilidades o dimensiones latentes del evaluado  $i$ , de modo

que  $\boldsymbol{\theta}_i = (\theta_{i1}, \theta_{i2}, \dots, \theta_{im})'$ ;

$\mathbf{a}_j$  es el vector de parámetros de los ítems relacionados con la discriminación, de

modo que  $\mathbf{a}_j = (a_{j1}, a_{j2}, \dots, a_{jm})'$ ;

$d_j$  es el parámetro relacionado con la dificultad del ítem;

$c_j$  es el parámetro de adivinación del ítem;

$m = 1, 2, \dots, M$ ;

$i = 1, 2, \dots, I$ , y

$j = 1, 2, \dots, J$ .

Así, cuanto mayor sea la puntuación de un evaluado en los  $M$  rasgos latentes ( $\theta_1, \dots, \theta_M$ ) mayor será su probabilidad de elegir la respuesta correcta en un test de rendimiento óptimo o, en el caso de los tests de rendimiento típico, de mostrarse de acuerdo con el enunciado ítem. Asimismo, si una de las  $\theta$ -dimensiones toma un valor negativo, la probabilidad de acertar el ítem o de mostrarse de acuerdo con el mismo puede seguir siendo alta si el valor de las otras dimensiones o rasgos latentes es lo suficientemente elevado.

Al igual que en los modelos logísticos unidimensionales, si se fija  $c_j = 1$ , entonces se obtendría el modelo logístico de dos parámetros multidimensional (L2PM).

El parámetro  $d_j$  se denomina intersección y determina el valor de

$P(X_{ij} = 1 | \theta_i, \mathbf{a}_j, c_j, d_j)$  cuando todos los rasgos latentes toman valor cero (Reckase, 2009, capítulo 4). Para facilitar la interpretación de los parámetros, suele transformarse la intersección,  $d_j$ , un parámetro de dificultad,  $b_j$ , aplicando la siguiente expresión (Reckase, 2009, capítulo 4):

$$b_j = \frac{-d_j}{\sqrt{\sum_1^m a_{jm}^2}}$$

(2)

Existen también otro tipo de modelos susceptibles de encajar bajo la categoría de los modelos compensatorios: los llamados *modelos jerárquicos de rasgo latente*<sup>2</sup> (Sheng y Wikle 2008), o sencillamente modelos de orden superior de la TRI (HIRT<sup>3</sup>). Estos modelos se caracterizan por asumir una estructura en la que aparecen dimensiones latentes de distintos órdenes o niveles (Yung, Thissen y McLeod, 1999), donde las dimensiones de orden superior influyen en las respuestas de los evaluados junto a las dimensiones de órdenes inferiores, de una manera similar a la de los modelos logísticos MIRT anteriores. Así, tener un alto valor en la dimensión de orden superior, hace más probable responder correctamente al ítem, aunque las habilidades de orden inferior tengan valores más bajos. Asimismo, aquellas personas con una habilidad de orden inferior alta relevante para responder adecuadamente a un ítem pueden resolverlo con una alta probabilidad aunque su habilidad de orden superior sea baja.

Los modelos HIRT subsumen una gran cantidad de modelos, desde los modelos más generales como el bi-factor (Gibbons y Hedeker, 1992), hasta modelos multinivel (Fox y Glas, 2001; Fox, 2005) o incluso modelos factoriales de distinto orden (Sheng y Wilke, 2008), ampliando los ámbitos de aplicación de los modelos MIRT. Sheng y Wikle (2008) y Huang *et al.* (2013) han generalizado los modelos TRI de ojiva normal

---

<sup>2</sup> *Hierarchical latent trait models.*

<sup>3</sup> *Higher-Order IRT models.*

y los modelos logísticos, respectivamente, para adaptarlos a los modelos jerárquicos. El desarrollo futuro de estos modelos permitirá explorar estructuras latentes más complejas que se ajusten mejor a los datos.

El segundo tipo de modelos MIRT propuesto por Reckase (2009, capítulo 4) da cuenta de aquellas situaciones en las que se están evaluando varios rasgos o dimensiones latentes en un mismo ítem, pero cada dimensión da cuenta de una única parte o proceso del mismo. A estos modelos se les denomina *modelos no-compensatorios* o *modelos parcialmente compensatorios*, en los que tener un alto valor en una dimensión latente no influye o influye levemente el rendimiento en los ítems pertenecientes a otras dimensiones. La formulación y peculiaridades de este tipo de modelos exceden las pretensiones de este trabajo. Para más información sobre los mismos ver Bolt y Lall (2003), Maris (1999) o Reckase (2009).

#### *Aplicaciones de los modelos MIRT*

Una de las ventajas de los modelos MIRT en comparación con los modelos unidimensionales de la TRI es que permiten explorar la dimensionalidad de un test o de una escala. Este procedimiento se conoce como *análisis factorial del ítem* (IFA<sup>4</sup>) y fue acuñado y desarrollado originalmente en los trabajos de McDonald (1967, 1985, 1999) como una alternativa a los modelos lineales del análisis factorial clásico. Si bien se han desarrollado distintos procedimientos para un tratamiento más adecuado de datos ordinales y dicotómicos en el análisis factorial (ver Abad, Olea, Ponsoda y García, 2011), el IFA parece una opción más adecuada para trabajar con datos de estas características (Wirth y Edwards, 2007). Estos autores recomiendan el empleo del IFA

---

<sup>4</sup> *Item Factor Analysis.*

al tratar con datos ordinales con cuatro o menos categorías –o incluso con cinco categorías (DiStefano, 2002), especialmente cuando el objetivo sea examinar las características psicométricas de una escala y las relaciones entre las distintas variables latentes.

Lord y Novick (1968) y Takane y Leeuw (1987) establecieron y demostraron – respectivamente- la correspondencia entre los parámetros del modelo L2PM y los parámetros del análisis factorial:

$$a_j = \frac{\lambda_j}{\sqrt{1-\lambda_j}} \quad \text{y} \quad b_j = \frac{\tau_j}{\lambda_j} \quad (3)$$

donde  $\lambda_j$  es la saturación o peso factorial del ítem  $j$  y  $\tau_j$  es el umbral entre una categoría y otra en el ítem  $j$ . Muraki y Carlson (1995) generalizaron también el modelo de respuesta graduada (Samejima, 1969) al caso multidimensional, vinculando el nuevo modelo con un modelo factorial clásico a través de una expresión muy parecida a (3) (ver Revuelta, Abad y Ponsoda 2006).

Otro de los principales puntos fuertes de los modelos MIRT es que permiten explorar estructuras latentes más complejas. Para abordar datos con este tipo de estructuras es habitual el empleo de los modelos de ecuaciones estructurales. El ejemplo más sencillo son los modelos factoriales confirmatorios, dónde distintos grupos de ítems –o de parcelas de ítems (Little *et al.*, 2002)- “saturan” o cargan en uno o en varios factores diferentes. Se forzaría así a que los ítems un test que evalúe distintos rasgos latentes saturen sólo en la dimensión a la que teóricamente pertenecen, evitando situaciones en las que un mismo ítem sea tomado como indicador de varios factores si este hecho no se contempla en el modelo original.

Los modelos MIRT pueden utilizarse asimismo como una alternativa al análisis factorial confirmatorio clásico para datos dicotómicos u ordinales. Edwards (2010)

propone el término *análisis factorial confirmatorio del ítem* (CIFA<sup>5</sup>) para referirse a este procedimiento. Al igual que en los modelos factoriales confirmatorios, la idea subyacente es aplicar la expresión (1) fijando algunos parámetros del modelo a un valor concreto. De esta manera, para modelar un test que evalúese más de un rasgo latente a partir de un conjunto de ítems unidimensionales, tan sólo habría que fijar a 0 los parámetros del vector  $\mathbf{a}_j$  que no perteneciesen a la dimensión que mide cada ítem.

Finalmente, también es posible el uso de los modelos HIRT para modelar estructuras de datos complejas donde sea preciso contemplar un gran número de habilidades latentes en distintos niveles. Esta es un área incipiente de los modelos MIRT en los que algunos trabajos muestran ya resultados prometedores (de la Torre y Douglas, 2004; de la Torre y Song, 2009; Sheng y Wilke, 2009). Si bien es cierto que los modelos de ecuaciones estructurales incorporan métodos de estimación específicamente diseñados para tratar con datos categóricos o dicotómicos (DWLSM; Muthén y Muthén, 2006) relativamente precisos (Wirth y Edwards, 2007), los modelos MIRT empiezan a perfilarse como una buena alternativa para analizar este tipo de datos.

#### *Estimación MML-EM*

La consecuencia más inmediata que tiene el paso de los modelos unidimensionales de la TRI a los modelos MIRT es el aumento en la dificultad de la estimación de los parámetros del modelo. La estrategia más habitual para enfrentarse a este problema consiste en calibrar primero los ítems mediante máxima verosimilitud marginal (MML<sup>6</sup>), tratando las variables latentes como valores perdidos y utilizando el algoritmo de *expectación-maximización* (EM) para maximizar la función de

---

<sup>5</sup> *Confirmatory Item Factor Analysis.*

<sup>6</sup> *Marginal Maximum Likelihood.*

verosimilitud. A continuación se toman los parámetros de los ítems como conocidos y se estiman las puntuaciones en las variables latentes de cada evaluado. Bock y Aitkin (1981) fueron los primeros en aplicar el algoritmo EM en TRI. El método necesita evaluar de forma numérica la integral involucrada en la distribución marginal de las respuestas observadas, empleando para ello el método de Gauss-Hermite. A esta aproximación se la conoce como MML-EM, y está implementada en distintos programas como TESTFACT (Bock *et al.*, 2002), M-Plus (Muthén y Muthén, 2006) y en R (*mirt package*; Chalmers, 2012).

Supongamos que tenemos una matriz de respuestas  $\mathbf{X}$ , con  $I$  sujetos y  $J$  ítems. Según la ecuación (1), la probabilidad del vector de respuestas  $\mathbf{x}_i$  de cada sujeto está determinada por un número  $m$  de dimensiones latentes del sujeto y un vector  $\xi$  de parámetros de los ítems  $a_{jm}$  y  $d_j$ . El método de estimación MML-EM con integración de Gauss-Hermite utiliza un número  $Q$  de puntos de cuadratura para evaluar la integral sobre cada dimensión latente. El problema de integración quedaría:

$$f_{\xi}(\mathbf{x}) = \sum_{qM=1}^Q \dots \sum_{q2=1}^Q \sum_{q1=1}^Q f_{\xi}(\mathbf{x}|\theta_{q1}, \dots, \theta_{qM}) W_{q1} \dots W_{qM} \quad (4)$$

donde  $f_{\xi}(\mathbf{x})$  es la función de probabilidad marginal del patrón de respuestas  $\mathbf{x}_i$ ,  $f_{\xi}(\mathbf{x}|\theta_{q1}, \dots, \theta_{qM})$  es la distribución de  $\mathbf{x}_i$  condicionada en las  $M$  dimensiones latentes, y  $W_{qm}$  son los pesos asignados al punto  $q$  de la dimensión  $m$ . Los detalles más técnicos concernientes tanto a la formulación como al desarrollo del método aplicado al IFA o a los modelos MIRT pueden consultarse en Bock y Aitkin (1981), Cai, Yang y Hansen (2010) o Chalmers (2012).

Este método funciona bien para modelos con pocas dimensiones, pero comienza a dar problemas cuando aumenta el número de rasgos latentes en el modelo (Junker y Patz, 1999) o bien cuando el número de parámetros de los ítems es muy elevado

(Wollack *et al.*, 2002), pues a medida que se incrementa el número de dimensiones,  $M$ , se incrementa exponencialmente el número de puntos de cuadratura necesarios para estimar adecuadamente el modelo (Bock *et al.*, 1988; Muraki y Carlson, 1995). En concreto, el número de puntos de cuadratura involucrados en la ecuación (4) es  $M^Q$ , lo que puede llegar a hacer inviable la estimación de los modelos MIRT a partir de siete u ocho dimensiones latentes. Schilling y Bock (2005) propusieron una solución parcial consistente en utilizar cuadratura adaptativa, en la que los valores de  $W_{qm}$  son distintos para cada patrón de respuestas. La cuadratura adaptativa consigue mayor precisión, con lo que necesita menos puntos de cuadratura y es aplicable con más dimensiones latentes que la cuadratura estática. Aun así, el problema persiste para modelos con un gran número de variables latentes y es necesario encontrar métodos alternativos.

#### *Estimación Bayesiana*

A diferencia del método de estimación MML-EM, los métodos de estimación bayesiana abordan el problema de estimación empleando métodos de simulación, aunque el número de parámetros o de dimensiones sea muy elevado. A pesar de que conceptualmente ambos métodos son parecidos, esta aproximación es mucho más fácil de implementar, ya que elude el coste computacional de los puntos de cuadratura o del cálculo de derivadas. Sin embargo, el tiempo de ejecución de la estimación tiende a ser mayor.

Desde esta perspectiva, se asume que los parámetros del modelo son una variable aleatoria que sigue su propia distribución. El proceso de inferencia de estos métodos consiste en obtener estimaciones de una distribución posterior de cada parámetro. Si, al igual que antes, tenemos una matriz  $\mathbf{X}$  de  $I$  sujetos por  $J$  ítems,

tomamos  $\xi$  como el vector de parámetros de los ítems y  $\theta$  como el vector de  $M$  rasgos latentes o dimensiones, podemos obtener la distribución posterior aplicando el teorema de Bayes:

$$f(\theta, \xi | \mathbf{x}) = \frac{f(\mathbf{x} | \theta, \xi) f(\theta)}{f(\mathbf{x})} \propto f(\mathbf{x} | \theta, \xi) f(\theta) \quad (5)$$

donde  $f(\theta, \xi | \mathbf{x})$  es la distribución a-posteriori,  $f(\theta)$  es la distribución a-priori y  $f(\mathbf{x} | \theta, \xi) = \prod_{i=1}^I f(x_i | \theta)$  es la función de verosimilitud. Por tanto, la distribución posterior es proporcional a la función de verosimilitud multiplicada por la distribución a-priori.

La estrategia más utilizada para calcular la distribución posterior es simularla mediante *Monte Carlo con cadenas de Markov* (MCMC<sup>7</sup>). Una cadena de Markov es un proceso estocástico consistente en la generación de una serie de variables aleatorias que están ordenadas en el tiempo. El valor que toma la cadena en el momento  $t$  depende sólo del valor que tomó en el momento  $t - 1$  y no en momentos anteriores (salvo en  $t^{(0)}$ , cuyo valor se toma al azar de la distribución a-priori). La distribución de las muestras simuladas va evolucionando a medida que la cadena avanza. Eventualmente, la cadena converge a la *distribución equilibrio* o estacionaria. La forma de tomar muestras en las cadenas de Markov -utilizando el *Gibbs Sampling*- garantiza que dicha distribución estacionaria es la distribución *objetivo* (Gilks *et al.*, 1996), que en este caso viene definida por la ecuación (5). Para evitar una influencia excesiva de la distribución a priori, lo que suele hacerse es emplear distribuciones a priori *no informativas* (Gelman, Stern y Rubin, 1995). Además, como los valores iniciales son arbitrarios, se intenta

---

<sup>7</sup> *Markov Chain Monte Carlo.*

eliminar su influencia del resultado final eliminando la primera mitad de las cadenas simuladas (*Burn-in period*, ver Ntzoufras, 2009).

