Brown, Anna and Maydeu-Olivares, Alberto (2012) *Fitting a Thurstonian IRT model to forced-choice data using Mplus.* Behavior Research Methods, 44 (4). pp. 1135-1147. ISSN 1554-351X.

Fitting a Thurstonian IRT model to forced-choice data using M*plus*

Anna Brown

University of Cambridge

Alberto Maydeu-Olivares

University of Barcelona

Author Note

Anna Brown, Department of Psychiatry, University of Cambridge, UK. Alberto Maydeu-Olivares, Faculty of Psychology, University of Barcelona, Spain.

Anna Brown was supported by Grant RG63087 from the Isaac Newton Trust, University of Cambridge. Alberto Maydeu-Olivares was supported by an ICREA-Academia Award, grant SGR 2009 74 from the Catalan Government and grants PSI2009-07726 and PR2010-0252 from the Spanish Ministry of Education.

Correspondence concerning this article should be addressed to: Anna Brown, School of Psychology, University of Kent, Canterbury, Kent CT2 7NP, United Kingdom. Emal: A.A.Brown@kent.ac.uk

Abstract

To counter response distortions associated with the use of rating scales (aka Likert scales), items can be presented in comparative fashion, where respondents are asked to rank the items within blocks (forced-choice format). However, classical scoring procedures for these forced-choice designs lead to ipsative data, which presents psychometric challenges well described in the literature. Recently Brown and Maydeu-Olivares (2011a) have introduced a model based on Thurstone's Law of Comparative Judgment that overcomes the problems of ipsative data. Here, we provide a step-by-step tutorial for coding forced-choice responses, specifying a Thurstonian IRT model appropriate for the design used, assessing its fit, and scoring individuals on psychological attributes. Estimation and scoring is performed using M*plus*, and a very straightforward Excel macro is provided that writes full M*plus* input files for any forced-choice design. Armed with these tools, using a forced-choice design is now as easy as using ratings.

Fitting a Thurstonian IRT model to forced-choice data using M*plus*

Typical questionnaire and survey items are presented to respondents one at a time (single-stimulus items), which often leads to indiscriminate endorsement of all desirable items by respondents, resulting in systematic score inflation. *Forced-choice* response formats were designed to reduce such biases by forcing to choose between similarly attractive options. In forced-choice questionnaires items are presented in blocks of two, three, four or more items at a time and respondents are asked to rank the items within each block according to some instruction (for instance, in terms of how well the items describe their behavior, or attitude). Sometimes, the respondents are asked to indicate only the top and the bottom ranks (for instance, select one item that best describes them and one that least describes them).

One special case of forced choice is the so-called multidimensional forced choice (MFC) where each item is assumed to measure only one psychological attribute, and all items within a block measure different attributes. MFC questionnaires are popular in psychological assessment industry because it is believed that this format is more robust against response sets, halo effects and impression management and there is experimental evidence to support this (e.g. Cheung & Chan, 2002; Bartram, 2007; Jackson, Wroblewski & Ashton, 2000; Christiansen, Burns & Montgomery, 2005).

The standard scoring used with forced-choice questionnaires involves adding the inverted rank orders of items within blocks to their respective scales. As a fixed number of points are allocated in every block, the total number of points on the test is the same for every individual (*ipsative* data). In other words, one scale score can be determined from the remaining scales. Ipsativity leads to some highly undesirable consequences, namely:

1) Scores are *relative* rather than absolute; therefore, while meaningful intra-individual interpretation can be made, comparisons between individuals are problematic.
2) *Construct validity* is distorted. Because one scale can be determined from the remaining scales, the scales' correlation matrix has one zero eigenvalue which prevents the use of factor analysis. More importantly, the average scale inter-correlation can be derived exactly from the number of

scales and it must be negative – regardless of the true relationships between the measured attributes (e.g. Clemans, 1966).

3) *Criterion-related validity* is distorted. Due to zero variance of the total score, the correlations between a questionnaire's scales and any external criterion must sum to zero (e.g. Clemans, 1966). Consequently, any positive correlations with the criterion must be compensated by spurious negative correlations, and vice versa.

4) *Reliability estimates* are distorted. Classical reliability coefficients are not appropriate for forced-choice questionnaires because ipsative data violates the assumptions underlying them, such as independence of measurement errors (e.g. Meade, 2004).

Much has been written about problems of ipsative data (for an overview see Brown, 2010; also Baron, 1996), and as the result forced-choice tests have been controversial. These psychometric problems, however, are due to the inappropriateness of classical procedures for scoring MFC items, not to the forced-choice format per se (Brown & Maydeu-Olivares, 2011a). The problem with classical scoring is that it completely disregards the response process that individuals engage in when making forced choices. However, because forced-choice blocks are simply sets of rankings (or partial rankings) existing response models for ranking data can be adapted for modeling and scoring forced-choice questionnaire data.

Drawing on Thurstone's Law of Comparative Judgment (Thurstone, 1927, 1931), Brown and Maydeu-Olivares (2011a) have recently introduced an item response theory (IRT) model capable of modeling responses to any MFC questionnaire (Thurstonian IRT model). Brown (2010) shows that modeling preference decisions in forced-choice questionnaires using this model yields scores on measured attributes that are free from the problems of ipsative data. The Thurstonian IRT model is a multidimensional item response model with some special features that can be straightforwardly estimated using the general modeling software M*plus* (Muthén & Muthén, 1998-2010), which also conveniently estimates trait scores for individuals. The estimation is fast; however, programming these models in

M*plus* is tedious and error-prone except for very small models, as one needs to impose parameter constraints that reflect the within-block patterned relationships among items. However, the model is conceptually so simple that the M*plus* programming can be easily automated. With this paper we provide a very simple Excel macro that writes the M*plus* syntax necessary to fit the IRT model to any MFC questionnaire. Furthermore, we provide a detailed tutorial on how to model different types of MFC questionnaires and how to score respondents on the measured attributes.

The paper is organized as follows. We begin by providing general theory for the Thurstonian IRT model. Thus, we describe how to code responses to forced-choice questionnaires and how to link these responses to the attributes that the questionnaire is intended to measure, building a factor analytic model with binary variables (an IRT model). We describe some special features of these models, as well as the identification constraints necessary to estimate them. We also show how general multidimensional IRT theory can be applied to score individuals. Next, we provide an extended tutorial for modeling specific forced-choice designs using simple numerical examples with simulated data. All the datasets, and M*plus* input files are available for download. In this tutorial, we cover different block sizes (items presented in pairs, triplets, quads) and their common and specific features. We cover both full ranking and partial ranking designs. Partial rankings arise when only top and bottom ranking choices (i.e. 'most' and 'least' choices) are requested to simplify the task of responding to blocks of four or more items. In this case, missing data arises and we provide an example of how to deal with this using multiple imputation in M*plus*.

Thurstonian IRT Model

Coding Forced-choice Responses

Consider a questionnaire consisting of items presented in blocks of n items each. Respondents are asked to rank the items within each block. To code their responses, $\tilde{n} = n(n\text{-}1)/2$ binary outcome (dummy) variables per block are used, one for every pairwise combination of items (Maydeu-Olivares & Böckenholt, 2005). For instance, to code a rank ordering of n = 4 items A, B, C and D, one needs to consider outcomes of $\tilde{n} = 6$ pairwise comparisons: whether A was preferred to B, to C and to D; whether B was preferred to C and to D, and whether C was preferred to D. To reach the ordering {B, A, D, C}, B must be preferred in all pairwise comparisons involving it, and C must be not preferred in any. For each pairwise combination $l = \{i, k\}$, a binary variable y_l is used to indicate the outcome of the comparison:

$$y_l = \begin{cases} 1, & \text{if item } i \text{ is preferred to item } k \\ 0, & \text{if item } k \text{ is preferred to item } i \end{cases}. \tag{1}$$

Then the ordering {B, A, D, C} can be coded using binary outcome variables as follows:

Ranking				Binary Outcomes					
A	B	C	D	{A,B}	{A,C}	{A,D}	{B,C}	{B,D}	{C,D}
2	1	4	3	0	1	1	1	1	0

Sometimes respondents are only asked to report one item that best describes them and one that least describes them. The partial ranking corresponding to our example above would yield one missing outcome – the ordering of items A and D is not known:

Partial ranking				Binary Outcomes					
A	B	C	D	{A,B}	{A,C}	{A,D}	{B,C}	{B,D}	{C,D}
	most	least		0	1	.	1	1	0

Partial ranking format results in missing binary outcome variables whenever the block size is four items or more. These outcomes are missing at random (MAR) because the patterns of missing responses do not depend on the missing outcomes; that is, the outcome of the comparison between items that have not been selected as ‘most’ or ‘least’ is missing not because any particular preference would be more or less likely, but because no preference was recorded. However, the outcome is NOT missing completely at random (MCAR) because the patterns of missing responses can be deduced from the observed choices made in the block. For instance, in the example above it is known from the observed responses (item B selected as ‘most’, and item C as ‘least’) that the comparison between the two remaining items, A and D, will not be recorded, so that the binary outcome {A, D} will be missing. Thus, given the observed most-least choices, the pattern of missing outcomes is known for each individual.