Lo que se consigue con los distintos algoritmos MCMC es tomar un número representativo de muestras de la distribución posterior mediante las cadenas de Markov con el que calcular el estimador bayesiano esperado a posteriori (EAP) mediante la media de las muestras. Asimismo, la desviación típica del estimador daría información sobre el error de estimación (Patz y Junker, 1999).

Existen distintos programas informáticos que facilitan la estimación de modelos aplicando estos algoritmos, como Winbugs (Lunn, Thomas, Best, Spiegelhalter, 2000), STAN (Stan Development Team, 2014) o JAGS (Plumer, 2003). Gracias a ellos la estimación mediante MCMC está proliferando y cobrando un gran protagonismo en la literatura académica, tanto a nivel teórico (ver, entre otros, Kim y Bolt, 2007; Levy, 2009) como a nivel aplicado (ver, Fox, 2010).

### *Objetivos*

Como se ha ilustrado anteriormente, los métodos de estimación habituales (MML-EM) funcionan bien cuando el número de parámetros o de dimensiones latentes incluidos en el modelo es moderado. Sin embargo, al aumentar la complejidad y, especialmente, el número de variables latentes en el modelo, la estimación MML-EM se vuelve inaplicable dado el alto número de puntos de cuadratura que requeriría. Los métodos bayesianos de estimación –como el *Gibbs Sampling*– suponen una buena alternativa cuando no pueden obtenerse los estimadores máximo-verosímiles. El propósito de este trabajo es estudiar mediante simulación de Monte Carlo qué método de estimación recupera mejor los parámetros de un modelo MIRT relativamente sencillo

bajo distintas condiciones. Dada la sencillez del modelo empleado en este estudio, se espera que la estimación mediante MCMC sea al menos tan precisa como la estimación MML-EM tanto en la recuperación de los parámetros de los ítems como en los parámetros de las personas.

## Método

Para poner a prueba el desempeño de ambos métodos de estimación se eligió un modelo MIRT fácilmente transferible a situaciones aplicadas: un modelo CIFA con tan sólo dos dimensiones en el que la mitad de los ítems pertenece a un factor y la otra mitad al otro. La idea es trabajar con modelo que pueda estimarse mediante MCMC con relativa rapidez y comparar los resultados con una estimación MML-EM, estudiando además el efecto de algunas variables independientes como el tamaño muestral, la longitud del test o la relación entre los factores sobre la precisión en las estimaciones.

### *Modelo*

El modelo utilizado en este estudio es una versión bidimensional del modelo L2PM, de forma que:

$$P(X_{ij} = 1 | \boldsymbol{\theta}_i, \mathbf{a}_j, d_j) = \frac{\exp(a_{j1}\theta_{i1} + a_{j2}\theta_{i2} + d_j)}{1 + \exp(a_{j1}\theta_{i1} + a_{j2}\theta_{i2} + d_j)} \quad (6)$$

donde, al igual que en (1),

$P(X_{ij} = 1 | \boldsymbol{\theta}_i, \mathbf{a}_j)$  es la probabilidad del evaluado  $i$  de acertar el ítem  $j$ ;

$X_{ij}$  es la respuesta del evaluado  $i$  en el ítem  $j$  (0 = incorrecta, 1 = correcta);

$\boldsymbol{\theta}_i$  es el vector de  $m$  habilidades o dimensiones latentes del evaluado  $i$ , de modo

que  $\theta_i = (\theta_{i1}, \theta_{i2})'$ ;

$a_j$  es el vector de parámetros de los ítems relacionados con la discriminación, de

modo que  $a_j = (a_{j1}, a_{j2})'$ ;

$d_j$  es el parámetro relacionado con la dificultad del ítem;

$i = 1, 2, \dots, I$ , y

$j = 1, 2, \dots, J$ .

Se asume que las dimensiones latentes  $\theta$  se distribuyen de acuerdo a una distribución normal multivariante (NMV), de forma que es posible modelar la relación entre las dos dimensiones. Se pretende además emplear este modelo como un CIFA en el que se supone que cada ítem mide sólo uno de los dos rasgos latentes, de modo que cuando  $a_{j1} > 0$  entonces  $a_{j2} = 0$  y viceversa. Así, a diferencia de los modelos compensatorios convencionales, las curvas características de cada ítem toman la misma forma que en el modelo logístico de dos parámetros unidimensional. Tal y como se observa en la figura 1, la probabilidad de dar la respuesta correcta depende sólo de la dimensión a la que pertenezca cada ítem.

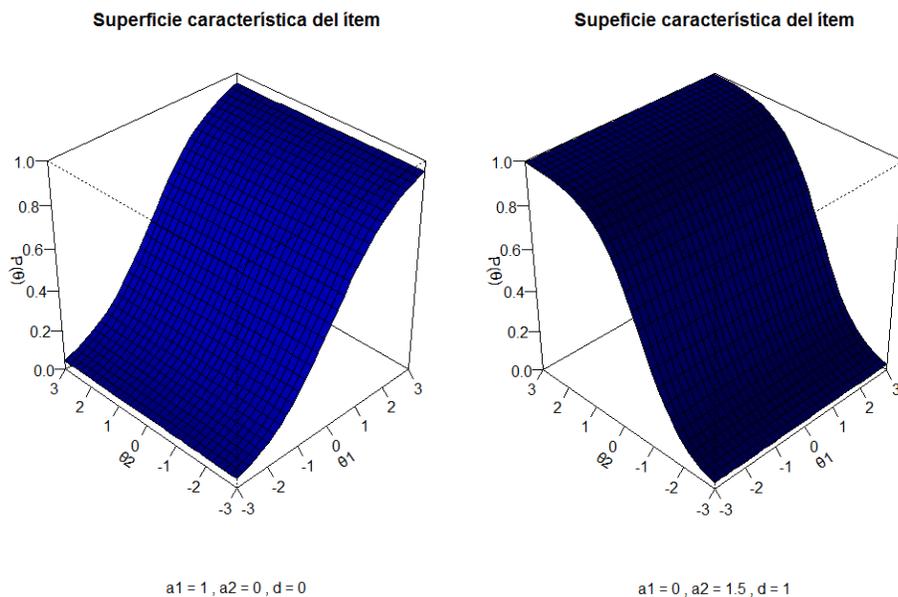


Figura 1. Superficies características del modelo para  $a_1 = 1, a_2 = 0, d = 0$  y para  $a_1 = 0, a_2 = 1.5$  y  $d = 1$ .

Asimismo, al asumirse que los ítems son unidimensionales, estos sólo serán informativos para su propia dimensión. Según Reckase (2009), la función de información de cada ítem se define como:

$$I_{\alpha}(\boldsymbol{\theta}) = P(\boldsymbol{\theta})Q(\boldsymbol{\theta}) \left( \sum_{m=1}^M a_m \cos \alpha_m \right)^2 \quad (7)$$

donde  $P(\boldsymbol{\theta})$  es la expresión (6),  $Q(\boldsymbol{\theta})$  es  $1-P(\boldsymbol{\theta})$  y  $\alpha_m$  es el ángulo o dirección en el espacio  $\boldsymbol{\theta}$  que se toma desde un punto  $\boldsymbol{\theta}$  determinado. El valor de  $\alpha_m$  depende del vector de parámetros  $\boldsymbol{a}$  de cada ítem. Si la dirección de  $\alpha_m$  es la pendiente con mayor inclinación –o discriminación–, entonces la información del ítem se define como:

$$I_{\alpha_{\max}}(\boldsymbol{\theta}) = P(\boldsymbol{\theta})Q(\boldsymbol{\theta}) \sum_{m=1}^M a_m^2 \quad (8)$$

Si, como en este caso, sólo uno de los parámetros  $a_j$  del ítem toma un valor distinto de cero, entonces (8) queda igual que la función de información para el modelo de dos parámetros (Birnbaum, 1968):

$$I(\theta_m) = a_m^2 P(\theta_m) Q(\theta_m) \quad (9)$$

### *Simulación*

Se diseñó un estudio de simulación que pone a prueba el desempeño de dos métodos de estimación a la hora de recuperar los parámetros del modelo bajo una serie de condiciones. Las variables independientes manipuladas en este estudio fueron: (a) la correlación entre las dos dimensiones latentes del modelo, (b) la longitud del test a través del número de ítems de cada dimensión, (c) el tamaño muestral y (d) el método de estimación. Las correlaciones entre las dimensiones que se utilizaron en este estudio

fueron de 0 y 0.60; los diferentes números de ítems fueron 5 y 10 por dimensión (longitud del test de 10 ítems ó 20 ítems respectivamente); los tamaños muestrales considerados fueron de 200 y de 500 sujetos; y los métodos de estimación fueron MML-EM y MCMC. Se seleccionaron estos niveles para las variables independientes tratando de emular condiciones de estimación plausibles en la práctica aplicada, utilizando tamaños muestrales “asequibles”, longitudes del test sencillas de conseguir, y procurando que la relación entre las dimensiones sea alta cuando no se asuma independencia para apreciar el efecto de esta condición en el estudio.

Se cruzaron todas las posibles combinaciones, resultando un total de ocho condiciones a estimar por cada método (ver tabla 1). Se realizaron 50 réplicas por cada condición.

Tabla 1.  
*Condiciones del estudio de simulación.*

	Dimensiones Independientes ( $\rho = 0$ )		Dimensiones Relacionadas ( $\rho = .6$ )	
	10 ítems	20 ítems	10 ítem	20 ítems
200 Evaluados	<i>Condición 1</i>	<i>Condición 2</i>	<i>Condición 5</i>	<i>Condición 6</i>
500 Evaluados	<i>Condición 3</i>	<i>Condición 4</i>	<i>Condición 7</i>	<i>Condición 8</i>

Las cincuenta muestras de datos de cada réplica se simularon de acuerdo con el modelo de la expresión (6). Los parámetros de los ítems fueron generados para que perteneciesen a una única dimensión, de modo que cada rasgo latente es medido por la mitad de los ítems del test. Los parámetros  $a_j$  del modelo o pendientes fueron extraídos de una distribución uniforme con mínimo 0.3 y máximo 2.5 para la dimensión a la que pertenecen, y fijados a 0 en la otra dimensión latente del test. Los parámetros  $b_j$  fueron extraídos de una normal con parámetros  $\mu = 0$  y  $\sigma^2 = 1$ , y convertidos después en

parámetros  $d_j$  o intersecciones para adecuar el modelo a la expresión (6). Por su parte, los parámetros de las personas fueron simulados a partir de una normal multivariante con parámetros

$$\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ y } \Sigma = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix},$$

donde  $\rho = 0$  ó  $\rho = 0.6$  en función de la condición. Los parámetros verdaderos empleados para simular las respuestas aparecen en la tabla 2.

Dado que se estimó la matriz  $\Sigma$  completa, se fijó a 1 el primer parámetro  $a$  de cada una de las dimensiones, evitando así problemas de identificación del modelo.

Tabla 2.  
*Parámetros verdaderos e información máxima de cada ítem.*

	$a_{j1}$	$a_{j2}$	$d_j$	Max Info
Ítem 1	1	0	-1.56	0.25
Ítem 2	2.28	0	0.32	0.57
Ítem 3	2.17	0	-0.4	0.54
Ítem 4	2.02	0	0.35	0.5
Ítem 5	0.53	0	-1.5	0.13
Ítem 6	0	1	-0.39	0.25
Ítem 7	0	1.04	-0.21	0.26
Ítem 8	0	1.52	0.01	0.38
Ítem 9	0	0.98	2.83	0.24
Ítem 10	0	0.86	-2.31	0.21
Ítem 11	2.33	0	-0.76	0.58
Ítem 12	1.12	0	-0.05	0.28
Ítem 13	1.26	0	-0.74	0.31
Ítem 14	1.01	0	0.41	0.25
Ítem 15	1.5	0	-0.15	0.37
Ítem 16	0	0.87	1.21	0.22
Ítem 17	0	1.79	0.78	0.45
Ítem 18	0	0.91	0.91	0.22
Ítem 19	0	0.64	3.54	0.09
Ítem 20	0	0.91	0.94	0.23

La información conjunta en ambos test –de 10 y de 20 ítems- se ilustra en la *figura 2*. En ella se aprecia cómo se incrementa la información de ambos tests a medida que los valores de  $\theta$  se aproximan a cero en las dos dimensiones.

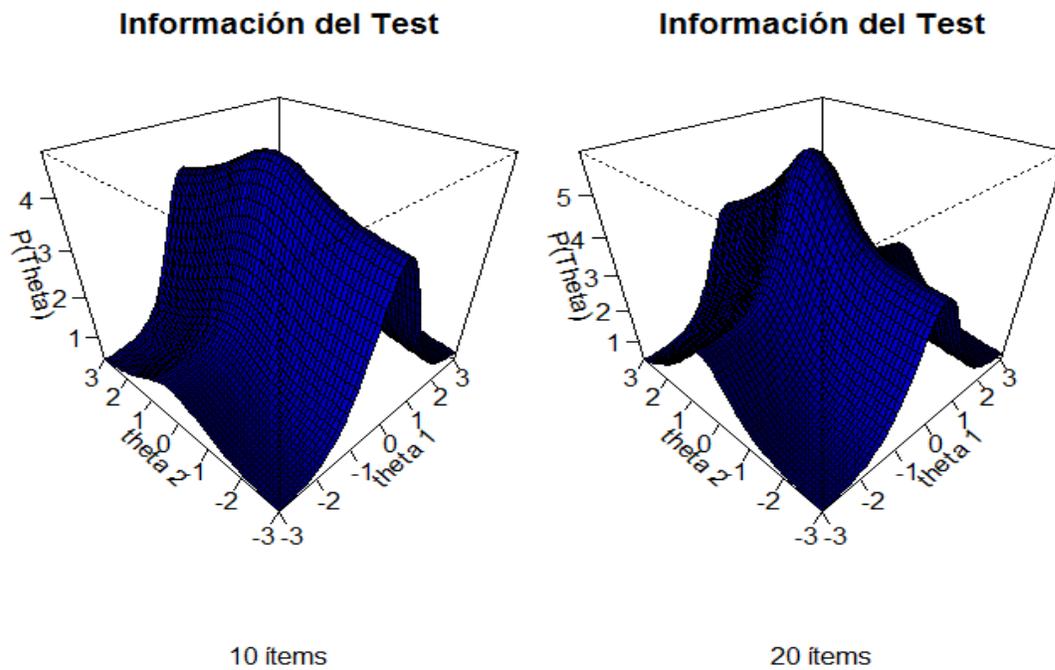


Figura 2. Superficie de Información del Test.

### *Estimación de parámetros*

La estimación MML-EM se realizó con el paquete *mirt* de R (Chalmers, 2012), mientras que la estimación MCMC se llevó a cabo a través de paquete *BRugs* (Thomas, O’Hara, Ligges y Sturtz, 2006), una interfaz de R que llama al programa Openbugs (Lunn *et al.*, 2000) en “*batch mode*”. A continuación se describen las especificaciones seguidas en cada método de estimación.