Modeling Preference Responses in Relation to Latent Traits

To relate observed binary outcomes to psychological attributes measured by the questionnaire, we use the notion of item *utility* – an unobserved psychological value placed on the item by a respondent. The utility of item *i* is denoted t_i. According to Thurstone's (1927) Law of Comparative Judgment items' utilities are assumed to be normally distributed across respondents and to determine preferential choices. That is, given any two items, the respondent deterministically chooses the item with highest utility. For computational reasons, it is convenient to express Thurstone's model using differences of utilities. Let y_l^* denote the (unobserved) difference of utilities for the pair of items $l = \{i, k\}$

$$y_l^* = t_i - t_k \,. \tag{2}$$

Then Thurstone's law can be written by relating the observed binary outcome to the unobserved difference of two utilities (we can think of it as a response tendency),

$$y_l = \begin{cases} 1 & \text{if} \quad y_l^* \geq 0 \\ 0 & \text{if} \quad y_l^* < 0 \end{cases} . \tag{3}$$

In multi-trait questionnaires, utilities of items are assumed to be governed by a set of *d* psychological attributes (common factors, or latent traits) according to a linear factor analysis model

$$t_i = \mu_i + \sum_{a=1}^{d} \lambda_{ia}\eta_a + \varepsilon_i \,, \tag{4}$$

or, in matrix form

$$\mathbf{t} = \boldsymbol{\mu} + \boldsymbol{\Lambda\eta} + \boldsymbol{\varepsilon} \,, \tag{5}$$

where $\boldsymbol{\eta} = (\eta_1, \eta_2, ... \eta_d)$ is a vector of common attributes, $\boldsymbol{\Lambda}$ is a matrix of factor loadings, $\boldsymbol{\mu}$ is a vector of item intercepts, and $\boldsymbol{\varepsilon}$ is a vector of unique factors (specification and measurement errors) – assumed to be mutually uncorrelated. We let $\boldsymbol{\Phi} = \text{var}(\boldsymbol{\eta})$ be the factors' covariance matrix (for identification we set all variances equal to one so that it is a correlation matrix), and $\boldsymbol{\Psi}^2 = \text{var}(\boldsymbol{\varepsilon})$ be the diagonal matrix of errors' variances.

Combining (2) and (4) we obtain a factor model that links the preference response tendency to the hypothesized common attributes

$$y_l^* = t_i - t_k = -\gamma_l + \sum_{a=1}^{d} (\lambda_{ia} - \lambda_{ka}) \eta_a + (\varepsilon_i - \varepsilon_k) , \tag{6}$$

where the threshold γ_l replaces the difference of the item intercepts: $\gamma_l = -(\mu_i - \mu_k)$. When items are presented in p blocks of size n, there are $\tilde{n} = n(n\text{-}1)/2$ binary outcomes per block, and the total number of binary outcomes in the questionnaire is $p \times \tilde{n}$. In matrix form, the ($p \times \tilde{n}$) vector of response tendencies $\mathbf{y}^*$ of the binary outcomes $\mathbf{y}$ is written as

$$\mathbf{y}^* = -\boldsymbol{\gamma} + \breve{\boldsymbol{\Lambda}}\boldsymbol{\eta} + \breve{\boldsymbol{\varepsilon}} . \tag{7}$$

Here $\boldsymbol{\gamma}$ is a ($p \times \tilde{n}$) vector of thresholds; $\breve{\boldsymbol{\Lambda}}$ is a ($p \times \tilde{n}$) × d matrix of factor loadings; and $\breve{\boldsymbol{\varepsilon}}$ is a ($p \times \tilde{n}$) vector of errors with covariance matrix $\text{var}(\breve{\boldsymbol{\varepsilon}}) = \breve{\boldsymbol{\Psi}}^2$. The relationships between the matrices $\breve{\boldsymbol{\Lambda}}$ and $\breve{\boldsymbol{\Psi}}^2$ of the Thurstonian IRT model and the matrices $\boldsymbol{\Lambda}$ and $\boldsymbol{\Psi}^2$ of the factor analysis model (5) describing the relationship between the items and the common attributes they measure are given by

$$\breve{\boldsymbol{\Lambda}} = \mathbf{A}\boldsymbol{\Lambda} , \qquad \breve{\boldsymbol{\Psi}}^2 = \mathbf{A}\boldsymbol{\Psi}^2\mathbf{A}' , \tag{8}$$

where $\mathbf{A}$ is a block diagonal matrix. When n = 2, each block in $\mathbf{A}$ is $\begin{pmatrix} 1 & -1 \end{pmatrix}$, whereas when n = 3, and n = 4, they are, respectively,

$$\begin{pmatrix} 1 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & 1 & -1 \end{pmatrix}, \qquad \begin{pmatrix} 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{pmatrix}. \tag{9}$$

Parameters of the Independent-clusters Thurstonian IRT Model

Most confirmatory applications assume that each item measures only one trait, and the factor model underlying the item utilities possesses an *independent-clusters basis* (McDonald, 1999). This factorial simplicity is certainly the aim in typical forced-choice questionnaires, and in what follows, we concentrate on independent-clusters factorial structures. When questionnaire items measure two or more attributes, the general theory in (6) applies. In this case, the IRT model can be estimated in the same fashion as the independent clusters; however, additional identification constraints are needed (see the Model Identification section).When items *i* and *k* measure different attributes, η_a and η_b (i.e., a **multi**dimensional comparison), equation (6) simplifies to

$$y_l^* = -\gamma_l + \left(\lambda_i\eta_a - \lambda_k\eta_b\right) + \left(\varepsilon_i - \varepsilon_k\right). \tag{10}$$

If instead, *i* and *k* measure the same attribute η_a (i.e., a **one**-dimensional comparison), equation (6) becomes

$$y_l^* = -\gamma_l + \left(\lambda_i - \lambda_k\right)\eta_a + \left(\varepsilon_i - \varepsilon_k\right). \tag{11}$$

Thus the Thurstonian IRT model with $p \times \tilde{n}$ binary outcomes contains:

1) $p \times \tilde{n}$ **threshold parameters** γ_l. One threshold $\gamma_l = -\left(\mu_i - \mu_k\right)$ is estimated for each binary outcome (i.e. we do not estimate the original intercepts of utilities).
2) $p \times n$ **factor loading parameters**. These are the factor loadings of utilities. Two factor loadings are estimated for each binary outcome – these relate the response tendency to the two attributes measured by the items making up the pairwise comparison. When the block size is $n = 2$ (i.e., items are presented in pairs), each item is involved in one pairwise comparison only, and therefore each utility's factor loading appears only once in matrix $\breve{\Lambda}$ (for example, see matrix $\breve{\Lambda}$ in (21)). When the block size is $n > 2$, each item is involved in $n - 1$ pairwise comparisons, and therefore each utility's

factor loading occurs more than once (*n* – 1 times) in matrix $\breve{\boldsymbol{\Lambda}}$, forming patterns (for example, see matrices $\breve{\boldsymbol{\Lambda}}$ for a triplet design in (19), and for a quad design in (20)).

3) $p \times n$ **uniqueness parameters** ψ_i^2**.** These are uniquenesses of utilities, and when the block size is *n* = 2 (i.e., items are presented in pairs), the residual variance matrix $\text{var}\left(\breve{\boldsymbol{\varepsilon}}\right) = \breve{\boldsymbol{\Psi}}^2$ is a *p* × *p* diagonal matrix:

$$\breve{\boldsymbol{\Psi}}^2 = \begin{pmatrix} \psi_1^2 + \psi_2^2 & & & \\ 0 & \psi_3^2 + \psi_4^2 & & \\ \vdots & & \ddots & \\ 0 & 0 & & \psi_{2p-1}^2 + \psi_{2p}^2 \end{pmatrix}. \qquad (12)$$

When the block size is *n* > 2, there is shared variance between binary outcomes involving the same item, and $\breve{\boldsymbol{\Psi}}^2$ is a (*p* × *ñ*) × (*p* × *ñ*) block-diagonal matrix, with the following blocks for *n* = 3 and *n* = 4 respectively:

$$\breve{\boldsymbol{\Psi}}_3^2 = \begin{pmatrix} \psi_1^2 + \psi_2^2 & & \\ \psi_1^2 & \psi_1^2 + \psi_3^2 & \\ -\psi_2^2 & \psi_3^2 & \psi_2^2 + \psi_3^2 \end{pmatrix}, \qquad (13)$$

$$\breve{\boldsymbol{\Psi}}_4^2 = \begin{pmatrix} \psi_1^2 + \psi_2^2 & & & & & \\ \psi_1^2 & \psi_1^2 + \psi_3^2 & & & & \\ \psi_1^2 & \psi_1^2 & \psi_1^2 + \psi_4^2 & & & \\ -\psi_2^2 & \psi_3^2 & 0 & \psi_2^2 + \psi_3^2 & & \\ -\psi_2^2 & 0 & \psi_4^2 & \psi_2^2 & \psi_2^2 + \psi_4^2 & \\ 0 & -\psi_3^2 & \psi_4^2 & -\psi_3^2 & \psi_4^2 & \psi_3^2 + \psi_4^2 \end{pmatrix}. \qquad (14)$$

The above special features of matrices $\breve{\boldsymbol{\Lambda}}$ and $\breve{\boldsymbol{\Psi}}^2$ complete the definition of the Thurstonian IRT model.

Model Identification

To identify a Thurstonian IRT model (10) built for MFC items that are designed to measure one trait only (also referred to as multi-unidimensional structure in the IRT literature) one needs to set a metric for the latent traits and item errors. The latent traits' variances are set to one. To set a metric for item errors, for blocks of size $n > 2$ (items are presented in triplets, quads, etc.) it suffices to fix the uniqueness of one item per block. Throughout this paper we use the convention of (arbitrarily) fixing the uniqueness of the first item in each block to one. When the block size is $n = 2$ (i.e. items are presented in pairs), no item uniqueness can be identified. In this case, it is convenient to fix the uniqueness of each binary outcome (which is the sum of two item uniquenesses as can be seen from (12)) to one.

The above constraints are generally sufficient to identify most forced-choice designs. A special case arises when multidimensional pairs ($n = 2$) are used to assess exactly two attributes ($d = 2$). Because this model is essentially an exploratory factor model, additional identification constraints need to be imposed on some factor loadings. This case is discussed in Example 4.

When questionnaire items measure two or more attributes, such as in the general case described by (6), additional constraints may be needed to identify factor loadings, because only their differences can be estimated without constraints. This is similar to the unidimensional model described in (11), where setting one factor loading is necessary to identify the model (Maydeu-Olivares & Brown, 2010).

Non-identified models may occasionally arise when item factor loadings within the same block are equal, or indistinguishable from the empirical data. This might happen in designs where positively keyed items measure a small number of attributes, or the attributes are positively correlated, so that the item parameters are more difficult to estimate accurately (Brown & Maydeu-Olivares, 2011a). When the factor loadings λ_i and λ_k are equal (say, they equal λ), the difference of utilities in (10) is described by

$$y_l^* = -\gamma_l + \lambda\left(\eta_a - \eta_b\right) + \left(\varepsilon_i - \varepsilon_k\right). \qquad (15)$$

In this case, the data is sufficiently described by d–1 *differences* between each attribute and, say, the last attribute η_d. Indeed, for any pair of attributes η_a and η_b, their difference $\eta_a - \eta_b$ can be written as $(\eta_a - \eta_d) - (\eta_b - \eta_d)$. The factor space is therefore reduced and additional constraints are needed to identify the model. In practice, it may not be easy to spot such empirical under-identification because no warning of a non-identified model may be given by M*plus*. The researcher needs to examine the program output very carefully to ensure that everything is as expected. Typical signs of the described special case are that estimated factor loadings for one of the factors are close to zero, standard errors of correlation estimates between that factor and other factors are large, and factor correlations are not as expected (usually too high). In some cases, M*plus* might give a warning in the output that ‘the latent variable covariance matrix (Psi) is not positive definite’, and indicate which factor presents a problem. To remedy this situation, it usually suffices to constrain the factor loadings within each block to be equal (without setting their values), and setting just one correlation between the latent traits to its expected value (for instance, to a value predicted by substantive psychological theory).

Parameter Estimation and Goodness-of-fit Testing Using M*plus*

After the choices are coded as described above, a multi-unidimensonal model (10) or the unidimensional model (11) is fitted to the differences of utilities y_l^*. However, the difference variables y_l^* are not observed, only their dichotomizations y_l using the threshold process (3) are observed. Hence, a factor model for binary data (the IRT model) is fitted to the binary outcome variables. All that is needed is a program capable of estimating such a model. The program M*plus* (Muthén & Muthén, 1998-2010) conveniently implements all the necessary features.

The presence of correlated errors, along with the large number of latent traits typically measured by forced-choice questionnaires precludes the estimation of the model by full information maximum likelihood (Bock & Aitkin, 1981). However, the model can be straightforwardly estimated using limited information methods. Unweighted least squares (ULS) or diagonally weighted least squares (DWLS) can be used to this end, and the difference between the two is negligible (Forero, Maydeu-Olivares & Gallardo-Pujol, 2009). When estimating models with discrete dependent variables, M*plus* offers two choices of parameterization, unstandardized and standardized parameters, referred to as ‘theta’ and ‘delta’ respectively. The Thurstonian IRT model is estimated as a factor analysis for binary data using the 'theta' parameterization with the additional constraints on $\breve{\mathbf{\Lambda}}$ and $\breve{\mathbf{\Psi}}^2$ described above. Because contrast matrices **A** are not of full rank (Maydeu-Olivares & Böckenholt, 2005), the matrix of residual variances and covariances $\breve{\mathbf{\Psi}}^2 = \mathbf{A}\mathbf{\Psi}^2\mathbf{A}'$ is also not of full rank. This is by design, and therefore for all forced-choice models M*plus* will give a warning that 'the residual covariance matrix (theta) is not positive definite'.

The goodness of fit of the model to the tetrachoric correlations is tested by M*plus*. The program provides mean or mean and variance Satorra-Bentler (1994) adjustments to the ULS/DWLS fit functions. Mean and variance adjustments provide more accurate *p*-values at the expense of more computations. The mean and variance adjustment for the ULS estimation is denoted as 'estimator' ULSMV in M*plus*, and it is denoted WLSMV for the DWLS estimation. All models presented in this article are estimated with M*plus* using ULS with mean and variance corrected Satorra–Bentler goodness-of-fit tests (ULSMV).

With this article, we supply an Excel macro that automates writing the full code so that all the necessary parameter constraints are specified. Moreover, the Excel macro takes care of specifying the estimator and parameterization.

When the number of items per block is $n > 2$, a correction to degrees of freedom is needed when testing model fit. This is because for each block there are $r = n(n-1)(n-2)/6$ redundancies among the thresholds and tetrachoric correlations estimated from the binary outcome variables (Maydeu-Olivares, 1999). With *p* ranking blocks in the questionnaire, the number of redundancies is $p \times r$. Thus, when n > 2, one needs to subtract $p \times r$ from the degrees of freedom given by M*plus* to obtain the correct *p* value for the test of exact fit. Goodness-of-fit indices involving degrees of freedom in their formula, such as the root mean square error of approximation (RMSEA)

$$RMSEA = \sqrt{\frac{\chi^2 - df}{df \times (N-1)}}\,, \qquad (16)$$

also need to be recomputed using the correct number of degrees of freedom. When *n* = 2, no degrees of freedom adjustment is needed; the *p* value and RMSEA printed by the program are correct.

Estimation of Individuals' Scores

The item characteristic function (ICF) of the binary outcome variable y_l described, which is the result of comparing item *i* measuring trait η_a and item *k* measuring trait η_b, is given by

$$\Pr\left(y_l = 1 \middle| \eta_a, \eta_b\right) = \Phi\left(\frac{-\gamma_l + \lambda_i \eta_a - \lambda_k \eta_b}{\sqrt{\psi_i^2 + \psi_k^2}}\right). \tag{17}$$

In this function, γ_l is the threshold for binary outcome, λ_i and λ_k are the items' factor loadings, and ψ_i^2 and ψ_k^2 are the items' uniquenesses. Therefore, the Thurstonian IRT model can be seen as an extension of the normal ogive IRT model (Lord, 1952) to situations where items are presented in blocks and the underlying structure is multidimensional. A special feature of this model is that, when block size is *n* >2, the item characteristic functions are not independent (local independence conditional on the latent traits does not hold). Rather, there are patterned covariances among the binary outcomes' residuals as shown in (13) and (14).

After the model parameters have been estimated, respondents' attributes can be estimated using a Bayes modal procedure (*maximum a posteriori*, or MAP estimator)

$$F(\boldsymbol{\eta}) = \frac{1}{2}\boldsymbol{\eta}'\boldsymbol{\Phi}^{-1}\boldsymbol{\eta} - \sum_l \ln\left[\Pr\left(y_l = 1 \middle| \boldsymbol{\eta}\right)^{y_l}\left(1 - \Pr\left(y_l = 1 \middle| \boldsymbol{\eta}\right)\right)^{1-y_l}\right] \tag{18}$$

and this is conveniently implemented in M*plus* as an option within the estimation process (Muthén, 1998-2004). When using (18), M*plus* makes the simplifying assumption that local independence holds. The use of this simplification for scoring individuals has little impact on the accuracy of the estimates (Maydeu-Olivares & Brown, 2010).

Tutorial on Writing M*plus* Code with the Excel Macro

Despite the fact that the factorial models (10) and (11) underlying forced-choice comparisons are simple, the programming is complicated by the fact that factor loading and uniqueness for the differences $\mathbf{y}^*$ must be expressed as linear functions of the factor loadings and uniquenesses of the items. There are constraints on the parameter matrices $\breve{\mathbf{\Lambda}}$ and $\breve{\mathbf{\Psi}}^2$, which depend on block size, and writing them out is tedious and error prone. In subsequent sections we provide details on how to estimate the model for a set of examples (using blocks of different size and different numbers of common attributes, etc.) using the supplied Excel macro that writes the full M*plus* syntax.

Coding the Data

M*plus* expects the forced-choice responses to be coded using binary outcomes (dummy variables), as described in this paper; one line per individual. If, however, the forced-choice data have been recorded using rank orders of items within each block, or reversed rank orders as is often the case with already "ipsative scored" items, the responses should be recoded as binary outcomes of pairwise comparisons. Recall that this coding requires each ranking block of size n to be presented as $\tilde{n} = n(n\text{-}1)/2$ pairwise comparisons $\{i, k\}$, each of which takes value 1 if i was preferred to k, and 0 otherwise. This recoding can be easily performed using standard statistical software prior to modeling with M*plus*. Alternatively, DEFINE commands can be used to recode the data within M*plus*. For rank-orderings, binary outcomes of all pairwise combinations of n items are computed as 'i1i2 = i2-i1;' (for ipsative item scores, we use 'i1i2 = i1-i2;'), and then all outcomes are cut as binary variables using 'CUT i1i2 i1i3 … (0);'.

For incomplete rankings, preferences between all items not selected as 'most' or 'least' in blocks of size $n \geq 4$ should be coded as missing data, using conditional statements, for example: 'IF (i2 GT i1) THEN i1i2=1; IF (i2 LT i1) THEN i1i2=0; IF (i2 EQ i1) THEN i1i2=_MISSING;'. In addition, when missing data is present, the missing responses have to be imputed prior to model estimation. This is described in Example 2.

Writing Model Syntax

To aid programming of Thurstonian IRT models, we created an Excel macro that can be downloaded from the journal's website. Excel was chosen because it is widely available, and because it enables simple 'copying and pasting' of questionnaire keys, correlation matrices etc. straight into provided cells. At **Step 1**, the macro just requires as input the name of the data file containing the binary outcomes (the data file may contain additional variables), the name of a file to save the respondents scores (this is optional), the number of forced-choice blocks in the questionnaire and the block size. At **Step 2**, the user is required to enter the number of attributes measured by the questionnaire, and also gives a table for inserting the questionnaire "key". The "key" is simply a numbered list of all questionnaire items, and the user has to indicate which attribute (referred to by its number) each item measures. The macro also has an option to indicate any *negatively keyed* items. These are items designed to represent low attribute scores, such as "I keep in the background" to indicate Extraversion. This information is optional and is only used for assigning better (negative) starting values for factor loading parameters. Finally, **Step 3** (also optional) enables the user to provide starting values for the attribute correlation matrix. With this information, the Excel macro creates the full M*plus* syntax, which can be viewed immediately in Excel, and also copied to a ready-to-execute M*plus* input.