### Estimación MML-EM vía *mirt*

Para llevar a cabo la estimación por este procedimiento lo primero que se hizo fue fijar los valores iniciales de los parámetros a estimar al valor que el programa incorpora por defecto. El paquete *mirt* utiliza 0.851 como valor inicial para los

parámetros de discriminación y un valor aleatorio extraído de la distribución normal (con  $\mu = 0$  y  $\sigma^2 = 1$ ) para la intersección o parámetro  $d_j$ . Para los parámetros  $\theta_i$  de las personas, se fijó a 0 la media y se dejó libre la matriz  $\Sigma$ . Los valores iniciales de esta matriz fueron los de una matriz identidad salvo en las condiciones con dimensiones independientes, donde se fijó a 0 el valor de las covarianzas.

Para estimar los parámetros de las personas se utilizaron 21 puntos de cuadratura por dimensión (el valor por defecto que incorpora el programa), y para calcular el error típico de estimación se utilizó el procedimiento de Bock y Lieberman (1970). El algoritmo se programó para que llegase a un máximo de 500 iteraciones –el valor establecido por defecto, de modo que parase con el último valor estimado si no converge en una solución antes de la última iteración. No se han registrado problemas de convergencia salvo en una única ocasión –con 200 sujetos y 10 ítems- en la que el algoritmo tuvo que finalizar en la iteración 500.

#### Estimación MCMC vía Openbugs

Antes de empezar la estimación por este método fue necesario establecer las distribuciones a-priori de cada uno de los parámetros del modelo. En este estudio se manejaron las siguientes distribuciones a-priori:

$$a_{jm} \sim \text{Gamma}(0.25, 0.25); E(a) = 1, \text{Var}(a) = 4$$

$$d_j \sim N(0, 1/4); E(d) = 0, \text{Var}(d) = 4$$

$$\theta_i | \Sigma \sim \text{NMV}(0, \Sigma); E(\theta) = \{0, 0\}, \text{Var}(\theta) = \Sigma$$

$$\Sigma \sim \text{Inv-Wishart}(I, 2); E(a) = 2I, \text{Var}(\Sigma) = 4$$

Al igual que con la estimación MML-EM, la matriz  $\Sigma$  sólo se estima entera en las condiciones con dimensiones latentes relacionadas. En cambio, en las condiciones con dimensiones independientes, las covarianzas se fijaron a 0 y se estimaron sólo las varianzas. Del mismo modo, se fijó el primer parámetro  $a_{jm}$  de cada dimensión a 1 para

evitar problemas de identificación del modelo, y sólo se estimaron los parámetros  $a_{jm}$  pertenecientes a la dimensión medida por cada ítem, fijando a 0 los demás. Además, utilizando la distribución *Gamma* con los hiper-parámetros  $\alpha = 0.25$  y  $\beta = 0.25$  como distribución a priori de  $a_{jm}$ , se obtiene una distribución con media 1 y varianza 4.

Openbugs usa por defecto la distribución normal con parámetro de precisión  $\tau$  en lugar del parámetro habitual  $\sigma^2$ , de manera que  $\sigma^2 = 1/\tau$ . Se ha utilizado como distribución a-priori de  $d_j$  una normal con hiper-parámetros  $\mu = 0$  y  $\tau = 4$ , por lo que su media es 0 y la varianza es  $\sigma^2 = 1/\tau = 0.25$ .

Como distribución a-priori de la matriz de varianzas-covarianzas,  $\Sigma$ , se utilizó la distribución *Wishart* (Ntzoufras, 2009), con hiper-parámetros  $\nu$  y  $\mathbf{A}$ . De este modo

$$E(\Sigma) = \nu \mathbf{A} \quad \text{y} \quad \text{Cov}(\sigma_{hi}, \sigma_{jk}) = \nu(A_{hj}A_{ik} + A_{hk}A_{ij})$$

En este caso se tomó  $\mathbf{A}$  como una matriz identidad y el parámetro de escala  $\nu$  como el número de dimensiones latentes. Como distribución a-priori de  $\theta$ , se utilizó una distribución normal multivariante con parámetros  $(\boldsymbol{\mu} = \mathbf{0}, \Sigma)$ , donde  $\Sigma$  no toma un valor constante sino que es el valor muestreado de la distribución Wishart, según se ha indicado.

Por su parte, si utilizamos la notación  $X = \{X_{ij}\}$  como los datos,  $\theta = (\theta_i)$  como el vector de dimensiones latentes y  $\xi = \{a_{jm}, d_j\}$  como el vector de parámetros de cada ítem, la distribución condicionada posterior queda como:

$$P(\theta, \Sigma | X, \xi) \propto P(X | \theta, \Sigma, \xi) P(\theta, \Sigma) \tag{10}$$

si a su vez se le aplica el teorema de Bayes sobre  $P(\theta, \Sigma)$ :

$$P(\Sigma | \theta) \propto P(\theta | \Sigma) P(\Sigma), \tag{11}$$

la distribución a-posteriori condicionada queda finalmente como:

$$P(\theta, \Sigma | X, \xi) \propto P(X | \theta, \Sigma, \xi) P(\theta | \Sigma) P(\Sigma) \tag{12}$$

Para tomar muestras de la distribución a-posteriori se empleó el algoritmo MCMC conocido como *Gibbs Sampling* (Geman y Geman, 1984), un caso especial del algoritmo *Metropolis-Hasting*. A continuación se definen los pasos del algoritmo:

(1) Inicializar los parámetros asignándoles un valor aleatorio de sus respectivas distribuciones a-priori.

(2) Para un número  $T$  de iteraciones, se extraen muestras de cada parámetro a estimar de su distribución condicional completa dados los valores actuales de los demás parámetros y de los datos observados. Es decir, si tomamos  $\gamma$  como el vector completo de parámetros a estimar (aunando por tanto los parámetros de los ítems y de las personas:  $\gamma = (\theta, \xi)$ ) el valor del parámetro  $\gamma_n$  en la iteración  $t$  de la cadena se obtiene extrayéndolo de  $P(\gamma_n | X, \gamma_1^t, \gamma_2^t, \dots, \gamma_{n-1}^t, \gamma_{n+1}^{t-1}, \dots, \gamma_N^{t-1})$ .

Una iteración  $t$  completa se obtiene sacando valores secuencialmente de

$$\begin{aligned}
 \gamma_1^t &\sim P(\gamma_1 | X, \gamma_2^{t-1}, \dots, \gamma_N^{t-1}) \\
 \gamma_2^t &\sim P(\gamma_2 | X, \gamma_1^t, \gamma_3^{t-1}, \dots, \gamma_N^{t-1}) \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 \gamma_N^t &\sim P(\gamma_N | X, \gamma_1^t, \dots, \gamma_{N-1}^t)
 \end{aligned} \tag{13}$$

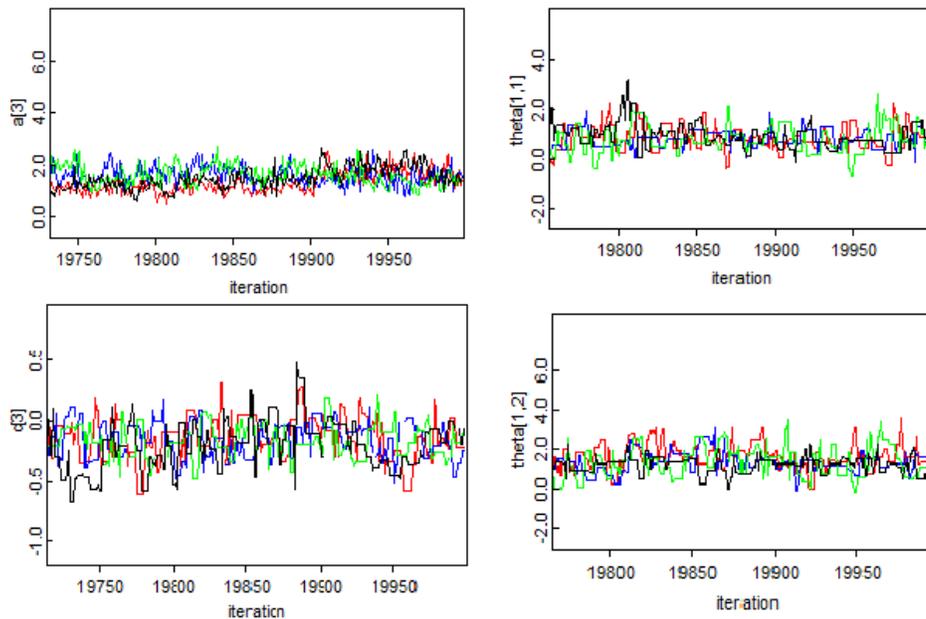
(3) Repetir los pasos 1 y 2 durante la  $T$  iteraciones.

La principal ventaja del *Gibbs Sampling* en comparación con otros algoritmos MCMC es que permite extraer muestras de cada parámetro por separado cuando se trabaja con una distribución conjunta (Gilks, 1996), lo que reduce considerablemente el número de parámetros condicionados en cada una de las distribuciones del segundo paso (Levy, 2009).

En este trabajo se fijó  $T = 20000$  y se inicializaron 4 cadenas diferentes. Sin embargo, se descartaron las primeras 10000 iteraciones (*Burn-in period*) y el estimador EAP de cada parámetro se obtuvo calculando la media de las últimas 10000 iteraciones:

$$\hat{\gamma}_n = \frac{1}{10000} \sum_{t=10001}^T \gamma_n^t \quad (14)$$

En la *figura 3* se ilustra a modo de ejemplo el historial de las últimas iteraciones de las cuatro cadenas de los parámetros  $a_{31}$ ,  $d_3$  y del parámetro  $\theta_l$  en una de las réplicas de la condición 7 (factores relacionados, 200 sujetos y 20 ítems). Ntzoufras (2009) recomienda comprobar el historial de iteraciones para verificar que las diferentes cadenas converjan hacia un mismo valor.



*Figura 3. Historial de iteraciones*

Si bien existen estadísticos concretos para estudiar la convergencia del algoritmo, como el  $\sqrt{R}$  (Gelman y Rubin, 1992), no fue posible calcular los valores numéricos de este estadístico para el total de los parámetros de las estimaciones

MCMC, ya que prolongaba en exceso el tiempo de estimación a través del paquete *BRugs* (llegando a duplicar o incluso triplicar la duración del proceso). Para evaluar la convergencia general de los parámetros sí que se obtuvieron posteriormente representaciones gráficas de algunas de las réplicas y se comprobó que al menos este estadístico siguiese el mismo patrón en cada una de ellas.

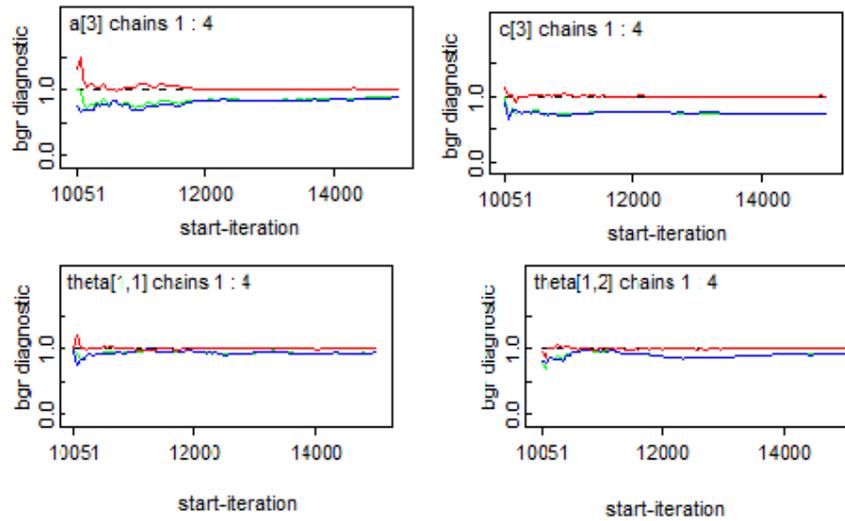


Figura 4. Estadísticos  $\sqrt{R}$ .

Las líneas rojas de la *figura 4* corresponderían al estadístico  $\sqrt{R}$ . Si este estadístico tiende a 1, es indicativo entonces de que no hay no-convergencia entre las cadenas durante la estimación del parámetro (Brooks y Gelman, 1998; Gelman y Rubin, 1992). Se considera que la convergencia es aceptable si este estadístico toma valores cercanos a 1, pues este estadístico se basa en el cociente entre la varianza de un parámetro entre las cadenas frente a la varianza dentro de las cadenas. Cuando las distintas cadenas convergen y las muestras oscilan en torno al valor del estimador, la varianza inter- y la varianza intra-cadena son similares, por lo que el cociente tiende a 1. En general parece que la convergencia no es mala entre las distintas cadenas en las diferentes condiciones.

### *Recuperación de parámetros*

Se utilizaron el sesgo, y la raíz del error cuadrático medio (RMSE) de las 50 réplicas de cada condición para evaluar la precisión de las estimaciones de cada método.

$$sesgo(\hat{\gamma}_r) = \frac{1}{N} \sum_{n=1}^N \hat{\gamma}_{rn} - \gamma_r$$

$$RMSE(\hat{\gamma}_r) = \sqrt{\frac{1}{N} \sum_{n=1}^N (\hat{\gamma}_{rn} - \gamma_r)^2}$$

donde  $n$  son las réplicas,  $N = 50$ ,  $\hat{\gamma}_{rn}$  es el parámetro estimado en la réplica  $n$  y  $\gamma_r$  es el parámetro verdadero.

El sesgo y el RMSE serán tomados como las variables dependientes a la hora de comparar el desempeño de cada método de estimación, teniendo presente que el RMSE resume tanto la precisión como la dispersión de las estimaciones.

Asimismo, para analizar la precisión a la hora de recuperar tanto los parámetros de los ítems como de las personas, se empleará también la correlación media de Pearson entre la media de los parámetros estimados y verdaderos:

$$r_{\hat{\gamma}\gamma} = cor\left(\frac{1}{N} \sum_{n=1}^N \hat{\gamma}_n, \gamma\right)$$

donde  $N=50$ ,  $\hat{\gamma}$  sería el vector con la media de cada uno de los parámetros estimados y  $\gamma$  sería el vector de parámetros verdaderos.

### *Materiales*

Tanto la simulación de las muestras de datos como el proceso de estimación del modelo se realizaron en el mismo equipo informático: un Toshiba Satellite L50-A con un procesador de cuatro núcleos Intel-Core i7-4700MQ a 240GHz y con 4GB de

memoria RAM. Para analizar los resultados se empleó el programa estadístico *R* (R Core Team, 2013).

## Resultados

### *Parámetros de los Ítems*

El sesgo de cada parámetro de los ítems pueden observarse en las tablas 3 y 4, y en las *figuras 5 y 6*. En general los sesgos son bajos, aunque hay algún parámetro que ambos métodos tienen dificultades para estimar, especialmente cuando la longitud del test es menor.

Tabla 3.