Numerical Examples

Below we present some numerical examples using simulated data. The examples have been designed for illustration only and are necessarily very short. Synthetic data, available for download together with Mplus input files, was used to better illustrate the behavior of the model. As a general foreword for the following examples, we remind the reader that designing forced-choice measures with given block size requires careful consideration of several factors – such as keyed direction of items, number of measured attributes, and correlations between the attributes (Brown & Maydeu-Olivares, 2011a).In examples below all these factors have been balanced to create very short but fully working "fragments" of forced-choice tests. Such short questionnaires in practice would necessarily yield latent trait estimates with high measurement error. Therefore, these examples should only be used as a guide for modeling longer questionnaires. Examples of applications with real questionnaire data are given in the Concluding Remarks section.

Example 1: Block Size n = 3, Full Ranking Response Format. Consider a very simple multidimensional forced-choice design using p = 4 blocks of n = 3 items (triplets), measuring d = 3 common attributes. For simplicity, let the first item in each block measure the first common attribute, the second item measure the second attribute, and the third item measure the third attribute, therefore each attribute is measured by 4 items. We assume that each item measures a single trait and that the traits are possibly correlated (their correlation matrix is **Φ)**. The data is coded using $p \times \tilde{n} = 4 \times 3 = 12$ binary outcomes in total.

According to this forced-choice design, the item utilities' loading matrix **Λ** in equation (4) and the pairwise outcomes' loading matrix $\breve{\boldsymbol{\Lambda}}$ in equation (7) are:

$$\boldsymbol{\Lambda} = \left(\begin{array}{ccc} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \\ \hdashline \lambda_4 & 0 & 0 \\ 0 & \lambda_5 & 0 \\ 0 & 0 & \lambda_6 \\ \hdashline \lambda_7 & 0 & 0 \\ 0 & \lambda_8 & 0 \\ 0 & 0 & \lambda_9 \\ \hdashline \lambda_{10} & 0 & 0 \\ 0 & \lambda_{11} & 0 \\ 0 & 0 & \lambda_{12} \end{array}\right), \qquad \breve{\boldsymbol{\Lambda}} = \left(\begin{array}{ccc} \lambda_1 & -\lambda_2 & 0 \\ \lambda_1 & 0 & -\lambda_3 \\ 0 & \lambda_2 & -\lambda_3 \\ \hdashline \lambda_4 & -\lambda_5 & 0 \\ \lambda_4 & 0 & -\lambda_6 \\ 0 & \lambda_5 & -\lambda_6 \\ \hdashline \lambda_7 & -\lambda_8 & 0 \\ \lambda_7 & 0 & -\lambda_9 \\ 0 & \lambda_8 & -\lambda_9 \\ \hdashline \lambda_{10} & -\lambda_{11} & 0 \\ \lambda_{10} & 0 & -\lambda_{12} \\ 0 & \lambda_{11} & -\lambda_{12} \end{array}\right). \qquad (19)$$

As can be seen, the loading matrix $\breve{\boldsymbol{\Lambda}}$ is patterned, with each utility loading appearing exactly twice. The fact that loadings related to comparisons involving the same items are the same (may differ in sign) need to be written out in M*plus* using the MODEL CONSTRAINT command (automatically written by the Excel macro).

The item residual matrix is $\mathbf{\Psi}^2 = diag\left(\psi_1^2, \ldots, \psi_{12}^2\right)$, and the pairwise outcomes' residual matrix $\breve{\mathbf{\Psi}}^2$ is block-diagonal with elements $\breve{\mathbf{\Psi}}_3^2$ as described in (13). The other model parameters of the Thurstonian IRT model are the factors correlation matrix **Φ**, and a set of *p* × *ñ* thresholds $\boldsymbol{\gamma}$. To identify the model, we just need to set trait variances to one, and set the first uniqueness within each block to one.

To illustrate the discussion we generated responses from *N* = 2000 individuals using the parameter values shown in Table 1. Some factor loadings shown in Table 1 are larger than unity. This is because these are unstandardized factor loadings. The data was simulated by generating latent traits $\boldsymbol{\eta}$ with mean zero and correlation matrix **Φ**, errors $\breve{\boldsymbol{\varepsilon}}$ with mean zero and covariance matrix $\breve{\mathbf{\Psi}}^2$ and computing $\mathbf{y}^* = -\boldsymbol{\gamma} + \breve{\mathbf{\Lambda}}\boldsymbol{\eta} + \breve{\boldsymbol{\varepsilon}}$. These difference values where then dichotomized at zero as per (3). The resulting responses are provided in the file '*triplets.dat*', which consists of 2000 rows and 12 columns, one for each binary outcome variable.

————————————————————

Insert Table 1 about here

————————————————————

To create M*plus* syntax to test this simple model with the supplied data, one can use the Excel macro. One would need to specify the data file ('*triplets.dat*'), the block size (3), the number of blocks (4), the number of attributes measured (3), and supply the questionnaire key which in this example will look as follows: (1, 2, 3, 1x, 2, 3, 1, 2, 3x, 1, 2x, 3). The numbers indicate which trait is measured by each item, and 'x' indicates that the item is negatively keyed. The latter is optional, as it is only used to supply better (negative) starting values for factor loading parameters. Also, starting values for correlations between the attributes can optionally be given. Once input is complete, the syntax written by the Excel macro can be saved as an M*plus* input file, and executed making sure that the file containing the data is located in the same directory as the M*plus* input file. Our syntax '*triplets.inp*' can be found in the supplementary materials; it is also given in Appendix A.

After completing the estimation of the supplied dataset, M*plus* yields a chi-square test of χ^2 = 30.21 on 43 degrees of freedom. However, each triplet has $r = n(n – 1)(n – 2)/6 = 1$ redundancy and there are 4 redundancies in total, so that the correct number of degrees of freedom is *df* = 39, leading to a *p*-value *p* = 0.84. The RMSEA computed using formula (16) with the correct number of degrees of freedom in this case corresponds to the value reported by the program (zero) because the chi-square value is smaller than *df*. The estimated item parameters are reported in Table 1, along with their standard errors. We can see in this table that we are able to recover the true parameter values reasonably well. The reader must be warned, however, that an extremely short questionnaire represented by this small model would not be capable of estimating persons' scores with sufficient precision. In practical applications, many more items per trait are generally required for reliable score estimation.

Example 2: Block Size n = 4, Full Ranking and 'Most-least' Response Formats. When block size, *n*, is larger than 3, no new statistical theory is involved. Bear in mind, however, that if we wish for each item within a block to measure a different trait, the number of traits measured by the questionnaire, *d*, must be equal or larger than block size. In the present example we use *p* = 3 quads (blocks of *n* = 4 items) to measure *d* = 4 traits. Hence, each trait is measured by only 3 items. Specifically, trait 1 is measured by items 1, 5, 9; trait 2 is measured by items 2, 6, 10; trait 3 is measured by items 3, 7, 11; and trait 4 is measured by items 4, 8, 12. We provide in Table 2 a set of true parameter values for this example.

When items are presented in quads, 6 binary outcomes are needed to code the responses to each quad; hence, $p \times \tilde{n} = 3 \times 6 = 18$ binary outcomes are needed in total. The utilities' factor loadings matrix $\boldsymbol{\Lambda}$ and the pairs' loading matrix $\breve{\boldsymbol{\Lambda}}$ are:

$$\boldsymbol{\Lambda}=\left(\begin{array}{cccc}\lambda_1 & 0 & 0 & 0\\ 0 & \lambda_2 & 0 & 0\\ 0 & 0 & \lambda_3 & 0\\ 0 & 0 & 0 & \lambda_4\\ \hdashline \lambda_5 & 0 & 0 & 0\\ 0 & \lambda_6 & 0 & 0\\ 0 & 0 & \lambda_7 & 0\\ 0 & 0 & 0 & \lambda_8\\ \hdashline \lambda_9 & 0 & 0 & 0\\ 0 & \lambda_{10} & 0 & 0\\ 0 & 0 & \lambda_{11} & 0\\ 0 & 0 & 0 & \lambda_{12}\end{array}\right),\qquad \breve{\boldsymbol{\Lambda}}=\left(\begin{array}{cccc}\lambda_1 & -\lambda_2 & 0 & 0\\ \lambda_1 & 0 & -\lambda_3 & 0\\ \lambda_1 & 0 & 0 & -\lambda_4\\ 0 & \lambda_2 & -\lambda_3 & 0\\ 0 & \lambda_2 & 0 & -\lambda_4\\ 0 & 0 & \lambda_3 & -\lambda_4\\ \hdashline \vdots & \vdots & \vdots & \vdots\\ \hdashline \lambda_9 & -\lambda_{10} & 0 & 0\\ \lambda_9 & 0 & -\lambda_{11} & 0\\ \lambda_9 & 0 & 0 & -\lambda_{12}\\ 0 & \lambda_{10} & -\lambda_{11} & 0\\ 0 & \lambda_{10} & 0 & -\lambda_{12}\\ 0 & 0 & \lambda_{11} & -\lambda_{12}\end{array}\right). \tag{20}$$

As can be seen, each utility loading appears exactly three times in the pairs' loading matrix $\breve{\boldsymbol{\Lambda}}$. The item residual matrix is $\boldsymbol{\Psi}^2 = diag\left(\psi_1^2,\ldots,\psi_{12}^2\right)$, and the pairwise outcomes' residual matrix $\breve{\boldsymbol{\Psi}}^2$ is block-diagonal with elements $\breve{\boldsymbol{\Psi}}_4^2$ as shown in (14). In addition to the factor loadings and uniquenesses, the model implies estimating the factor correlation matrix **Φ**, and a set of $p \times \tilde{n}$ thresholds **γ**. Again, the model is identified simply by setting trait variances to one, and setting the first item uniqueness in each block to one.

The purpose of this example is to discuss estimation when the 'most-least' response format is used with ranking blocks of size $n > 3$. In this case, not all binary outcomes are observed, and the missing data is MAR (missing at random), but not MCAR (missing completely at random). Asparauhov & Muthén (2010a) illustrate the deficiencies of the least squares estimation under the MAR condition and show that multiple imputation approach is effective in addressing these problems. We will use the multiple imputation facility available in M*plus* when estimating the IRT model for the 'most-least' data.

The file '*quads_most_least.dat*' contains a simulated sample of 2000 respondents providing 'most-least' partial rankings. Except for the missing data, the responses are equal to those in the file '*quads_full_ranking.dat*', which is given for comparison. Both datasets were generated by dichotomizing difference variables $\mathbf{y}^* = -\boldsymbol{\gamma} + \breve{\boldsymbol{\Lambda}}\boldsymbol{\eta} + \breve{\boldsymbol{\varepsilon}}$, computed using the true model parameters. In the most-least data, the binary comparison involving the two items not selected as ‘most-like-me’ or ‘least-like-me’ was set as missing.

The file ‘*quads_full_ranking.inp*’, which can be readily generated with the Excel macro, contains the M*plus* syntax for estimating the full ranking data ‘*quads_full_ranking.dat*’. To generate this syntax, one has to specify the block size (4), the number of blocks (3), the number of attributes measured (4), and supply the questionnaire key which in this example will look as follows: (1, 2x, 3, 4, 1x, 2, 3, 4, 1, 2, 3x, 4). The numbers indicate which trait is measured by each item, and ‘x’ indicates which items are negatively keyed in relation to the measured trait.

For the full rankings, M*plus* yields a chi-square test of χ^2 = 112.20 on 126 degrees of freedom. However, each quad has $r = n(n - 1)(n - 2)/6 = 4$ redundancies, and there are in total 12 redundancies in the questionnaire, so that the correct number of degrees of freedom is df = 114, leading to a p-value p = 0.530, and the correct RMSEA is 0. The estimated model parameters are reported in Table 2, along with their standard errors.

———————————————————

Insert Table 2 about here

———————————————————

Estimation of the Thurstonian IRT model for quads using ‘most-least’ response format is performed using syntax ‘*quads_most_least.inp*’, which is given in Appendix B. This syntax is identical to the syntax for full rankings except that multiple datasets are generated prior to estimation using the DATA IMPUTATION command. Here, we order 20 datasets to be generated where missing responses are imputed using Bayesian estimation of the unrestricted model (Asparauhov & Muthén, 2010b). This multiple imputation is followed by the estimation of the forced-choice model for full rankings on each of the imputed datasets, using the ULSMV estimator as usual.

When multiple imputations are used, there is no easy way to combine the model fit test statistics and other fit indices from the imputed samples. M*plus* prints simple averages, which should not be interpreted for model fit (Muthén, 2011). Across 20 imputations, we obtain an average chi-square of χ^2 = 206.15 (SD = 25.01), and using the correct value for degrees of freedom, *df* = 114, the average *p*-value is *p* < 0.001, and the average RMSEA is 0.020. For each individual imputation, the model fit has deteriorated somewhat compared to when the full ranking data was used, which is generally the case with imputed data (Asparauhov & Muthén, 2010b). For comparison, fitting the IRT model straight to data with missing responses '*quads_most_least.dat'* results in very poor model fit (χ^2 = 1009.06, *p* = 0.000, and RMSEA = 0.063). In addition, the model fitted to imputed data recovers the true parameter values well, as can be seen from Table 2, while the model fitted straight to data with missing responses yields factor loadings that are too high. Therefore, multiple imputation is the recommended solution to estimating the Thurstonian IRT model for partial rankings.

Example 3: Block Size n = 2, Measuring More Than Two Attributes (d > 2). In this example we consider a special case of the general theory: items presented in pairs. In this case, no item uniqueness can be identified. It is convenient to assume that both uniquenesses equal 0.5 because in that case the residual variance of the binary outcome equals unity, and the factor loadings and thresholds will be automatically scaled in the IRT intercept/slope parameterization (17). Another feature of this special case is that there are no redundancies among the thresholds and tetrachoric correlations. As a result, the degrees of freedom printed by M*plus* do not need to be adjusted.

To illustrate this case, consider three attributes (*d* = 3), each measured by four items arranged in *p* = 6 pairs (*n* = 2). Thus, there are $p \times \tilde{n} = 6 \times 1 = 6$ binary outcomes in total. For this model, the items' loading matrix $\mathbf{\Lambda}$ (12 × 3) and the pairs' loading matrix $\breve{\mathbf{\Lambda}}$ (6 × 3) are

$$\boldsymbol{\Lambda} = \left(\begin{array}{ccc} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ \hdashline 0 & 0 & \lambda_3 \\ \lambda_4 & 0 & 0 \\ \hdashline 0 & \lambda_5 & 0 \\ 0 & 0 & \lambda_6 \\ \hdashline \vdots & \vdots & \vdots \end{array} \right), \qquad \breve{\boldsymbol{\Lambda}} = \begin{pmatrix} \lambda_1 & -\lambda_2 & 0 \\ -\lambda_4 & 0 & \lambda_3 \\ 0 & \lambda_5 & -\lambda_6 \\ \lambda_7 & -\lambda_8 & 0 \\ -\lambda_{10} & 0 & \lambda_9 \\ 0 & \lambda_{11} & -\lambda_{12} \end{pmatrix}. \qquad (21)$$

It can be seen that presenting the items in pairs as opposed to presenting them one at a time using binary ratings halves the number of obtained observed variables (binary outcomes). It can also be seen that, given the same number of items, pairs yield least binary outcomes compared to triplets (Example 1) and quads (Example 2), hence the pairs design will require more items to achieve a similar amount of information.

The item residual matrix $\boldsymbol{\Psi}^2 = diag\left(\psi_1^2,\dots,\psi_{12}^2\right)$ is diagonal, and the pairwise outcomes' residual matrix $\breve{\boldsymbol{\Psi}}^2$ is also diagonal as shown in (12), $\breve{\boldsymbol{\Psi}}_2^2 = diag\left(\psi_i^2 + \psi_k^2\right)$, with 6 elements that are sums of the original 12 item residuals. In the Thurstonian IRT model, there are 12 factor loadings, 3 correlations between factors, and 6 thresholds to estimate (21 parameters in total). We have only 6 binary outcomes providing 6×7/2 = 21 pieces of information – the model is just identified and the number of degrees of freedom is zero. We can still estimate the model parameters but cannot test the goodness of fit of the model – for that the number of items in the questionnaire has to be larger.

Using the Excel macro for creating syntax in this case is no different from the previous models: one has to specify the data file ('*pairs3traits.dat*'), the block size (2), the number of blocks (6), the number of attributes measured (3), and supply the questionnaire key which in this example will look as follows: (1, 2, 3, 1, 2, 3, 1, 2x, 3, 1x, 2, 3x). The numbers indicate which trait is measured by each item, and 'x' indicates which items are negatively keyed in relation to the measured trait. The syntax written by the Excel macro can be saved as an M*plus* input file. Our syntax '*pairs3traits.inp*' can be found in the supplementary materials; it is also given in Appendix C.

The true and estimated model parameters for this example are reported in Table 3. It can be seen that, again, the true parameters are recovered well.

———————————————

Insert Table 3 about here

———————————————

Example 4: Block Size n = 2, Measuring Exactly Two Attributes (d = 2). In this example we consider a further special case – items presented in *p* pairs (*n* = 2) with exactly two dimensions being measured (*d* = 2). In this case we have an exploratory two-factor analysis model with *p* binary variables.

To see this consider an example where 12 items are presented in *p* = 6 pairs. For simplicity let's assume that the first item in each pair measures the first trait and the second item measures the second trait. For the Thurstonian IRT model, we obtain the residual variance matrix $\breve{\mathbf{\Psi}}^2$ as described in (12), and it is the same as in Example 3. However, while the item factor loading matrix $\mathbf{\Lambda}$ is an independent-clusters solution, the pairs' loading matrix $\breve{\mathbf{\Lambda}}$ has no zero elements:

$$\mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \\ \hdashline \lambda_3 & 0 \\ 0 & \lambda_4 \\ \hdashline \vdots & \vdots \\ \hdashline \lambda_{11} & 0 \\ 0 & \lambda_{12} \end{pmatrix}, \qquad \breve{\mathbf{\Lambda}} = \begin{pmatrix} \lambda_1 & -\lambda_2 \\ \lambda_3 & -\lambda_4 \\ \vdots & \vdots \\ \lambda_{11} & -\lambda_{12} \end{pmatrix}. \qquad (22)$$

Therefore, this is simply an exploratory two-factor model for *p* binary variables. Since the two factors are assumed to be correlated, two elements of $\breve{\mathbf{\Lambda}}$ need to be fixed to identify the model (McDonald, 1999; p. 181). In practice, this can be easily accomplished by fixing the factor loadings of the first item. Any two values will do, provided that the factor loading on the second factor is opposite to its expected value – see (22). For this example, since we wish to show how well we are able to recover the true solution, we set the factor loadings of the first item to their true values.

To create M*plus* syntax using the Excel macro, one has to specify the data file (‘*pairs2traits.dat*’), the block size (2), the number of blocks (6), the number of attributes measured (2), and supply the questionnaire key (1, 2, 1, 2, 1, 2, 1, 2x, 1x, 2, 1, 2x). Our syntax written by the Excel macro ‘*pairs2traits.inp*’ can be found in the supplementary materials; it is also given in Appendix D.

Testing this model with the supplied data yields χ^2 = 3.40 on 4 degrees of freedom (which is the correct number and does not need adjustment when items are presented in pairs), the p-value is p = 0.494 and RMSEA = 0. The estimated and the true model parameter values are presented in Table 4 – and it can be seen that the model recovers the true parameter values well.

––––––––––––––––––––

Insert Table 4 here

––––––––––––––––––––

Concluding Remarks

Because of their advantages in reducing or counteracting some response biases commonly arising when using the rating scales, forced-choice assessments are becoming increasingly popular and forced-choice measurement is a growing area of research. With development of models suitably describing comparative data, such as the Thurstonian IRT model discussed here, or the Multi-Unidimensional Pairwise-Preference Model (Stark, Chernyshenko & Drasgow, 2005), and availability of software capable of fitting them, such modeling will become more accessible to researchers.