*Parámetros  $a$  verdaderos y sesgo de los parámetros  $\hat{a}$  estimados.*

Ítems	par. Verdadero	MML-EM				MCMC			
		D. Independientes ( $\rho=0$ )		D. Relacionadas ( $\rho=0.6$ )		D. Independientes ( $\rho=0$ )		D. Relacionadas ( $\rho=0.6$ )	
		N = 200	N = 500						
10	$a_{1,1} = 1$	-	-	-	-	-	-	-	-
	$a_{2,1} = 2.28$	0.71	0.04	0.76	0.02	0.12	0.22	0.38	0.1
	$a_{3,1} = 2.17$	0.58	0.02	0.7	0.07	0.33	0.25	0.57	0.15
	$a_{4,1} = 2.02$	0.37	-0.02	0.43	0.03	0.11	0.13	0.32	0.08
	$a_{5,1} = 0.53$	0.06	0.03	0.2	0.02	0.02	-0.03	-0.09	-0.06
	$a_{6,2} = 1$	-	-	-	-	-	-	-	-
	$a_{7,2} = 1.04$	0.35	0.07	0.16	0.09	0.42	0.14	0.19	0.07
	$a_{8,2} = 1.52$	0.65	0.5	0.49	0.15	0.87	1.23	0.8	0.25
	$a_{9,2} = 0.98$	0.19	0.21	0.18	0.12	-0.49	-0.36	-0.55	-0.32
	$a_{10,2} = 0.86$	0.06	0.1	0.55	0.03	-0.32	-0.22	-0.33	-0.2
20	$a_{1,1} = 1$	-	-	-	-	-	-	-	-
	$a_{2,1} = 2.28$	0.58	0.31	0.18	-0.03	0.44	-0.1	0.26	0.03
	$a_{3,1} = 2.17$	0.52	0.23	0.06	-0.07	0.37	-0.09	0.19	-0.01
	$a_{4,1} = 2.02$	0.74	0.27	0.19	-0.13	0.39	-0.05	0.27	-0.09
	$a_{5,1} = 0.53$	0.22	0.08	0.06	0.02	0.03	0.11	-0.09	-0.03
	$a_{6,2} = 1$	-	-	-	-	-	-	-	-
	$a_{7,2} = 1.04$	0.1	0.04	0.15	0.12	0.14	0.03	0.17	0.13
	$a_{8,2} = 1.52$	0.18	0.07	0.09	0.09	0.11	-0.09	0.05	0.09
	$a_{9,2} = 0.98$	0.14	0.09	0.15	0.13	0.17	0.07	0.13	0.14
	$a_{10,2} = 0.86$	0.04	0.07	0.04	0.06	0.03	0.09	0.02	0.06
	$a_{11,1} = 2.33$	0.61	0.29	-0.13	-0.17	0.37	-0.06	0	-0.11
	$a_{12,1} = 1.12$	0.3	0.18	0.06	-0.05	0.27	0.11	0.09	-0.03
	$a_{13,1} = 1.26$	0.38	0.13	0.16	0.09	0.36	0.05	0.21	0.12
	$a_{14,1} = 1.01$	0.37	0.2	0.24	0.04	-0.42	-0.16	-0.41	-0.27
	$a_{15,1} = 1.50$	0.34	0.16	0.18	0.03	-0.28	-0.2	-0.31	-0.19
	$a_{16,2} = 0.87$	0.11	0.06	0.11	0.93	0	0.04	0.01	0.11
	$a_{17,2} = 1.79$	0.23	0.07	0.03	0.11	0.29	-0.16	0.03	0.1
	$a_{18,2} = 0.91$	0.13	0.05	0.07	0.08	0.09	0.04	0.01	0.06
	$a_{19,2} = 0.64$	0.29	0.14	0.01	0.07	-0.46	-0.19	-0.42	-0.34
	$a_{20,2} = 0.91$	0.07	0.06	0.19	0.11	0.03	0.04	0.12	0.09

Nota: sesgo de cada parámetro  $\hat{a}$  estimado en cada una de las condiciones. par. = parámetro ; N = tamaño muestral; D. = Dimensiones; N° Ítems = longitud del test, la mitad de los ítems miden el rasgo  $\theta_1$  y la otra mitad el rasgo  $\theta_2$ .

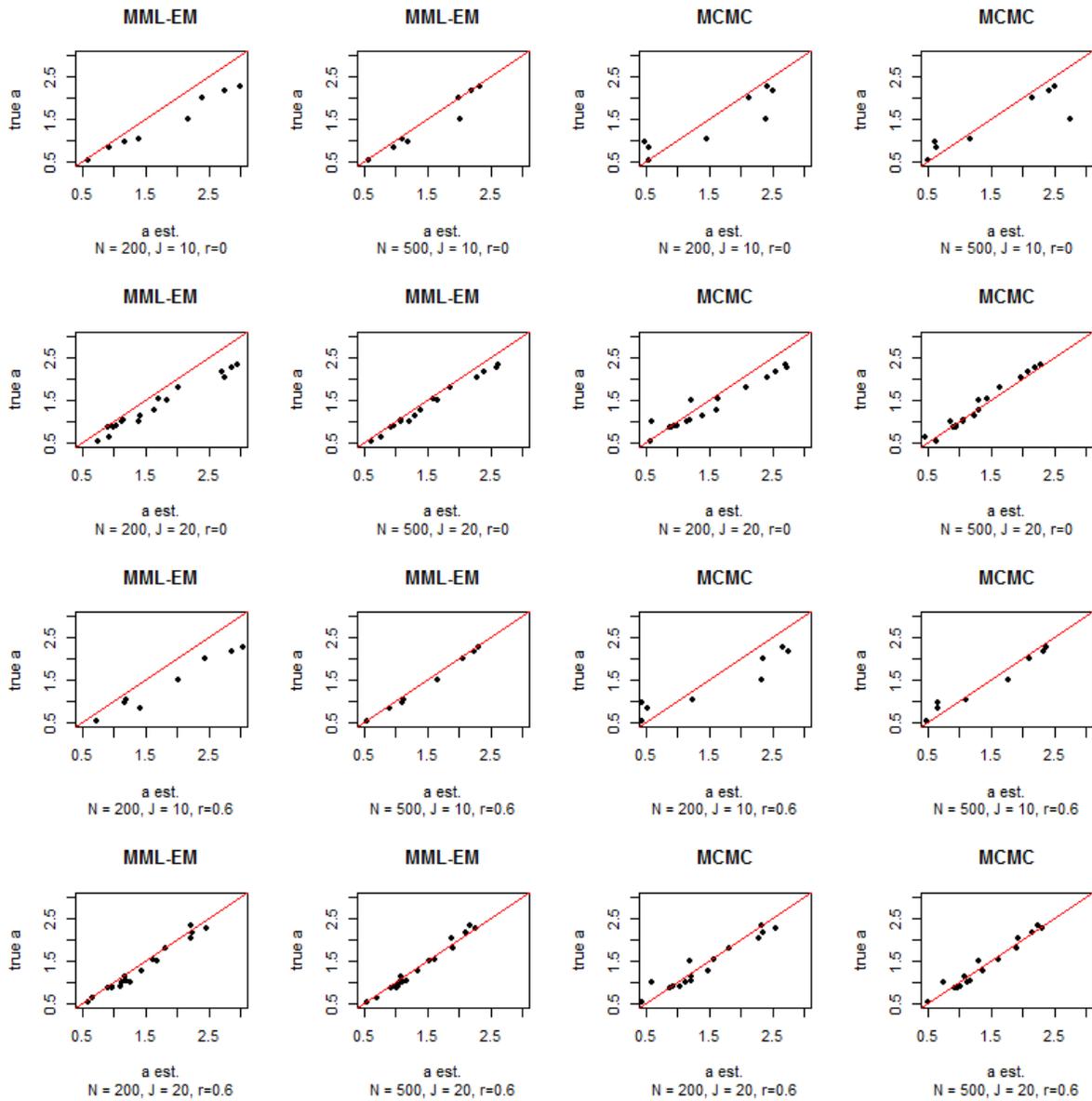


Figura 5. Parámetros  $a$  verdaderos y estimados por cada método en cada condición.

Tabla 4.

*Parámetros  $d$  verdaderos y sesgo de los parámetros  $\hat{d}$  estimados.*

Ítems	par. Verdadero	MML-EM				MCMC			
		D. Independientes ( $\rho=0$ )		D. Relacionadas ( $\rho=0.6$ )		D. Independientes ( $\rho=0$ )		D. Relacionadas ( $\rho=0.6$ )	
		N = 200	N = 500						
10	$d_1 = -1.56$	0.07	0.04	0.07	0.07	0.29	0.12	0.25	0.15
	$d_2 = 0.32$	0.22	0.16	0.13	0.13	0.07	0.14	0.09	0.11
	$d_3 = -0.4$	0.17	0.1	0.17	0.15	0.21	0.13	0.22	0.17
	$d_4 = 0.35$	0.17	0.1	0.05	-0.02	0.05	0.09	0.02	-0.02
	$d_5 = -1.5$	0	0.05	0.16	0.13	0.21	0.13	0.33	0.21
	$d_6 = -0.39$	-0.04	0.02	0.04	0.06	0.01	0.03	0.08	0.07
	$d_7 = -0.21$	-0.08	0	-0.01	0.05	-0.02	0.01	0.02	0.06
	$d_8 = 0.01$	-0.11	-0.01	-0.05	0	-0.07	0	-0.02	0.01
	$d_9 = 2.86$	0.13	0.12	-0.08	-0.05	-0.82	-0.47	-0.89	-0.54
	$d_{10} = -2.31$	-0.11	-0.09	-0.36	0.13	0.48	0.23	0.52	0.38
20	$d1 = -1.56$	0.05	0.07	0.03	0.06	0.25	0.19	0.21	0.13
	$d2 = 0.32$	0.18	0.13	0.19	0.12	0.13	0.08	0.12	0.1
	$d3 = -0.4$	0.17	0.08	0.23	0.17	0.24	0.15	0.26	0.18
	$d4 = 0.35$	0.15	0.13	0.03	-0.03	0.1	0.08	-0.01	-0.04
	$d5 = -1.5$	-0.03	-0.01	0.16	0.13	0.18	0.09	0.33	0.2
	$d6 = -0.39$	0.21	0.12	0.21	0.15	0.27	0.18	0.25	0.17
	$d7 = -0.21$	0.07	0.05	0.07	0.05	0.1	0.07	0.09	0.06
	$d8 = 0.01$	0.05	0.04	0.07	0.06	0.06	0.05	0.06	0.06
	$d9 = 2.86$	0.14	0.08	0.08	-0.05	-0.74	-0.44	-0.8	-0.49
	$d10 = -2.31$	-0.02	0.05	0.07	0.11	0.6	0.43	0.65	0.4
	$d11 = -0.76$	-0.04	0	0.02	0.08	0	0.04	0.07	0.09
	$d12 = -0.05$	-0.08	-0.02	-0.06	0	-0.1	-0.03	-0.07	-0.01
	$d13 = -0.74$	-0.09	0.03	-0.09	0.05	-0.01	0.09	0	0.07
	$d14 = 0.41$	-0.03	0.02	-0.06	-0.06	-0.1	-0.03	-0.1	-0.08
	$d15 = -0.15$	-0.04	0.01	-0.07	-0.02	-0.05	0	-0.05	-0.02
	$d16 = 1.21$	0	0.02	-0.02	0.03	-0.18	-0.1	-0.18	-0.05
	$d17 = 0.78$	-0.05	0.11	-0.09	-0.02	-0.21	-0.1	-0.22	-0.1
	$d18 = 0.91$	0.03	0.01	-0.09	0	-0.11	-0.08	-0.19	-0.05
	$d19 = 3.54$	0.5	0.13	0.07	-0.09	-1.07	-0.68	-1.2	-0.79
	$d20 = 0.94$	-0.08	0	-0.05	-0.07	-0.2	-0.09	-0.17	-0.12

Nota: sesgo de cada parámetro  $\hat{d}$  estimado en cada una de las condiciones. par. = parámetro; N = tamaño muestral; D. = Dimensiones; N° Ítems = longitud del test, mismo parámetro  $d$  para cada dimensión.

Asimismo, el sesgo absoluto medio, el RMSE medio y la correlación entre cada uno de los parámetros de los ítems verdaderos y los parámetros estimados en cada una de las condiciones se ilustran en las tablas 5 y 6.

En general, parece los parámetros de discriminación se estiman con menor sesgo cuando aumenta el tamaño de la muestra y al incrementarse el número de ítems por rasgo latente. Parece a su vez que la estimación de los parámetros  $a$  se realiza de manera más insesgada en las condiciones con dimensiones relacionadas en comparación con las condiciones con factores independientes.

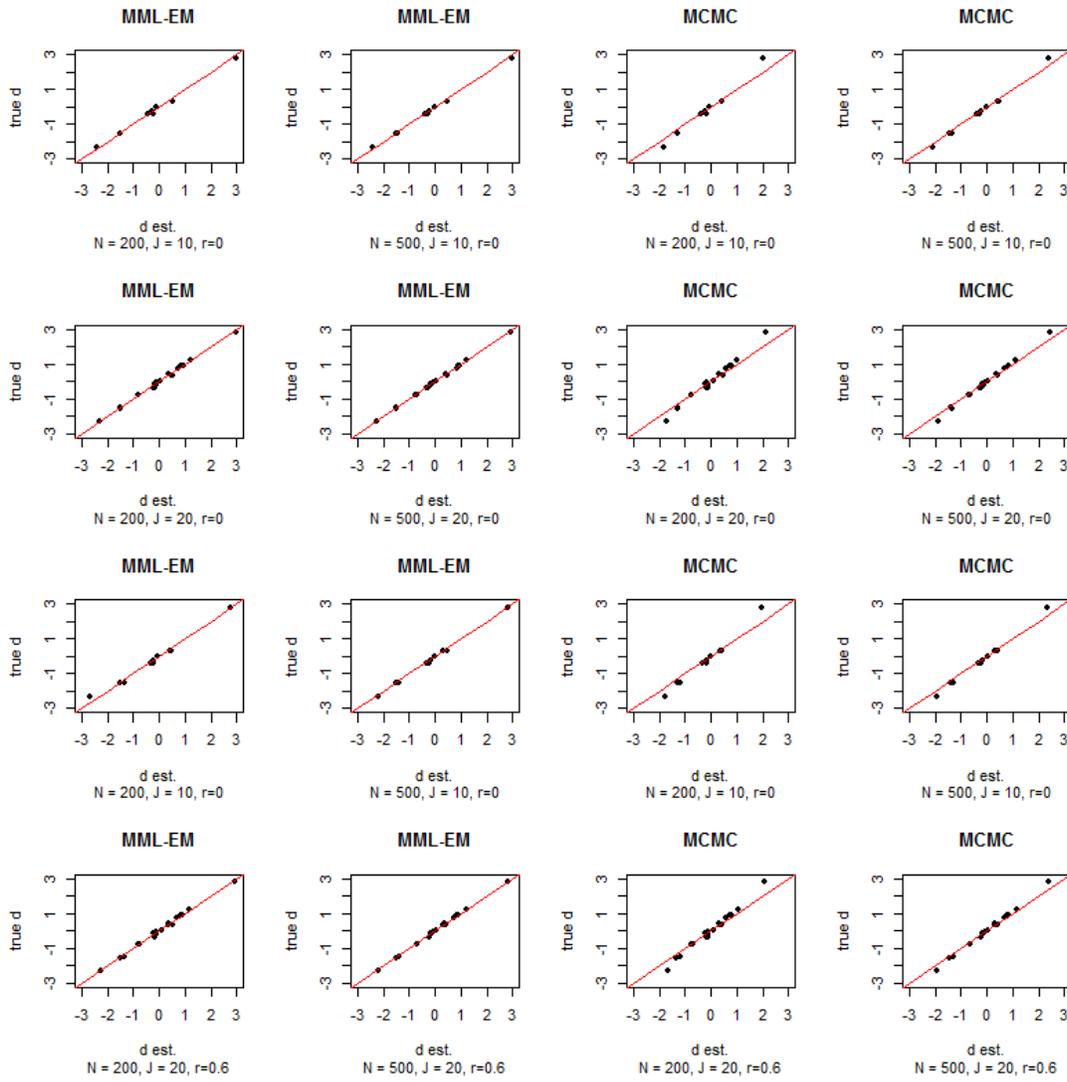


Figura 6. Parámetros  $d$  verdaderos y estimados por cada método en cada

Tabla 5.  
Estadísticos descriptivos para el parámetro  $a$ .

Ítems	N = 200			N = 500		
	sesgo	RMSE	$r$	sesgo	RMSE	$r$
<u>D. Independientes (<math>\rho = 0</math>) – MML-EM</u>						
10	0.37	1.364	.988	0.12	0.507	.968
20	0.30	0.965	.985	0.14	0.455	.996
<u>D. Relacionadas (<math>\rho = 0.6</math>) – MML-EM</u>						
10	0.43	1.675	.984	0.07	0.436	.997
20	0.12	0.557	.988	0.08	0.336	.991
<u>D. Independientes (<math>\rho = 0</math>) – MCMC</u>						
10	0.34	0.722	.899	0.32	0.642	.948
20	0.24	0.561	.963	0.09	0.279	.984
<u>D. Relacionadas (<math>\rho = 0.6</math>) – MCMC</u>						
10	0.40	0.855	.946	0.15	0.467	.983
20	0.16	0.491	.961	0.11	0.337	.971

Nota: RMSE = Root Mean Square Error (raíz del error cuadrático medio).

Si bien los valores del sesgo absoluto medio son semejantes en las condiciones estimadas por cada procedimiento, al emplear el método MCMC los valores del RMSE medio son más bajos, por lo que la dispersión entre los parámetros estimados es menor que al utilizar MML-EM. Por su parte, las correlaciones entre los parámetros  $a$  verdaderos y los parámetros estimados son bastante altas en todas las condiciones, en especial al realizar la estimación mediante MML-EM.

Al estimar el parámetro  $d$  se aprecia que el sesgo absoluto medio es menor para las condiciones estimadas con MML-EM que para aquellas estimadas con MCMC. Por el contrario, el RMSE es en general menor cuando se estiman las condiciones con MCMC, por lo que la dispersión de los estimadores bayesianos es menor que la de los estimadores máximo-verosímiles. Tanto el sesgo como el RMSE toman valores más bajos cuando se asume independencia entre las dimensiones latentes.

Tabla 6.  
*Estadísticos descriptivos para el parámetro  $d$ .*

Ítems	N = 200			N = 500		
	sesgo	RMSE	$r$	sesgo	RMSE	$r$
<u>D. Independientes (<math>\rho = 0</math>) – MML-EM</u>						
10	0.11	0.315	.997	0.07	0.176	.999
20	0.10	0.273	.997	0.05	0.152	.999
<u>D. Relacionadas (<math>\rho = 0.6</math>) – MML-EM</u>						
10	0.11	0.571	.994	0.08	0.356	.999
20	0.09	0.420	.997	0.07	0.343	.999
<u>D. Independientes (<math>\rho = 0</math>) – MCMC</u>						
10	0.22	0.285	.995	0.13	0.187	.997
20	0.23	0.295	.992	0.15	0.192	.998
<u>D. Relacionadas (<math>\rho = 0.6</math>) – MCMC</u>						
10	0.24	0.422	.995	0.17	0.359	.998
20	0.25	0.413	.992	0.16	0.348	.997

Nota: RMSE = Root Mean Square Error (raíz del error cuadrático medio).

Al igual que con las correlaciones del parámetro de discriminación, las correlaciones entre los parámetros  $d$  generados y los estimados son muy altas en ambos procedimientos.



$p < .006$ ,  $1 - \beta = .99$ ,  $\hat{\omega}^2 = .095$ , y la interacción entre ambas,  $F(1) = 13.34$ ,  $p < .001$ ,  $1 - \beta = .953$ ,  $\hat{\omega}^2 = .057$ . La interacción indica que cuando el número de ítems es reducido, la recuperación de los parámetros de discriminación del modelo es más precisa en muestras grandes ( $M = -0.81$ ,  $DT = 0.67$ ) que en muestras más reducidas ( $M = -0.06$ ,  $DT = 0.53$ ),  $F(1) = 27.09$ ,  $p < .001$ ,  $1 - \beta = 1.00$ ,  $\hat{\omega}^2 = .119$ ; mientras que al aumentar la longitud del test, se atenúan diferencias entre las muestras de 200 participantes ( $M = -0.73$ ,  $DT = 0.29$ ) y las de 500 ( $M = -0.88$ ,  $DT = 0.55$ ),  $F(1) = 0.11$ ,  $p = .514$ ,  $1 - \beta = 1.00$ ,  $\hat{\omega}^2 = -.000$ .

Más interesante es el efecto del método de estimación,  $F(1) = 10.62$ ,  $p = .001$ ,  $1 - \beta = .750$ ,  $\hat{\omega}^2 = .045$ , que indica que el RMSE de los parámetros  $a$  estimados a través de MCMC ( $M = -0.83$ ,  $DT = 0.51$ ) es significativamente menor que los estimados mediante MML-EM ( $M = -0.60$ ,  $DT = 0.67$ ).

Asimismo, la recuperación de los parámetros  $d$  del modelo es mejor cuando se asume independencia entre las dimensiones latentes ( $M = -1.64$ ,  $DT = 0.55$ ) que cuando la relación entre ambas es alta y positiva ( $M = -1.09$ ,  $DT = 0.56$ ),  $F(1) = 58.74$ ,  $p < .001$ ,  $1 - \beta = 1.00$ ,  $\hat{\omega}^2 = .194$ . También afecta a la estimación de estos parámetros el tamaño muestral,  $F(1) = 7.85$ ,  $p = .005$ ,  $1 - \beta = .85$ ,  $\hat{\omega}^2 = .027$ , de modo que con mayores tamaños muestrales se consiguen menores RMSE. Tampoco parece que en este caso haya grandes diferencias en la precisión de las estimaciones al emplear un procedimiento u otro,  $F(1) = 0.02$ ,  $p = .893$ ,  $1 - \beta = .073$ ,  $\hat{\omega}^2 = -.000$ .

Los tamaños del efecto son en general bajos si los comparamos con los obtenidos por DeMars (2003), donde las variables con mayor efecto toman valores de  $\hat{\omega}^2$  por encima de 0.15. Salvo el efecto de la relación entre las dimensiones latentes sobre la recuperación del parámetro  $d$ , los tamaños del efecto encontrados son bajos.

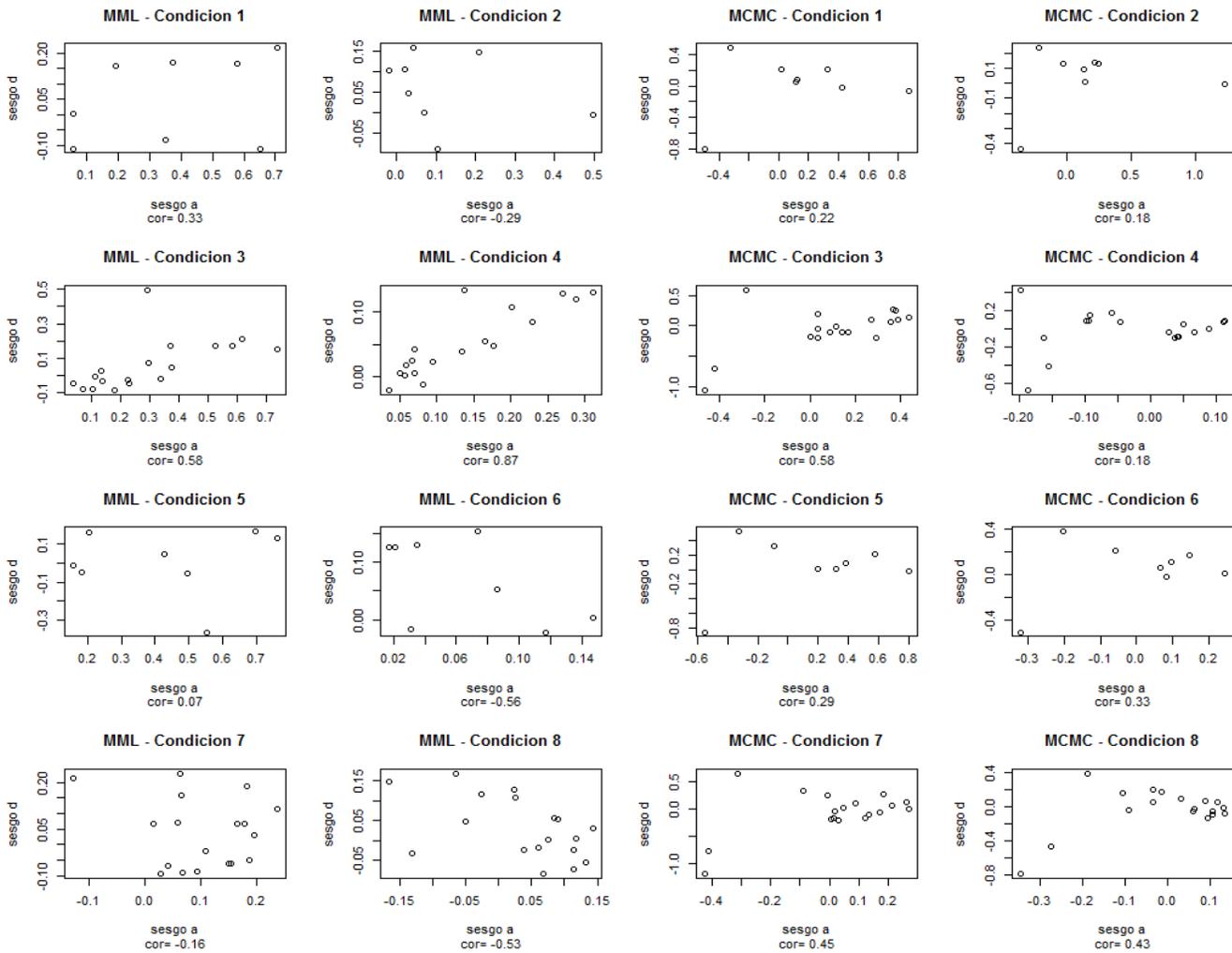


Figura 7. Relación entre los parámetros  $a$  y  $d$  estimados

Finalmente, si se comparan los sesgos de los parámetros  $a$  con los obtenidos para el parámetro  $d$  (ver figura 7) se observan algunos efectos interesantes. En primer lugar, en las condiciones estimadas con MML-EM se observa que la correlación entre el sesgo de los parámetros  $a$  y  $d$  estimados tiende a incrementarse en la misma magnitud al aumentar el tamaño muestral (al pasar de las condiciones pares a las impares), en especial en las condiciones de 20 ítems con dimensiones independientes (condiciones 3 y 4). Parece por tanto que cuanto mayor es el sesgo en el parámetro de discriminación mejor se recuperan las intersecciones. En segundo lugar, en las condiciones estimadas con MCMC, se observa como disminuye la correlación de los parámetros estimados en las condiciones con factores independientes (condiciones de 1 a 4) al aumentar el

tamaño muestral, mientras que en las condiciones con dimensiones relacionadas apenas cambia esta correlación. De hecho, en estas condiciones (de la 5 a la 8), la relación entre los parámetros estimados  $a$  y  $d$  es notablemente mayor que en las condiciones de factores independientes.

Se aprecia también, que al aumentar tanto el número de ítems como el tamaño muestral se reduce en rango de valores del sesgo de los parámetros  $a$ , en ambos métodos, mientras que aumenta ligeramente el sesgo de los parámetros  $d$ , en especial en las estimaciones MCMC.

### *Parámetros de las Personas*

En este estudio, todos los elementos de la matriz  $\Sigma$  también eran también parámetros libres en el modelo. Se estimaron así las varianzas de cada variable latente en todas las condiciones y las covarianzas sólo en aquellas condiciones con dimensiones relacionadas. En la tabla 7 se presentan las estimaciones de estos parámetros.

Tabla 7.  
*Sesgo de los parámetros de la matriz  $\Sigma$ .*

	par. V.	MML-EM				MCMC			
		10 ítems		20 ítems		10 ítems		20 ítems	
		N=200	N= 500	N= 200	N= 500	N= 200	N= 500	N= 200	N= 500
Dimensiones Independientes	$\sigma_{1,1} = 1$	-0.094	-0.09	-0.195	-0.231	-0.32	-0.092	-0.186	-0.353
	$\sigma_{1,2} = 0$	-	-	-	-	-	-	-	-
	$\sigma_{2,2} = 1$	-0.111	-0.093	0.003	-0.022	-0.212	-0.019	0.016	-0.239
Dimensiones Relacionadas	$\sigma_{1,1} = 1$	-0.146	-0.091	-0.056	-0.034	-0.11	-0.061	-0.02	-0.001
	$\sigma_{1,2} = .6$	-0.055	0.009	0.014	0.003	-0.035	0.042	0.02	0.014
	$\sigma_{2,2} = 1$	-0.054	0.01	0.058	-0.065	0.15	0.181	0.131	-0.014

Nota: par. V = parámetro verdadero.

Ambos métodos de estimación recuperan las varianzas razonablemente bien. No obstante, parece que los valores verdaderos se estiman con mayor sesgo al asumirse independencia entre las dimensiones, en especial cuando el número de ítems y el

tamaño muestral son más reducidos. Los dos procedimientos estiman adecuadamente la covarianza entre los factores.

Al igual que los parámetros de los ítems, el sesgo absoluto medio, el RMSE medio y la correlación entre los parámetros generados y los estimados se muestran en la tabla 8. Las puntuaciones estimadas en cada dimensión están fuertemente correlacionadas con las puntuaciones verdaderas de los sujetos simulados, tomando valores ligeramente más altos en aquellas condiciones estimadas mediante MCMC. El sesgo y el RMSE son similares en las condiciones estimadas por cada método, de modo que ambos recuperan con una precisión y una dispersión semejante los parámetros  $\theta_i$ . Además, parece que al aumentar la longitud del test se consiguen estimaciones más adecuadas, en particular en las condiciones con factores independientes.

Para contrastar el efecto concreto de cada variable independiente sobre el conjunto de las estimaciones se realizó otro ANOVA, esta vez sobre el logaritmo neperiano de los RMSE de las estimaciones de los parámetros  $\theta_i$ . Se incluyeron un total

Tabla 8.  
*Estadísticos descriptivos para los parámetros  $\theta$ .*

Ítems	N = 200			N = 500		
	sesgo	RMSE	$r$	sesgo	RMSE	$r$
<u>D. Independientes (<math>\rho = 0</math>) – MML-EM</u>						
10	0.33	0.646	0.885	0.32	0.622	0.886
20	0.23	0.541	0.915	0.23	0.519	0.921
<u>D. Relacionadas (<math>\rho = 0.6</math>) – MML-EM</u>						
10	0.34	0.640	0.948	0.33	0.631	0.946
20	0.23	0.553	0.971	0.24	0.537	0.970
<u>D. Independientes (<math>\rho = 0</math>) – MCMC</u>						
10	0.43	0.655	0.960	0.32	0.632	0.963
20	0.25	0.540	0.976	0.35	0.522	0.983
<u>D. Relacionadas (<math>\rho = 0.6</math>) – MCMC</u>						
10	0.34	0.639	0.949	0.32	0.642	0.950
20	0.24	0.553	0.969	0.24	0.539	0.976

Nota: RMSE = Root Mean Square Error (raíz del error cuadrático medio).

de 11200 observaciones en el análisis: 400 RMSE de las condiciones con tamaños muestrales de 200 sujetos (200 x 2 dimensiones latentes) x 2 longitudes del test, x 2

posibles relaciones entre los factores, x 2 métodos de estimación; más 1000 RMSE de las condiciones de 500 sujetos (500 x 2 dimensiones) x 2 x 2 x 2. Los resultados de este análisis se muestran en la tabla 9.