Despite the ease with which forced-choice data can be tested using the provided tutorial and the automated syntax writer (Excel macro), however, one needs to pause and consider all ‘specialties’ of the forced-choice format and the data arising from it. Because every judgment made in this format is a *relative* judgment, careful consideration is needed to design forced-choice questionnaires that will be capable of recovering *absolute* trait scores from these relative judgments.

Maydeu-Olivares and Brown (2010) discuss rules governing good forced-choice measurement with one measured trait. As can be seen from (11), in the one-dimensional case the discrimination power of each comparison is determined by the *difference* of factor loadings of the two items involved. Two perfectly good, equally discriminating items, therefore, could be put together to produce a useless forced-choice pair with near-zero discrimination. To maximize efficiency of the forced-choice format in this case, one needs to combine items with widely varying factor loadings – for instance, with positive and negative loadings, or with high and low positive loadings. If socially desirable responding is a concern, special care must be taken to create pairs with no obvious valence. This might be challenging when items with positive and negative loadings are combined in one block, and consequently measuring one trait with forced-choice items might not be any more robust to socially desirable responding than using single-stimulus items. The universal benefit of the forced-choice format – removal of uniforms biases, such as acquiescence or central tendency responding – will of course remain.

When multidimensional forced-choice blocks are used, yet more factors need to be taken to account. All of the following – keyed direction of items, number of measured attributes, correlations between the attributes, and block size – are important (Brown & Maydeu-Olivares, 2011a). For instance, when a larger number of attributes (15 or more) are modeled, all positively keyed items may be used to successfully recover the individual scores (Brown, 2010) provided that the traits are not too highly correlated. However, if only a small number of latent traits are assessed, as was the case in the numerical examples in this paper, both positively and negatively keyed items must be combined in blocks in order to accurately recover the true model parameters and the individual scores. In this case, considerations of socially desirable responding discussed above also apply, although matching positively and negatively keyed items on social desirability may be easier when the items measure different attributes.

In closing, since the purpose of this paper is expository, very short questionnaires were used. Yet, IRT parameter recovery and latent trait estimation accuracy depend critically on the number of items per dimension. In applications, a larger number of indicators per dimension should be used, leading to more accurate item parameter and latent trait estimates than those reported here – see Brown and Maydeu-Olivares (2011a) for detailed simulation studies results. An additional consideration is that, given the same number of items, smaller blocks (i.e. pairs) produce fewer binary outcomes per items used, and therefore provide less information for the person’s scores estimation than larger blocks (i.e. triplets, quads).

The Thurstonian IRT model has been successfully used with real questionnaire data, with the primary objectives to estimate the item parameters and the correlations between the latent traits, and to score test takers on the measured attributes. One example is the Forced-Choice Five Factor Markers (Brown & Maydeu-Olivares, 2011b), which is a short forced-choice questionnaire consisting of 20 triplets with both positively and negatively keyed items. Its IRT modeling in a research sample yielded successful estimation of the absolute trait standing as compared to the normative scores using rating scales (Brown & Maydeu-Olivares, 2011a). Other applications with real questionnaire data include the development of the IRT-scored Occupational Personality Questionnaire (OPQ32r; Brown & Bartram, 2009), and the construct and criterion validity study using the Customer Contact Styles Questionnaire (CCSQ; Brown, 2010). These large-scale workplace questionnaires measuring 32 and 16 attributes respectively are based on multidimensional comparisons with positively keyed items only.

In this paper we have provided a tutorial on how to fit the Thurstonian IRT model to any forced-choice questionnaire design using M*plus*. With this paper we also supply an easy-to-use Excel macro that writes M*plus* syntax for all such designs. Equipped with these tools, the reader can model any forced-choice data – e.g. estimate model-based correlations between the psychological attributes – adequately, without distortions caused by the use of classical scoring procedures. Most importantly, this modeling enables access to persons’ scores on latent attributes that are no longer ipsative.

References

Asparouhov, T. & Muthén, B. (2010a). *Bayesian analysis of latent variable models using Mplus.* Version 4. Retrieved from http://www.statmodel.com/download/BayesAdvantages18.pdf

Asparouhov, T. & Muthén, B. (2010b). *Multiple imputation with Mplus*. *Version 2*. Retrieved from http://www.statmodel.com/download/Imputations7.pdf

Baron, H. (1996). Strengths and limitations of ipsative measurement. *Journal of Occupational and Organizational Psychology*, *69*, 49–56. doi: 10.1111/j.2044-8325.1996.tb00599.x

Bartram, D. (2007). Increasing validity with forced-choice criterion measurement formats. *International Journal of Selection and Assessment*, *15*, 263-272. doi: 10.1111/j.1468-2389.2007.00386.x

Bock, R. D. & Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. *Psychometrika*, *46*, 443–459. doi: 10.1007/BF02293801

Brown, A. (2010). How IRT can solve problems of ipsative data (Doctoral dissertation). University of Barcelona. Retrieved from http://hdl.handle.net/10803/80006

Brown, A. & Bartram, D. (2009, April). *Doing less but getting more: Improving forced-choice measures with IRT*. Paper presented at the 24th conference of the Society for Industrial and Organizational Psychology, New Orleans, LA. Retrieved from http://www.shl.com/assets/resources/Presentation-2009-Doing-less-but-getting-more-SIOP.pdf

Brown, A. & Maydeu-Olivares, A. (2011a). Item response modeling of forced-choice questionnaires. *Educational and Psychological Measurement*, *71*, 460-502. doi: 10.1177/0013164410375112

Brown, A. & Maydeu-Olivares, A. (2011b). Forced-choice Five Factor markers. Retrieved from PsycTESTS. doi:10.1037/t05430-000

Cheung, M.W.L, & Chan, W. (2002). Reducing uniform response bias with ipsative measurement in multiple-group confirmatory factor analysis. *Structural Equation Modeling*, *9*, 55-77. doi: 10.1207/S15328007SEM0901_4

Christiansen, N, Burns, G., & Montgomery, G. (2005). Reconsidering the use of forced-choice formats for applicant personality assessment. *Human Performance*, *18*, 267-307. doi: 10.1207/s15327043hup1803_4

Clemans, W. V. (1966). *An Analytical and Empirical Examination of Some Properties of Ipsative Measures* (Psychometric Monograph No. 14). Richmond, VA: Psychometric Society. Retrieved from http://www.psychometrika.org/journal/online/MN14.pdf

Forero, C.G., Maydeu-Olivares, A. & Gallardo-Pujol, D. (2009). Factor analysis with ordinal indicators: A Monte Carlo study comparing DWLS and ULS estimation. *Structural Equation Modeling*, *16*, 625–641. doi: 10.1080/10705510903203573

Jackson, D., Wroblewski, V., & Ashton, M. (2000). The Impact of Faking on Employment Tests: Does Forced Choice Offer a Solution? *Human Performance*, *13*, 371–388. doi: 10.1207/S15327043HUP1304_3

Lord, F. (1952). *A Theory of Test Scores* (Psychometric Monograph No. 7). Richmond, VA: Psychometric Corporation. Retrieved from http://www.psychometrika.org/journal/online/MN07.pdf

Maydeu-Olivares, A. (1999). Thurstonian modeling of ranking data via mean and covariance structure analysis. *Psychometrika*, *64*, 325-340. doi: 10.1007/BF02294299

Maydeu-Olivares, A. & Böckenholt, U. (2005). Structural equation modeling of paired-comparison and ranking data. *Psychological Methods*, *10*, 285-304. doi: 10.1037/1082-989X.10.3.285

Maydeu-Olivares, A. & Brown, A. (2010). Item response modeling of paired comparison and ranking data. *Multivariate Behavioral Research*, *45*, 935-974. doi: 10.1177/0013164410375112

McDonald, R.P. (1999). *Test theory. A unified approach*. Mahwah, NJ: Lawrence Erlbaum.

Meade, A. (2004). Psychometric problems and issues involved with creating and using ipsative measures for selection. *Journal of Occupational and Organisational Psychology*, *77*, 531-552. doi: 10.1348/0963179042596504

Muthén, B.O. (1998-2004). *Mplus Technical Appendices*. Los Angeles, CA: Muthén & Muthén. Retrieved from http://www.statmodel.com/download/techappen.pdf

Muthén, L. K. (2011, June 28). Multiple imputations. Message posted to http://www.statmodel.com/discussion/messages/22/381.html

Muthén, L.K. *&* Muthén, B.O. (1998-2010). *Mplus User's guide. Sixth edition*. Los Angeles, CA: Muthén & Muthén. Retrieved from www.statmodel.com

Satorra, A., & Bentler, P. M. (1994). Corrections to test statistics and standard errors in covariance structure analysis. In A. von Eye & C. C. Clogg (Eds.), *Latent variable analysis: Applications to developmental research* (pp. 399–419). Thousand Oaks, CA: Sage.