Si bien los efectos principales de la relación entre los factores,  $F(1)=71.49$ ,  $p<.001$ ,  $1-\beta=1.00$ ,  $\hat{\omega}^2=.001$ , la longitud del test,  $F(1)=145.09$ ,  $p<.001$ ,  $1-\beta=1.00$ ,  $\hat{\omega}^2=.009$ , y el tamaño muestral,  $F(1)=4.57$ ,  $p=.033$ ,  $1-\beta=1.00$ ,  $\hat{\omega}^2=.000$ , resultaron significativos, el tamaño de estos efectos es muy bajo. Sin embargo, la interacción entre la longitud del test y el tamaño de la muestra,  $F(1)=234.42$ ,  $p<.001$ ,  $1-\beta=1.00$ ,  $\hat{\omega}^2=.020$ , tienen un tamaño del efecto bajo. Concretamente, la precisión de las estimaciones de las  $\theta_i$  en aquellas condiciones con un menor número de ítems mejora al aumentarse el tamaño muestral,  $F(1)=351.161$ ,  $p<.001$ ,  $1-\beta=1.00$ ,  $\hat{\omega}^2=.015$ ; mientras que en aquellas condiciones una longitud del test mayor, no hay diferencias entre utilizar una muestra de 200 o de 500 sujetos,  $F(1)=0.97$ ,  $p=.325$ ,  $1-\beta=.116$ ,  $\hat{\omega}^2=-.000$ .

Tabla 9.  
Resultados del ANOVA para los parámetros de las personas.

	gl	SC	MC	F	p	$\hat{\omega}^2$
			<u>Ln[RMSE (theta)]</u>			
Método Estimación	1	0.20	0.20	2.08	0.149	0.000
Relación D. latentes	1	6.89	6.89	71.49	<0.001	0.001
Núm. Ítems	1	13.98	13.98	145.09	<0.001	0.009
Tamaño Muestral	1	0.40	0.40	4.57	0.033	0.000
N. ítems x T. Muestral	1	22.59	22.59	234.418	<0.001	0.020
	22384	2157.3	0.10			

Nota: No se muestran en la tabla las interacciones con  $\hat{\omega}^2 < 0.01$ , aunque sí se incluyeron en el modelo. RMSE = Root Mean Square Error (raíz del error cuadrático medio).

No se encontraron tampoco diferencias entre ambos métodos de estimación en la recuperación de estos parámetros,  $F(1)=2.08$ ,  $p=.149$ ,  $1-\beta=.30$ ,  $\hat{\omega}^2=.000$ , por lo que podría asumirse que ambos métodos estiman las  $\theta_i$  con una precisión similar.

Finalmente, en la *figura 8* se aprecia la relación entre cada parámetro verdadero y cada parámetro estimado en cada una de las condiciones. Lo más reseñable de estos gráficos es que a medida que los parámetros  $\theta_i$  se alejan de los valores centrales, los dos

procedimientos tienden a sobrestimar o infraestimar –en función de si el parámetro verdadero es negativo o positivo, respectivamente- los valores de los mismos. Este efecto se debe a la asunción de que la distribución de estos parámetros es una normal, lo que provoca una regresión a la media de los parámetros más extremos.

En la misma línea de los análisis anteriores, se observa que la precisión con la que cada método recupera las  $\theta_i$  es semejante.

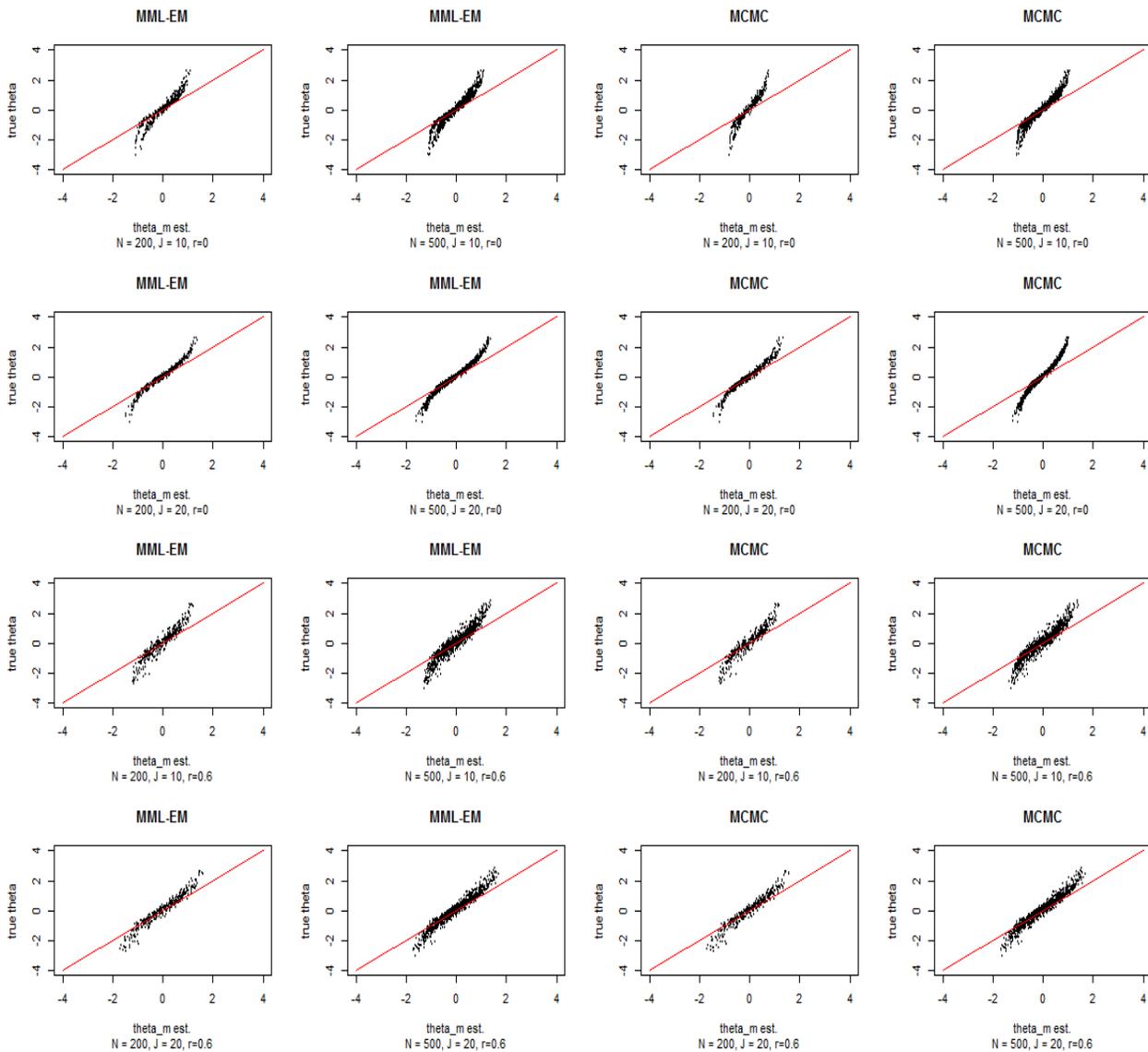


Figura 8. Parámetros  $\theta_{im}$  estimados y verdaderos por cada método en cada condición.

*Tiempo y número de iteraciones*

En la tabla 10 se muestra el tiempo medio de estimación de cada método en cada condición. Las estimaciones mediante MML-EM son notablemente más rápidas que las estimaciones MCMC. Allí donde MML-EM sólo necesita unos cuantos segundos o un par de minutos, las estimaciones MCMC pueden llevar varios minutos o incluso horas. En general, el mayor incremento de tiempo se da al pasar de las muestras de 200 sujetos a las de 500. También tiende a incrementarse el tiempo de computación al añadir ítems, en aunque en menor medida.

Tabla 10.  
*Tiempos de Estimación.*

	<i>N</i>	$\rho$	<i>ítems</i>	<i>Media</i>	<i>DT</i>	<i>Min</i>	<i>Max</i>
MML-EM (segundos)	200	0	10	5.46	1.26	0.88	21.33
	500	0	10	7.70	2.35	4.44	14.08
	200	0	20	41.36	16.21	25.33	101.50
	500	0	20	91.84	10.83	48.39	186.11
	200	.6	10	6.10	1.14	1.00	18.47
	500	.6	10	8.80	2.78	7.42	26.99
	200	.6	20	48.51	21.66	31.68	128.55
	500	.6	20	96.44	32.33	45.04	236.22
MCMC (minutos)	200	0	10	41.35	20.32	15.96	61.08
	500	0	10	39.46	12.56	17.24	121.32
	200	0	20	85.75	21.36	26.32	180.31
	500	0	20	89.67	36.33	37.84	726.35
	200	.6	10	25.34	16.79	12.02	71.16
	500	.6	10	24.46	7.10	15.19	37.78
	200	.6	20	40.21	16.91	25.36	65.21
	500	.6	20	56.67	23.15	38.05	187.89

Por último, en la tabla 11 aparecen el número de iteraciones medias necesarias para que el algoritmo MML-EM convergiera. Se eliminó el único caso en el que el algoritmo necesitó las 500 iteraciones.

Tabla 11.  
*Número de Iteraciones MML-EM.*

<i>n</i>	$\rho$	<i>ítems</i>	<i>Media</i>	<i>DT</i>	<i>Min</i>	<i>Max</i>
200	0	10	146.07	80.48	34	312
500	0	10	142.37	47.5	84	264
200	0	20	143.52	72.99	49	290
500	0	20	171	73.34	61	288
200	.6	10	142.74	73.4	37	357
500	.6	10	132.48	52.89	46	282
200	.6	20	137.8	77.85	31	411
500	.6	20	125.26	41.2	46	237

### Discusión

Este trabajo pone a prueba el desempeño de dos métodos de estimación al recuperar los parámetros de un modelo MIRT bidimensional que asume ítems unidimensionales, o lo que es lo mismo en este caso, un *análisis factorial confirmatorio del ítem* (CIFA) en el que la mitad de los ítems satura en una dimensión y el resto en la otra. Los resultados del estudio de simulación ponen de manifiesto que tanto la estimación por MML-EM como por MCMC recuperan razonablemente bien los parámetros estructurales,  $a$  y  $d$ , y los parámetros incidentales,  $\theta$ , del modelo. No obstante, a pesar de que la precisión alcanzada con ambos procedimientos es similar, parece que la estimación bayesiana recupera mejor los parámetros de discriminación que la estimación máximo-verisímil.

Este resultado es congruente con los resultados obtenidos en otras investigaciones. Beguin y Glas (2001) encontraron que los intervalos de confianza de los parámetros estimados mediante MCMC apenas diferían de los intervalos obtenidos con MML. Sin embargo, parece que la precisión alcanzada con MCMC fue ligeramente mayor a la obtenida con MML (Fox y Glas, 2001; Glas, Weiner y Bradlow, 2000). Por su parte, de la Torre y Patz (2005) hallaron que la estimación MCMC funcionaba

bastante bien al recuperar los parámetros de un modelo logístico multidimensional de dos y de cinco dimensiones. Wollack, Bolt, Cohen y Lee (2002) sugieren asimismo el empleo de los algoritmos MCMC como una alternativa sólida a los procedimientos máximo-versosímiles cuando estos no estén disponibles. Patz y Jünker (1999) recomiendan también la utilización de los métodos MCMC cuando el número de factores sea muy elevado, pues es entonces cuando MML-EM tiende a fallar.

La gran desventaja de las estimaciones bayesianas es el tiempo que lleva realizar un análisis mediante estos procedimientos. Si para un test de 20 ítems y 2 dimensiones el tiempo medio es cercano a una hora, cabe esperar que a medida que aumente la complejidad de los modelos utilizados, más se incremente el tiempo de estimación. Cai (2010a, 2010b) presenta una solución intermedia: el algoritmo *Metropolis-Hastings Robison-Monroe* (MHRM), un método de estimación “híbrido” que incorpora elementos MCMC para proporcionar un estimador máximo-verosímil. Este método consigue reducir sensiblemente los tiempos de estimación y da soluciones relativamente precisas.

Otro de los resultados interesantes de este trabajo es la interacción entre el tamaño muestral y la longitud del test, de modo que cuántos más ítems por dimensión latente se incluyan en el análisis, menor es el impacto del tamaño muestral en la recuperación de los parámetros. Es decir, al añadir más ítems por factor son necesarios menos sujetos en la muestra para obtener una buena estimación de los parámetros del modelo. Edwards (2010), en cambio, encontró que las estimaciones MCMC mejoran sustantivamente cuando las muestras son muy amplias ( $N=2000$ ) en comparación con muestras más “habituales” ( $N=250$ ).

Por último, un resultado sorprendente del estudio de simulación es que la relación entre las dimensiones latentes no afecte a los parámetros  $a$  de discriminación y

que apenas afecte a la recuperación de los parámetros  $\theta$ . Cuando se fija la covarianza de las dimensiones latentes a cero, la recuperación del único parámetro que parece mejorar sustantivamente es la del parámetro  $d$  o intersección. Por tanto, si lo que interesa es ubicar precisamente la dificultad de los ítems (como por ejemplo, en los modelos MIRT de crédito parcial), modelar factores independientes puede resultar ventajoso. En el resto de casos, modelar la relación de factores no parece que tenga un gran efecto en la estimación paramétrica, y conocer la relación entre dos o más variables latentes puede resultar un dato muy útil tanto en la construcción teórica de modelos como en la práctica aplicada.

Dados los resultados de este trabajo, el siguiente paso sería aumentar el número de dimensiones latentes para estudiar entonces la recuperación de parámetros de los dos métodos con modelos más realistas. Para ello, habría que añadir una nueva condición al estudio con más de dos dimensiones, por ejemplo cinco y ocho factores. Charlmers (2012) recomienda el empleo del algoritmo MHRM a partir de las cinco dimensiones en el modelo, pues los puntos de cuadratura del algoritmo EM se reducen a medida que se añaden dimensiones al modelo. Si con dos dimensiones el paquete *mirt* fija a 21 el número de puntos de cuadratura, con 5 factores el autor recomienda utilizar tan sólo 3 puntos de cuadratura para hacer computacionalmente viable la estimación. Por tanto, un nuevo estudio que manipule el número de dimensiones incluidas en el modelo pondría en relieve las ventajas de la estimación bayesiana cuando aumenta la complejidad del modelo. Ya hay estudios en los que se consiguen recuperar relativamente bien los parámetros de modelos complejos en condiciones de alta dimensionalidad (Edwards, 2010; Huang *et al.* 2013). Quedaría por tanto poner a prueba el desempeño de MML-EM en tales condiciones y buscar el punto en la que esta estimación deje de ser viable y se haga necesario el empleo de otros métodos, como MCMC.

Otra de las limitaciones de este estudio son los tamaños del efecto encontrados en los análisis, moderados o bajos la mayoría de ellos, según la regla de Cohen (1988). Sin embargo, Pardo y San Martín (2010) recomiendan comparar los tamaños del efecto de los análisis con los hallados en otros estudios del campo de investigación. Si comparamos los tamaños del efecto por los encontrados en la simulación de DeMars (2003) no parece que los efectos encontrados sean especialmente elevados, con la salvedad de aquel referido a las intersecciones, cuyo valor se acerca a los obtenidos por esta investigadora.

En resumen, este estudio pone de manifiesto que al emplear modelos MIRT relativamente sencillos, tanto MML-EM como MCMC son dos muy buenas alternativas para estimar los parámetros de estos modelos. No obstante, el tiempo de MML-EM es sustantivamente menor al estimar el modelo que el tiempo necesario con MCMC. Esta sería la razón principal –la eficiencia computacional- por la que la utilización de este método de estimación máximo verosímil sería preferible sobre la estimación MCMC al emplear modelos MIRT con un número reducido de dimensiones.

## Referencias

- Abad, F.J., Olea, J., Ponsoda, V. y García, C. (2011). *Medición en Ciencias del Comportamiento y de la Salud*. Madrid: Síntesis.
- Béguin, A. A., & Glas, C. a. W. (2001). MCMC estimation and some model-fit analysis of multidimensional IRT models. *Psychometrika*, *66*(4), 541–561.  
doi:10.1007/BF02296195
- Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. Lord & M. Novick, *Statistical theories of mental test scores*. Reading: Mass. Addison-Wesley.
- Bock, R. D., & Lieberman, M. (1970). Fitting a response model for  $n$  dichotomously scored items. *Psychometrika*, *35*, 179–197.
- Bock, R. D., y Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: An application of the EM algorithm. *Psychometrika*, *46*, 443–459.
- Bock, R. D., Gibbons, R., Schilling, S. G., Muraki, E., Wilson, D. T., y Wood, R. (2002). *TESTFACT 4*. Chicago: Scientific Software International.
- Bock, R. D., Gibbons, R., & Muraki, E. (1988). Full-information item factor analysis. *Applied Psychological Measurement*, *12*, 261–280.
- Bolt, D.M., y Lall, V.F. (2003). Estimation of compensatory and noncompensatory multidimensional IRT models using Markov chain Monte Carlo. *Applied Psychological Measurement*, *27*, 395–414.
- Brooks, S. P., & Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of computational and graphical statistics*, *7*(4), 434-455.
- Cai, L. (2010a). High-dimensional exploratory item factor analysis by a Metropolis–Hastings Robbins–Monro algorithm. *Psychometrika*.

- Cai, L. (2010b). Metropolis–Hastings Robbins–Monro algorithm for confirmatory item factor analysis. *Journal of Educational and Behavioral Statistics*.
- Cai, L., Yang, J. S., & Hansen, M. (2010). Generalized full-information item bifactor analysis. *Psychological Methods, 16*(3), 221–248.  
doi:<http://dx.doi.org/10.1037/a0023350>
- Cattell, R.B. y Cattell, H.E.P. (1995). Personality structure and the new fifth edition of the 16PF. *Educational and Psychological Measurement, 6*, 926-937.
- Chalmers, P. (2012). mirt: A Multidimensional Item Response Theory Package for the R Enviroment. *Journal of Statistical Software, 48*(6), 1-29.
- Cohen, J. (1988). *Statistical power analysis for behavioral sciences*. New York: Academic Press.
- Costa, P.T. y McCrae, R.R. (1992d). *NEO-PI-R. Professional Manual*. Odessa, FL: Psychological Assessment Resources. Inc.
- de la Torre, J., y Douglas, J. (2004). Higher-order latent trait models for cognitive diagnosis. *Psychometrika, 69*, 333-353.
- de la Torre, J. y Patz, R.J. (2005). Making the most of what we have: a practical application of multidimensional item response theory in test scoring. *Journal of Educational and Behavioral Statistics, 30*(3), 295-311.  
doi:<http://dx.doi.org/10.3102/107699860300003295>
- de la Torre, J., y Song, H. (2009). Simultaneous estimation of overall and domain abilities: A higher-order IRT model approach. *Applied Psychological Measurement, 33*, 620-639.
- DeMars, C.E. (2003). Sample Size and the Recovery of Nominal Response Model Item Parameters. *Applied Psychological Measurement, 27*(4), 275-288.
- DiStefano, C. (2002). The impact of categorization with confirmatory factor analysis. *Structural Equation Modeling, 9*, 327–346.

- Eysenck, H.J. y Eysenck, S.B. (1975). *Manual of the Eysenck Personality Questionnaire*. London: Hodder and Stoughton.
- Edwards, M. C. (2010). A Markov Chain Monte Carlo Approach to Confirmatory Item Factor Analysis. *Psychometrika*, 75(3), 474–497. doi:<http://dx.doi.org/10.1007/s11336-010-9161-9>
- Fox, J., y Glas, C. A. (2001). Bayesian estimation of a multilevel IRT model using gibbs sampling. *Psychometrika*, 66(2), 271-288.
- Fox, J.-P. (2005). Multilevel IRT using dichotomous and polytomous response data. *British Journal of Mathematical and Statistical Psychology*, 58(1), 145–172. doi:<http://dx.doi.org/10.1348/000711005X38951>
- Fox, J. P. (2010) Bayesian Item Response Modeling: Theory and Applications. New York: Springer.
- Gelman, A., Carlin, J. B., Stern, H. S. and Rubin, D. B. (1995). *Bayesian Data Analysis*. Chapman and Hall, New York.
- Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4), 457-472.
- Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, 721-741.
- Gibbons, R. D., y Hedeker, D. (1992). Full information item bi-factor analysis. *Psychometrika*, 57,423-436.
- Gilks, W.R., Richardson, S. and Spiegelhalter, D.J., eds. (1996), *Markov Chain Monte Carlo in Practice*. London: Chapman and Hall.

- Glas, C.A.W., Wainer, H., y Bradlow, E.T. (2000). MML and EAP estimates for the testlet response model. En van der Linden, W.J. y W. Glas, C.A. (Eds.), *Computer adaptive testing: Theory and practice* (pp. 271-287). Boston MA: Kluwer-Nijhoff Publishing.
- Hays, W.L. (1995). *Statistics*. New York: Holt, Rinehart & Winston.
- Huang, H.-Y., y Wang, W.-C. (2013). Multilevel Higher-Order Item Response Theory Models. *Educational and Psychological Measurement*, 0013164413509628.  
doi:10.1177/0013164413509628
- Kim, J. S., & Bolt, D. M. (2007). Estimating item response theory models using Markov Chain Monte Carlo methods. *Educational Measurement: Issues and Practice*, 26(4), 38-51.
- Levy, R. (2009). The rise of Markov chain Monte Carlo estimation for psychometric modeling. *Journal of Probability and Statistics*, 2009.
- Little, T. D., Cunningham, W. A. y Sahar, G. (2002). To parcel or not to parcel: exploring the question, weighing the merits. *Structural Equation Modeling*, 9(2), 151-173.
- Lord, F. M., y Novick, M. R. (1968). *Statistical theories of mental test scores*. Reading, MA: Addison-Wesley.
- Lunn, D.J., Thomas, A., Best, N., and Spiegelhalter, D. (2000) WinBUGS -- a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing*, 10:325--337.
- Maris, E. (1999). Estimating multiple classification latent class models. *Psychometrika*, 64, 187-212.
- McDonald, R. P. (1962). Non-linear factor analysis. *Psychometric Monograph* 15.
- McDonald, R. P. (1985). *Factor analysis and related methods*. New Jersey: Lawrence Erlbaum Associates.

- McDonald, R. P. (1999). *Test theory: a unified treatment*. New Jersey: Lawrence Erlbaum Associates.
- Muraki, E. y Carlson, J. E. (1995). Full-information factor analysis for polytomous item response. *Applied Psychological Measurement, 19*, 73-90.
- Muthén, B. O., y Muthén, L. K. (2006). *Mplus: Statistical analysis with latent variables, Version 4.1*. Los Angeles: Muthén y Muthén.
- Ntzoufras, I. (2009). *Bayesian Modeling Using WinBUGS*. Wiley.
- Plummer, M. (2003). JAGS: A Program for Analysis of Bayesian Graphical Models Using Gibbs Sampling, Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003), March 20–22, Vienna, Austria.
- Thomas, A., O'Hara, B., Ligges, U., and Sturtz, S. (2006). Making BUGS Open. *R News 6 (1)*, 12-17.
- Pardo, A. y San Martín, R. (2010). *Análisis de Datos en Ciencias Sociales y de la Salud II*. Madrid: Síntesis.
- 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. doi:10.3102/10769986024002146
- R Core Team (2013). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Viena: Austria.
- Reckase, M. D. (1997). The past and the future of multidimensional item response theory. *Applied Psychological Measurement, 21*, 25-36.
- Reckase, M. D. (2009). *Multidimensional item response theory*. New York: Springer.
- Revuelta, J., Abad, F.J. y Ponsoda, V. (2006). *Modelos politómicos de respuesta al ítem*. Madrid: La muralla.

- Samejima, F. (1969). *Estimation of latent ability using a response pattern of graded scores*. New York: Psychometric Society.
- Schilling, S., & Bock, R. D. (2005). High-dimensional maximum marginal likelihood item factor analysis by adaptive quadrature. *Psychometrika*, *70*(3), 533–555. doi:<http://dx.doi.org/10.1007/s11336-003-1141-x>
- Sheng, Y., y Wikle, C. K. (2007). Comparing Multiunidimensional and Unidimensional Item Response Theory Models. *Educational and Psychological Measurement*, *67*(6), 899–919. doi:10.1177/0013164406296977
- Sheng, Y., y Wikle, C. K. (2008). Bayesian Multidimensional IRT Models With a Hierarchical Structure. *Educational and Psychological Measurement*, *68*(3), 413–430. doi:10.1177/0013164407308512
- Sheng, Y., y Wikle, C. K. (2009). Bayesian IRT models incorporating general and specific abilities. *Behaviormetrika*, *36*(1), 27–48. doi:<http://dx.doi.org/10.2333/bhmk.36.27>
- Stan Development Team. 2014. Stan: A C++ Library for Probability and Sampling, Version 2.2. <http://mc-stan.org>.
- Takane, Y., y de Leeuw, J. (1987). On the relationship between item response theory and factor analysis of discretized variables. *Psychometrika*, *52*, 393–408.
- Wirth, R. J., y Edwards, M. C. (2007). Item factor analysis: Current approaches and future directions. *Psychological Methods*, *12*(1), 58–79. doi:<http://dx.doi.org/10.1037/1082-989X.12.1.58>
- Wollack, J. A., Bolt, D. M., Cohen, A. S., & Lee, Y.-S. (2002). Recovery of Item Parameters in the Nominal Response Model: A Comparison of Marginal Maximum Likelihood Estimation and Markov Chain Monte Carlo Estimation. *Applied Psychological Measurement*, *26*(3), 339–352. doi:10.1177/0146621602026003007

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

## Anexo I – Sintaxis de Openbugs

#Este código corresponde al modelo de Openbugs utilizado para hacer  
#las estimaciones de este trabajo. Puede utilizarse a través del  
#paquete BRugs, guardando el código en un fichero de texto ".txt" a  
#través del comando modelCheck("directorio del fichero"); o bien puede  
#leerse directamente con el programa Openbugs añadiendo los datos en  
#otro fichero (ver código de R, comando bugsData).

# Modelo L2P Multidimensional - Factores Relacionados, Matriz Sigma  
Libre.

```
model{
### Expresión (1): modelo logístico de 2 parámetros multidimensional
  for (i in 1:I){
    for (d in 0:(D-1)){
      P[i,((d*J/D)+1)]<-
exp(theta[i,d+1]+c[((d*J/D)+1)])/(1+exp(theta[i,d+1]+c[((d*J/D)+1)]))
      # Primer ítem de cada dimensión fijado a 1
    }

    for (j in 2:(J/D)){
      for (d in 0:(D-1)){
        P[i,j+d*J/D]<-
exp(a[j+d*J/D]*theta[i,d+1]+c[j+d*J/D])/(1+exp(a[j+d*J/D]*theta[i,d+1]
+c[j+d*J/D]))
        # función de probabilidad para el resto de los ítems
      }
    }

    for(j in 1:J){
      X[i,j] ~ dbern(P[i,j])
    }

    theta[i,1:D] ~ dnorm(mu[],Tau[,])
  }

### Priors

for (j in 1:J){
  c[j] ~ dnorm(0,4)
}

for(j in 2:J/D){
  for(d in 0:(D-1)){
    a[j+d*J/D] ~ dgamma(0.25,0.25)
  }
}

for (d in 1:D) {mu[d]<-0}
```

```

    Tau[1:D,1:D] ~ dwish (W[,],D) # W[,] = matriz identidad, incluir
en los datos
    sigma2[1:D,1:D]<-inverse(Tau[,])
}

# Modelo L2P Multidimensional - Factores Independientes, diag(Sigma)
libre.
### válido sólo para dos dimensiones. Ampliar Tau[,] para añadir más
dim. al modelo.

model{
### Expresión (1): modelo logístico de 2 parámetros multidimensional
  for (i in 1:I){
    for (d in 0:(D-1)){
      P[i,((d*J/D)+1)]<-
exp(theta[i,d+1]+c[((d*J/D)+1)])/(1+exp(theta[i,d+1]+c[((d*J/D)+1)]))
      # Primer ítem de cada dimensión fijado a 1
    }
    for (j in 2:(J/D)){
      for (d in 0:(D-1)){
        P[i,j+d*J/D]<-
exp(a[j+d*J/D]*theta[i,d+1]+c[j+d*J/D])/(1+exp(a[j+d*J/D]*theta[i,d+1]
+c[j+d*J/D]))
        # función de probabilidad para el resto de los ítems
      }
    }
    for(j in 1:J){
      X[i,j] ~ dbern(P[i,j])
    }

    theta[i,1:D] ~ dmnorm(mu[],Tau[,])
  }

### Priors

  for (j in 1:J){
    c[j] ~ dnorm(0,4)
  }

  for(j in 2:J/D){
    for(d in 0:(D-1)){
      a[j+d*J/D] ~ dgamma(0.25,0.25)
    }
  }

  for (d in 1:D) {mu[d]<-0}
  Tau[1,1] ~ dgamma (0.25,0.25)
  Tau[1,2] <- 0
  Tau[2,1] <- 0
  Tau[2,2] ~ dgamma (0.25,0.25)
  sigma2[1:D,1:D]<-inverse(Tau[,])
}