Stark, S., Chernyshenko, O. & Drasgow, F. (2005). An IRT approach to constructing and scoring pairwise preference items involving stimuli on different dimensions: The Multi-Unidimensional Pairwise-Preference Model. *Applied Psychological Measurement*, *29*, 184-203. doi: 10.1177/0146621604273988

Thurstone, L.L. (1927). A law of comparative judgment. *Psychological Review*, *34, 273-286*. doi: 10.1037/h0070288

Thurstone, L.L. (1931). Rank order as a psychophysical method. *Journal of Experimental Psychology*, *14*, 187-201. doi: 10.1037/h0070025

Appendix A

Mplus Input File for Example 1: Block Size *n* = 3, Full Ranking Response Format

```
TITLE: Example 1 - Model with 3 triplets measuring 3 traits
DATA: FILE IS triplets.dat;

VARIABLE:
    NAMES =
    i1i2 i1i3 i2i3
    i4i5 i4i6 i5i6
    i7i8 i7i9 i8i9
    i10i11 i10i12 i11i12;
    CATEGORICAL = i1i2-i11i12;

ANALYSIS:
ESTIMATOR=ulsmv;  PARAMETERIZATION=THETA;

MODEL:
  Trait1  BY
        i1i2*1  i1i3*1  (L1)
        i4i5*-1  i4i6*-1  (L4)
        i7i8*1  i7i9*1  (L7)
        i10i11*1  i10i12*1  (L10);
  Trait2  BY
        i1i2*-1  (L2_n)
        i2i3*1  (L2)
        i4i5*-1  (L5_n)
        i5i6*1  (L5)
        i7i8*-1  (L8_n)
        i8i9*1  (L8)
         i10i11*1  (L11_n)
         i11i12*-1  (L11);
  Trait3  BY
        i1i3*-1  i2i3*-1  (L3_n)
        i4i6*-1  i5i6*-1  (L6_n)
        i7i9*1  i8i9*1  (L9_n)
         i10i12*-1  i11i12*-1  (L12_n);

Trait1-Trait3@1      ! variances for all traits are set to 1

! optional - starting values for correlations between traits
Trait1 WITH Trait2*-0.4  Trait3*0;
Trait2 WITH Trait3*0.3;

! declare uniquenesses
i1i2*2 (e1e2);
i1i3*2 (e1e3);
i2i3*2 (e2e3);
```

```
i4i5*2 (e4e5);
i4i6*2 (e4e6);
i5i6*2 (e5e6);
i7i8*2 (e7e8);
i7i9*2 (e7e9);
i8i9*2 (e8e9);
i10i11*2 (e10e11);
i10i12*2 (e10e12);
i11i12*2 (e11e12);

! declare correlated uniqunesses and set their starting values
i1i2 WITH i1i3*1 (e1);
i1i2 WITH i2i3*-1 (e2_n);
i1i3 WITH i2i3*1 (e3);
i4i5 WITH i4i6*1 (e4);
i4i5 WITH i5i6*-1 (e5_n);
i4i6 WITH i5i6*1 (e6);
i7i8 WITH i7i9*1 (e7);
i7i8 WITH i8i9*-1 (e8_n);
i7i9 WITH i8i9*1 (e9);
i10i11 WITH i10i12*1 (e10);
i10i11 WITH i11i12*-1 (e11_n);
i10i12 WITH i11i12*1 (e12);

MODEL CONSTRAINT:
!factor loadings relating to the same item are equal in absolute value
L2_n = -L2;    L5_n = -L5;    L8_n = -L8;    L11_n = -L11;

! pair's uniqueness is equal to sum of 2 utility uniqunesses
e1e2 = e1 - e2_n;
e1e3 = e1 + e3;
e2e3 = -e2_n + e3;
e4e5 = e4 - e5_n;
e4e6 = e4 + e6;
e5e6 = -e5_n + e6;
e7e8 = e7 - e8_n;
e7e9 = e7 + e9;
e8e9 = -e8_n + e9;
e10e11 = e10 - e11_n;
e10e12 = e10 + e12;
e11e12 = -e11_n + e12;

! fix one uniqueness per block for identification
e1=1;   e4=1;   e7=1;   e10=1;
```

Appendix B

Mplus Input File for Example 2: Block Size *n* = 4, ‘Most-Least’ Response Format

```
TITLE: Model with 3 'most-least' blocks measuring 4 traits, with imputation of missing data

DATA: FILE IS quads_most_least.dat;

VARIABLE:
    NAMES ARE
    i1i2 i1i3 i1i4 i2i3 i2i4 i3i4
    i5i6 i5i7 i5i8 i6i7 i6i8 i7i8
    i9i10 i9i11 i9i12 i10i11 i10i12 i11i12;

   CATEGORICAL = i1i2-i11i12;
   MISSING ARE ALL *;

DATA IMPUTATION:
  IMPUTE = i1i2-i11i12(c);
  NDATASETS = 20;

ANALYSIS:
ESTIMATOR=ULSMV;    PARAMETERIZATION=THETA;

MODEL:
Trait1  BY
        i1i2*1  i1i3*1  i1i4*1  (L1)
        i5i6*-1  i5i7*-1  i5i8*-1  (L5)
        i9i10*1  i9i11*1  i9i12*1  (L9);
Trait2  BY
        i1i2*1  (L2_n)
        i2i3*-1  i2i4*-1  (L2)
        i5i6*-1  (L6_n)
        i6i7*1  i6i8*1  (L6)
        i9i10*-1  (L10_n)
        i10i11*1  i10i12*1  (L10);
Trait3  BY
        i1i3*-1  i2i3*-1  (L3_n)
        i3i4*1  (L3)
        i5i7*-1  i6i7*-1  (L7_n)
        i7i8*1  (L7)
        i9i11*1  i10i11*1  (L11_n)
        i11i12*-1  (L11);
Trait4  BY
        i1i4*-1  i2i4*-1  i3i4*-1  (L4_n)
        i5i8*-1  i6i8*-1  i7i8*-1  (L8_n)
        i9i12*-1  i10i12*-1  i11i12*-1  (L12_n);
```

```
! variances for all traits are set to 1
Trait1-Trait4@1;

! optional - starting values for correlations between traits
Trait1 WITH Trait2*-0.4 Trait3*0 Trait4*0.4;
Trait2 WITH Trait3*0.3 Trait4*-0.3;
Trait3 WITH Trait4*0;

! declare uniquenesses and set their starting values
i1i2*2 (e1e2);
i1i3*2 (e1e3);
i1i4*2 (e1e4);
i2i3*2 (e2e3);
i2i4*2 (e2e4);
i3i4*2 (e3e4);
i5i6*2 (e5e6);
i5i7*2 (e5e7);
i5i8*2 (e5e8);
i6i7*2 (e6e7);
i6i8*2 (e6e8);
i7i8*2 (e7e8);
i9i10*2 (e9e10);
i9i11*2 (e9e11);
i9i12*2 (e9e12);
i10i11*2 (e10e11);
i10i12*2 (e10e12);
i11i12*2 (e11e12);

! declare correlated uniqunesses and set their starting values
i1i2 WITH i1i3*1 i1i4*1 (e1);
 i1i2 WITH i2i3*-1 i2i4*-1 (e2_n);
i1i3 WITH i1i4*1 (e1);
i1i3 WITH i2i3*1 (e3);
i1i3 WITH i3i4*-1 (e3_n);
i1i4 WITH i2i4*1 i3i4*1 (e4);
i2i3 WITH i2i4*1 (e2);
i2i3 WITH i3i4*-1 (e3_n);
i2i4 WITH i3i4*1 (e4);

i5i6 WITH i5i7*1 i5i8*1 (e5);
i5i6 WITH i6i7*-1 i6i8*-1 (e6_n);
i5i7 WITH i5i8*1 (e5);
i5i7 WITH i6i7*1 (e7);
i5i7 WITH i7i8*-1 (e7_n);
i5i8 WITH i6i8*1 i7i8*1 (e8);
i6i7 WITH i6i8*1 (e6);
i6i7 WITH i7i8*-1 (e7_n);
i6i8 WITH i7i8*1 (e8);
```

```
i9i10 WITH i9i11*1 i9i12*1 (e9);
i9i10 WITH i10i11*-1 i10i12*-1 (e10_n);
i9i11 WITH i9i12*1 (e9);
i9i11 WITH i10i11*1 (e11);
i9i11 WITH i11i12*-1 (e11_n);
i9i12 WITH i10i12*1 i11i12*1 (e12);
i10i11 WITH i10i12*1 (e10);
i10i11 WITH i11i12*-1 (e11_n);
i10i12 WITH i11i12*1 (e12);

MODEL CONSTRAINT:
!factor loadings relating to the same item are equal in absolute value
L2_n = -L2;
L3_n = -L3;
L6_n = -L6;
L7_n = -L7;
L10_n = -L10;
L11_n = -L11;

!uniquenesses relating to the same item are equal in absolute value
e2_n = -e2;
e3_n = -e3;
e6_n = -e6;
e7_n = -e7;
e10_n = -e10;
e11_n = -e11;

! pair's uniqueness is equal to sum of 2 utility uniqunesses
e1e2 = e1 + e2;         e1e3 = e1 + e3;         e1e4 = e1 + e4;
e2e3 = e2 + e3;         e2e4 = e2 + e4;         e3e4 = e3 + e4;
e5e6 = e5 + e6;         e5e7 = e5 + e7;         e5e8 = e5 + e8;
e6e7 = e6 + e7;         e6e8 = e6 + e8;         e7e8 = e7 + e8;
e9e10 = e9 + e10;       e9e11 = e9 + e11;       e9e12 = e9 + e12;
e10e11 = e10 + e11;     e10e12 = e10 + e12;     e11e12 = e11 + e12;

! fix one uniqueness per block for identification
e1=1;          e5=1;          e9=1;
```

Appendix C

Mplus Input File for Example 3: Block Size *n* = 2, Measuring 3 Attributes

```
TITLE: Model with 6 pairs measuring 3 traits
DATA: FILE IS pairs3traits.dat;

VARIABLE:
    NAMES = i1i2 i3i4 i5i6 i7i8 i9i10 i11i12;
    CATEGORICAL = i1i2-i11i12;
```

```
ANALYSIS:
ESTIMATOR=ulsmv;  PARAMETERIZATION=THETA;

MODEL:
Trait1  BY      i1i2*1  i3i4*-1  i7i8*1  i9i10*1;
Trait2  BY      i1i2*-1  i5i6*1  i7i8*1  i11i12*1;
Trait3  BY      i3i4*1  i5i6*-1  i9i10*1  i11i12*1;

! variances for all traits are set to 1
Trait1-Trait3@1;

! optional - starting values for correlations between traits
Trait1 WITH Trait2*-0.4 Trait3*0;
Trait2 WITH Trait3*0.3;

! set uniquenesses of all outcomes for identification
i1i2-i11i12@1;
```