```

## Anexo II – Código de R

```
### Programa para generar las muestras con las que estimar los
parámetros del modelo. Se incluyen a continuación dos scripts: uno
para generar las condiciones con dimensiones relacionadas y otro para
las dimensiones independientes

# TFM - Generación de muestras simuladas para dimensiones relacionadas
rm(list=ls())
setwd("~/TFM/simulaciones/piloto 1.1")
library(BRugs)
library(mvtnorm)
library(mirt)

l2p<-function(a,d,z){
  P<-exp(a*z+d)/(1+exp(a*z+d))
  return(P)
}

l2pm<-function(a,d,z){
  A<-ncol(a)
  I<-nrow(z)
  J<-length(d)
  P<-matrix(0,I,J)
  for(i in 1:I){
    for(j in 1:J){
      P[i,j]<-
round(exp(sum(a[j,]*z[i,])+d[j])/(1+exp(sum(a[j,]*z[i,])+d[j])),2)
    }
  }
  return(P)
}

set.seed(247865)
seed<-sample(1:1000000,100)

R<-1:50
D<-2 #dimensiones
rho<-.6 #correlación entre dimensiones

mu<-rep(0,D) # parámetro mu para la mvnorm
S<-matrix(rho,D,D) #parámetros sigma para la mvnorm
diag(S)<-1
W<-diag(D) # prior matrix para la wishart distribution

# generando los parámetros
I<-500
J<-20
a<-matrix(0,J,D)
a[1,1]<-a[6,2]<-1
h<-round(runif(J-D,0.3,2.5),2)
a[c(2:5,11:15),1]<-h[1:9];a[c(7:10,16:20),2]<-h[10:18];rm(h)
a1<-a[1:10,]
a2<-a[c(1:5,11:15,6:10,16:20),]
```

```

d<-round(-rnorm(J,0,1.5)/apply(a,1,sum),2)
# thetas
z<-rmvnorm(I,mu,S)
# probabilidades
condition<-expand.grid(I=c(200,500),J=c(10,20)) #condiciones: I
simulees, J items
P1<-l2pm(a1,d[1:10],z)
P2<-l2pm(a2,d,z)

write.table(data.frame(a=apply(a1,1,sum),d=d[1:10]),"par10.txt",row.names=F)
write.table(data.frame(a=apply(a2,1,sum),d=d),"par20.txt",row.names=F)
write.table(round(z,2),"thetas_rel.txt",row.names=F)

mcmc.p1<-mcmc.p2<-mcmc.p3<-mcmc.p4<-list() #mcmc parameters
mcmc.t1<-mcmc.t2<-mcmc.t3<-mcmc.t4<-list() #mcmc thetas

for (r in R){
  path<-paste("~/TFM/simulaciones/piloto 1.1/R =",r)
  dir.create(path=path)
  setwd(path)
  set.seed(seed[r])
  for(k in 1:4){
    I<-condition$I[k]
    J<-condition$J[k]
    if(J==10){X<-P1}
    if(J==20){X<-P2}
    if(I==200){X<-X[1:I,]}
    if(I==500){X<-X[1:I,]}
    for(i in 1:I){
      X[i,]<-rbinom(J,1,X[i,])
    }
    name<-paste(paste("J",J,sep=""),paste("I",I,sep=""),sep="")
    write.table(X,paste(name,".txt",sep=""),row.names=F,col.names=F)
    data<-list(I=I,J=J,D=D,X=X,W=W)
    bugsname<-file.path(getwd(),paste(name,"Bugsdata.txt",sep=""))
    bugsData(data,fileName=bugsname,format="fg")
    # BRugs package
    modelCheck("~/TFM/simulaciones/piloto 1.1/model.txt")
    modelData(bugsname)
    modelCompile(4)
    modelGenInits()
    modelUpdate(10000)
    samplesSet(c("a","c","sigma2","theta"))
    modelUpdate(10000)
    if(I==200&J==10){mcmc.p1<-samplesStats(c("a","c","sigma2"));
      mcmc.t1<-samplesStats(c("theta"));
      write.table(mcmc.p1,"mcmc.p1.txt");
      write.table(mcmc.t1,"mcmc.t1.txt")}
    if(I==500&J==10){mcmc.p2<-samplesStats(c("a","c","sigma2"));
      mcmc.t2<-samplesStats(c("theta"));
      write.table(mcmc.p2,"mcmc.p2.txt");
      write.table(mcmc.t2,"mcmc.t2.txt")}
  }
}

```

```

if(I==200&J==20){mcmc.p3<-samplesStats(c("a","c","sigma2"));
                mcmc.t3<-samplesStats(c("theta"));
                write.table(mcmc.p3,"mcmc.p3.txt");
                write.table(mcmc.t3,"mcmc.t3.txt")}
if(I==500&J==20){mcmc.p4<-samplesStats(c("a","c","sigma2"));
                mcmc.t4<-samplesStats(c("theta"));
                write.table(mcmc.p4,"mcmc.p4.txt");
                write.table(mcmc.t4,"mcmc.t4.txt")}

# mirt
f0<-
matrix(0,ncol=D,nrow=J,dimnames=list(NULL,c(paste("Factor",1:D,sep="")
)))
  for(d in 1:D){f0[1:(J/D)+(J/D*(d-1)),d]<-1} #los items
perteneientes a cada f
  cov<-matrix(T,D,D)
  model<-mirt.model(f0,COV=cov)
  sv<-mirt(X,model,itemtype="2PL",pars="values") #prior parameters,
mismo prior para todas las estimaciones
  a1.num<-(sv$parnum[sv$name=="a1"])
  a2.num<-(sv$parnum[sv$name=="a2"])
  sv$est[a1.num[1]]<-sv$est[a2.num[J/D+1]]<-FALSE #válido para 2
dimensiones
  sv$value[a1.num[1]]<-sv$value[a2.num[J/D+1]]<-1 #válido para 2
dimensiones
  sv$est[(nrow(sv)-D):nrow(sv)]<-TRUE # válido para 2 dimensiones
  mirt.sv<-mirt(X,model,pars=sv,SE=TRUE,SE.type="BL")
  if(I==200&J==10){write.table(coef(mirt.sv),"mirt.p1.txt");
write.table(fscores(mirt.sv),"fscores.patterns1.txt",row.names=F)
write.table(fscores(mirt.sv,full.scores=T),"mirt.t1.txt");
write.table(mirt.sv@time,"time1.txt");

  write.table(rbind(mirt.sv@logLik,mirt.sv@AIC,mirt.sv@BIC),"loglik1.txt",
              row.names=c("logLik","AIC","BIC"),col.names=F)}
  if(I==500&J==10){write.table(coef(mirt.sv),"mirt.p2.txt");

write.table(fscores(mirt.sv),"fscores.patterns2.txt",row.names=F);

write.table(fscores(mirt.sv,full.scores=T),"mirt.t2.txt");
              write.table(mirt.sv@time,"time2.txt");

write.table(rbind(mirt.sv@logLik,mirt.sv@AIC,mirt.sv@BIC),"loglik2.txt",
              row.names=c("logLik","AIC","BIC"),col.names=F)}
  if(I==200&J==20){write.table(coef(mirt.sv),"mirt.p3.txt");

write.table(fscores(mirt.sv),"fscores.patterns3.txt",row.names=F);

write.table(fscores(mirt.sv,full.scores=T),"mirt.t3.txt");
              write.table(mirt.sv@time,"time3.txt");

  write.table(rbind(mirt.sv@logLik,mirt.sv@AIC,mirt.sv@BIC),"loglik3.txt",
              row.names=c("logLik","AIC","BIC"),col.names=F)}
  if(I==500&J==20){write.table(coef(mirt.sv),"mirt.p4.txt");

```

```
write.table(fscores(mirt.sv), "fscores.patterns4.txt", row.names=F);  
  
write.table(fscores(mirt.sv, full.scores=T), "mirt.t4.txt");  
        write.table(mirt.sv@time, "time4.txt");  
  
write.table(rbind(mirt.sv@logLik, mirt.sv@AIC, mirt.sv@BIC), "loglik4.txt",  
            ",  
            row.names=c("logLik", "AIC", "BIC"), col.names=F)}  
    }  
}
```

```

# TFM - Generación de muestras simuladas para dimensiones
independientes
rm(list=ls())
setwd("~/TFM/simulaciones/piloto 1.1")
library(BRugs)
library(mvtnorm)
library(mirt)

l2p<-function(a,d,z){
  P<-exp(a*z+d)/(1+exp(a*z+d))
  return(P)

l2pm<-function(a,d,z){
  A<-ncol(a)
  I<-nrow(z)
  J<-length(d)
  P<-matrix(0,I,J)
  for(i in 1:I){
    for(j in 1:J){
      P[i,j]<-
round(exp(sum(a[j,]*z[i,])+d[j])/(1+exp(sum(a[j,]*z[i,])+d[j])),2)
    }
  }
  return(P)
}

set.seed(247865)
seed<-sample(1:1000000,100)

R<-51:100
D<-2 #dimensiones
rho<-0 #correlación entre dimensiones

mu<-rep(0,D) # parámetro mu para la mvnorm
S<-matrix(rho,D,D) #parámetros sigma para la mvnorm
diag(S)<-1

# generando los parámetros
I<-500
J<-20
a<-matrix(0,J,D)
a[1,1]<-a[6,2]<-1
h<-round(runif(J-D,0.3,2.5),2)
a[c(2:5,11:15),1]<-h[1:9];a[c(7:10,16:20),2]<-h[10:18];rm(h)
a1<-a[1:10,]
a2<-a[c(1:5,11:15,6:10,16:20),]
d<-round(-rnorm(J,0,1.5)/apply(a,1,sum),2)
# thetas
z<-rmvnorm(I,mu,S)
# probabilidades
condition<-expand.grid(I=c(200,500),J=c(10,20)) #condiciones: I
simulees, J items
P1<-l2pm(a1,d[1:10],z)

```

```

P2<-l2pm(a2,d,z)

write.table(data.frame(a=apply(a1,1,sum),d=d[1:10]),"par10.txt",row.names=F)
write.table(data.frame(a=apply(a2,1,sum),d=d),"par20.txt",row.names=F)
write.table(round(z,2),"thetas_indep.txt",row.names=F)

mcmc.p1<-mcmc.p2<-mcmc.p3<-mcmc.p4<-list() #mcmc parameters
mcmc.t1<-mcmc.t2<-mcmc.t3<-mcmc.t4<-list() #mcmc thetas

for (r in R){
  path<-paste("~/TFM/simulaciones/piloto 1.1/R =",r)
  dir.create(path=path)
  setwd(path)
  set.seed(seed[r])
  for(k in 1:4){
    I<-condition$I[k]
    J<-condition$J[k]
    if(J==10){X<-P1}
    if(J==20){X<-P2}
    if(I==200){X<-X[1:I,]}
    if(I==500){X<-X[1:I,]}
    for(i in 1:I){
      X[i,]<-rbinom(J,1,X[i,])
    }
    name<-paste(paste("J",J,sep=""),paste("I",I,sep=""),sep="")
    write.table(X,paste(name,".txt",sep=""),row.names=F,col.names=F)
    data<-list(I=I,J=J,D=D,X=X)
    bugsname<-file.path(getwd(),paste(name,"Bugsdata.txt",sep=""))
    bugsData(data,fileName=bugsname,format="fg")
    # BRugs package
    modelCheck("~/TFM/simulaciones/piloto 1.1/model2.txt")
    modelData(bugsname)
    modelCompile(4)
    modelGenInits()
    modelUpdate(10000)
    samplesSet(c("a","c","sigma2","theta"))
    modelUpdate(10000)
    if(I==200&J==10){mcmc.p1<-samplesStats(c("a","c","sigma2"));
      mcmc.t1<-samplesStats(c("theta"));
      write.table(mcmc.p1,"mcmc.p1.txt");
      write.table(mcmc.t1,"mcmc.t1.txt")}
    if(I==500&J==10){mcmc.p2<-samplesStats(c("a","c","sigma2"));
      mcmc.t2<-samplesStats(c("theta"));
      write.table(mcmc.p2,"mcmc.p2.txt");
      write.table(mcmc.t2,"mcmc.t2.txt")}
    if(I==200&J==20){mcmc.p3<-samplesStats(c("a","c","sigma2"));
      mcmc.t3<-samplesStats(c("theta"));
      write.table(mcmc.p3,"mcmc.p3.txt");
      write.table(mcmc.t3,"mcmc.t3.txt")}
    if(I==500&J==20){mcmc.p4<-samplesStats(c("a","c","sigma2"));
      mcmc.t4<-samplesStats(c("theta"));
      write.table(mcmc.p4,"mcmc.p4.txt");

```

```

write.table(mcmc.t4,"mcmc.t4.txt")}

# mirt
f0<-
matrix(0,ncol=D,nrow=J,dimnames=list(NULL,c(paste("Factor",1:D,sep="")
)))
for(d in 1:D){f0[1:(J/D)+(J/D*(d-1)),d]<-1} #los items
pertenecientes a cada f
#cov<-matrix(T,D,D)
model<-mirt.model(f0)
sv<-mirt(X,model,itemtype="2PL",pars="values") #prior parameters,
mismo prior para todas las estimaciones
a1.num<-(sv$parnum[sv$name=="a1"])
a2.num<-(sv$parnum[sv$name=="a2"])
sv$est[a1.num[1]]<-sv$est[a2.num[J/D+1]]<-FALSE #válido para 2
dimensiones
sv$value[a1.num[1]]<-sv$value[a2.num[J/D+1]]<-1 #válido para 2
dimensiones
sv$est[c((nrow(sv)-D),nrow(sv))]<-TRUE # válido para 2 dimensiones
mirt.sv<-mirt(X,model,pars=sv,SE=TRUE,SE.type="BL")
if(I==200&J==10){write.table(coef(mirt.sv),"mirt.pl.txt");

write.table(fscores(mirt.sv),"fscores.patterns1.txt",row.names=F)

write.table(fscores(mirt.sv,full.scores=T),"mirt.t1.txt");
write.table(mirt.sv@time,"time1.txt");

write.table(rbind(mirt.sv@logLik,mirt.sv@AIC,mirt.sv@BIC),"loglik1.txt
",row.names=c("logLik","AIC","BIC"),col.names=F)}
if(I==500&J==10){write.table(coef(mirt.sv),"mirt.p2.txt");

write.table(fscores(mirt.sv),"fscores.patterns2.txt",row.names=F);

write.table(fscores(mirt.sv,full.scores=T),"mirt.t2.txt");
write.table(mirt.sv@time,"time2.txt");

write.table(rbind(mirt.sv@logLik,mirt.sv@AIC,mirt.sv@BIC),"loglik2.txt
",
row.names=c("logLik","AIC","BIC"),col.names=F)}
if(I==200&J==20){write.table(coef(mirt.sv),"mirt.p3.txt");

write.table(fscores(mirt.sv),"fscores.patterns3.txt",row.names=F);

write.table(fscores(mirt.sv,full.scores=T),"mirt.t3.txt");
write.table(mirt.sv@time,"time3.txt");

write.table(rbind(mirt.sv@logLik,mirt.sv@AIC,mirt.sv@BIC),"loglik3.txt
",
row.names=c("logLik","AIC","BIC"),col.names=F)}
if(I==500&J==20){write.table(coef(mirt.sv),"mirt.p4.txt");

write.table(fscores(mirt.sv),"fscores.patterns4.txt",row.names=F);

write.table(fscores(mirt.sv,full.scores=T),"mirt.t4.txt");

```

```
write.table(mirt.sv@time,"time4.txt");

write.table(rbind(mirt.sv@logLik,mirt.sv@AIC,mirt.sv@BIC),"loglik4.txt",
",
row.names=c("logLik","AIC","BIC"),col.names=F)}
}
```