Appendix D

Mplus Input File for Example 4: Block Size *n* = 2, Measuring 2 Attributes

```
TITLE: Model with 6 pairs measuring only 2 traits
DATA: FILE IS pairs2traits.dat;

VARIABLE:
    NAMES = i1i2 i3i4 i5i6 i7i8 i9i10 i11i12;
    CATEGORICAL = i1i2-i11i12;

ANALYSIS:
ESTIMATOR=ulsmv;   PARAMETERIZATION=THETA;

MODEL:
Trait1  BY
        i1i2@.6   ! fixed for model identification
        i3i4*1  i5i6*1  i7i8*1  i9i10*-1  i11i12*1;
Trait2  BY
        i1i2@-.8  ! fixed for model identification
        i3i4*-1  i5i6*-1  i7i8*1  i9i10*-1  i11i12*1;

! variances for all traits are set to 1
Trait1-Trait2@1;

! optional - starting values for correlations between traits
Trait1 WITH Trait2*0;

! set uniquenesses of all outcomes for identification
i1i2-i11i12@1;
```

Table 1

True and estimated parameters for Example 1: 3 traits measured by 4 triplets

par.	true	est.	par.	true	est.	par.	true	est.
λ_1	1	1.08 (0.14)	ψ_1^2	1	1 (--)	γ_1	0.5	0.56 (0.08)
λ_2	0.8	0.86 (0.11)	ψ_2^2	1	1.17 (0.30)	γ_2	-1.2	-1.25 (0.12)
λ_3	1.3	1.36 (0.14)	ψ_3^2	1	0.88 (0.29)	γ_3	-1.7	-1.73 (0.18)
λ_4	-1.3	-1.30 (0.17)	ψ_4^2	1	1 (--)	γ_4	0.7	0.62 (0.07)
λ_5	1	1.00 (0.13)	ψ_5^2	1	0.87 (0.23)	γ_5	1	0.94 (0.10)
λ_6	0.8	0.80 (0.11)	ψ_6^2	1	1.23 (0.28)	γ_6	0.3	0.30 (0.06)
λ_7	0.8	0.80 (0.10)	ψ_7^2	1	1 (--)	γ_7	-0.7	-0.67 (0.08)
λ_8	1.3	1.32 (0.13)	ψ_8^2	1	0.76 (0.26)	γ_8	-1.2	-1.13 (0.09)
λ_9	-1	-0.97 (0.10)	ψ_9^2	1	0.80 (0.22)	γ_9	-0.5	-0.45 (0.07)
λ_{10}	1.3	1.08 (0.11)	ψ_{10}^2	1	1 (--)	γ_{10}	0.7	0.63 (0.06)
λ_{11}	-0.8	-0.63 (0.08)	ψ_{11}^2	1	0.89 (0.18)	γ_{11}	1.2	1.15 (0.09)
λ_{12}	1	0.81 (0.08)	ψ_{12}^2	1	0.79 (0.18)	γ_{12}	0.5	0.50 (0.06)
ϕ_{12}	-0.4	-0.39 (0.04)	ϕ_{13}	0	0.00 (0.05)			
			ϕ_{23}	0.3	0.34 (0.05)			

Notes: Standard errors in parentheses. *N* = 2000. First uniqueness in each block is set to 1 for identification, $\psi_1^2 = \psi_4^2 = \psi_7^2 = \psi_{10}^2 = 1$.

Table 2

True and estimated parameters for Example 2: 4 traits measured by 3 quads

par.	true	est. full ranking	est. most-least	par.	true	est. full ranking	est. most-least	par.	true	est. full ranking	est. most-least
λ_1	1	1.09 (0.13)	1.04 (0.14)	ψ_1^2	1	1 (--)	1 (--)	γ_1	0.5	0.57 (0.06)	0.56 (0.07)
λ_2	-0.8	-0.83 (0.09)	-0.77 (0.1)	ψ_2^2	1	1.02 (0.19)	0.97 (0.19)	γ_2	-1	-0.97 (0.09)	-0.96 (0.1)
λ_3	1.3	1.25 (0.12)	1.25 (0.13)	ψ_3^2	1	1.47 (0.33)	1.28 (0.34)	γ_3	0.5	0.59 (0.06)	0.6 (0.07)
λ_4	0.8	0.74 (0.09)	0.69 (0.09)	ψ_4^2	1	1.25 (0.21)	1.22 (0.22)	γ_4	-1.5	-1.5 (0.13)	-1.4 (0.13)
								γ_5	0	0.04 (0.05)	0.02 (0.06)
								γ_6	1.5	1.57 (0.13)	1.51 (0.13)
λ_5	-1.3	-1.25 (0.18)	-1.25 (0.23)	ψ_5^2	1	1 (--)	1 (--)	γ_7	-0.3	-0.34 (0.06)	-0.32 (0.07)
λ_6	1	1.08 (0.13)	1.08 (0.16)	ψ_6^2	1	0.83 (0.2)	0.78 (0.24)	γ_8	-0.3	-0.36 (0.07)	-0.33 (0.07)
λ_7	0.8	0.8 (0.11)	0.8 (0.12)	ψ_7^2	1	1.25 (0.21)	1.22 (0.22)	γ_9	-0.8	-0.79 (0.1)	-0.88 (0.13)
λ_8	1.3	1.3 (0.14)	1.22 (0.19)	ψ_8^2	1	0.65 (0.27)	0.83 (0.31)	γ_{10}	0	-0.09 (0.05)	-0.09 (0.05)
								γ_{11}	-0.5	-0.53 (0.08)	-0.47 (0.09)
								γ_{12}	-0.5	-0.5 (0.07)	-0.5 (0.08)
λ_9	0.8	0.84 (0.1)	0.9 (0.17)	ψ_9^2	1	1 (--)	1 (--)	γ_{13}	1.5	1.62 (0.13)	1.6 (0.16)
λ_{10}	1.3	1.41 (0.13)	1.38 (0.17)	ψ_{10}^2	1	1.35 (0.31)	1.37 (0.39)	γ_{14}	2	2.16 (0.15)	2.19 (0.24)
λ_{11}	-1	-1.06 (0.11)	-1.14 (0.14)	ψ_{11}^2	1	0.89 (0.24)	0.81 (0.28)	γ_{15}	0.5	0.49 (0.06)	0.55 (0.07)
λ_{12}	1	0.99 (0.1)	1.07 (0.16)	ψ_{12}^2	1	1.18 (0.23)	1.19 (0.28)	γ_{16}	0.5	0.51 (0.08)	0.5 (0.09)
								γ_{17}	-1	-1.04 (0.1)	-1.05 (0.13)
								γ_{18}	-1.5	-1.61 (0.13)	-1.71 (0.19)
ϕ_{12}	-0.4	-0.43 (0.04)	-0.43 (0.06)	ϕ_{13}	0	-0.02 (0.05)	-0.03 (0.06)	ϕ_{14}	0.4	0.39 (0.04)	0.39 (0.05)
				ϕ_{23}	0.3	0.33 (0.05)	0.35 (0.05)	ϕ_{24}	-0.3	-0.29 (0.05)	-0.31 (0.06)
								ϕ_{34}	0	0.08 (0.05)	0.10 (0.06)

Notes: Standard errors in parentheses. *N* = 2000. First uniqueness in each block is set to 1 for identification, $\psi_1^2 = \psi_5^2 = \psi_9^2 = 1$. Parameters for full ranking data are based on one dataset; parameters for most-least data are averaged across 20 imputed datasets.

Table 3

True and estimated parameters for Example 3: 3 traits measured by 6 pairs

par.	true	est.	par.	true	est.
λ_1	0.6	0.63 (0.12)	γ_1	0.5	0.59 (0.07)
λ_2	1.0	1.00 (0.17)			
λ_3	0.8	0.81 (0.16)	γ_2	-0.7	-0.66 (0.07)
λ_4	1.0	0.86 (0.16)			
λ_5	0.6	0.62 (0.16)	γ_3	0.5	0.42 (0.05)
λ_6	1.0	0.97 (0.18)			
λ_7	0.8	0.73 (0.18)	γ_4	-0.8	-0.82 (0.08)
λ_8	-1.0	-0.95 (0.20)			
λ_9	0.6	0.58 (0.12)	γ_5	0.3	0.37 (0.05)
λ_{10}	-0.6	-0.92 (0.15)			
λ_{11}	0.8	0.66 (0.11)	γ_6	0.7	0.66 (0.06)
λ_{12}	-0.8	-0.77 (0.11)			
ϕ_{12}	-0.4	-0.33 (0.09)	ϕ_{13}	0	0.07 (0.10)
			ϕ_{23}	0.3	0.36 (0.09)

Notes: Standard errors in parentheses. N = 2000. All item uniquenesses are set to 0.5 for identification.

Table 4

True and estimated parameters for Example 4: 2 traits measured by 6 pairs

par.	true	est.	par.	true	est.
λ_1	0.6	0.6 (--)	γ_1	0.50	0.51 (0.04)
λ_2	0.8	0.8 (--)			
λ_3	0.8	0.81 (0.13)	γ_2	-0.70	-0.64 (0.06)
λ_4	1.0	1.00 (0.16)			
λ_5	1.0	1.08 (0.15)	γ_3	0.50	0.53 (0.06)
λ_6	0.6	0.70 (0.16)			
λ_7	0.8	0.63 (0.09)	γ_4	-0.80	-0.72 (0.06)
λ_8	-1.0	-0.84 (0.11)			
λ_9	-0.6	-0.63 (0.07)	γ_5	0.30	0.29 (0.04)
λ_{10}	0.6	0.59 (0.09)			
λ_{11}	0.8	0.86 (0.12)	γ_6	0.70	0.77 (0.07)
λ_{12}	-0.8	-0.85 (0.14)			
ϕ_{21}	0	0.16 (0.15)			

Notes: Standard errors in parentheses. N = 2000. All item uniquenesses are set to 0.5 for identification.