KU LEUVEN

FACULTY OF SCIENCE

Generalized Linear Latent and Mixed Model:

method, bayesian estimation, advantages, and applications to educational data.

Jose Manuel Rivera Espejo

Supervisor: Prof. Geert Molenbegrhs
Leuven Biostatistics and Statistical Bioinformatics Centre (L-BioStat)

Co-supervisor: Prof. Wim Van den Noortgate
Faculty of Psychology and Educational Sciences

Thesis presented in fulfillment of the requirements for the degree of Master of Science in Statistics and Data Science for Social, Behavioral and Educational Sciences

Academic year 2020-2021

Dedication

To Manuel, for being my friend and father.
To Margarita, Karina, Susan, and Marysu, for their relentless encouragement.
To Ana, for showing me the value of family, here in this moorland.
To both of you, as you are always in my mind.
And to all that knowingly or not, help me to get here.
I am lucky due to all of you.
I hope I make you all proud.

A Manuel, por ser mi amigo y mi padre.
A Margarita, Karina, Susan y Marysu, por su incansable aliento.
A Ana, por mostrarme el valor de la familia, aquí en este páramo.
A ustedes dos, que siempre las tengo en mente.
Y a todos los que sabiendolo o no, me ayudaron a llegar aquí.
Soy un suertudo gracias todos ustedes.
Espero llenarlos de orgullo.

Acknowledgment

To the friends that managed to keep me motivated through this rewarding endeavor. Thank you for all the great discussions, and your cool friendship. Thank you Qian "Erika", Luc, Jonathan, Youhee, Sarah, Nura, Jingpu, and Jasper. You were sent from heaven, if something like that exists.

Abstract

The current research thesis described and implemented the Bayesian GLLAMM model for dichotomous outcomes [56, 58, 65, 59], in the context of an educational data. The model's proposal revolved around the fact that educational data often presents multiple types of dependencies, that left unchecked, can cause IRT models to violate their assumptions of local independence. The latter is particularly important, as violation of these assumption prevent IRT models to reach appropriate inferences from the parameter estimates [73, 9, 32].

Moreover, in the context of the previously defined model, the current research also provided an assessment of the benefits resulting from changing the posterior sampling geometries. Multiple evidence pointed out the performance improvement on the MCMC methods from using non-centered parameterizations [18, 19, 51, 52, 6]. However, most of the evidence have been developed under Gaussian hierarchical models. So, it seemed sensible to provide a similar assessment for nonlinear latent stochastic models, like our implementation [52].

Finally, the research applied the newfound knowledge to a large standardized teacher assessments from Peru. The purpose of the latter was to evaluate the change of parametrization on a real data setting, determine the evidence in favor of our models of interest, produce psychometric analysis, and finally assess specific research hypothesis.

Keywords: Bayesian modeling, MCMC, HMC, IRT, GLLAMM, centered parametrization, non-centered parametrization.

Contents

List of Figures

List of Tables

Abbreviations

ACF	Auto-correlation function.
BUGS	Bayesian Inference Using Gibbs Sampling.
CFA	Confirmatory Factor Analysis.
CLT	Central Limit Theorem.
CP	Centered Parametrization.
DAG	Directed Acyclic Graph.
DDM	Dual Dependence Models.
EFA	Exploratory Factor Analysis.
FOLV	First-order Latent Variable.
GCM	Graphical Causal Model.
GLLAMM	Generalized Linear Latent and Mixed Model.
GLM	Generalized Linear Model.
GLMM	Generalized Linear Mixed Model.
HMC	Hamiltonian Monte Carlo.
ICC	Item Characteristic Curve.
IIF	Item Information Function.
iHMC	Interleaved Hamiltonian Monte Carlo.
JAGS	Just Another Gibbs Sampler.
IRT	Item Response Theory.
MCMC	Markov Chain Monte Carlo.
ML	Maximum Likelihood.
MSEM	Multilevel Structural Equation Model.
NCP	Non-centered Parametrization.
RMSE	Root Mean Squared Error.
SEM	Structural Equation Model.
SOLV	Second-order Latent Variable.
ULI	Unit Loading Identification.
UVI	Unit Variance Identification.
VI	Variational Inference.

Chapter 1

Introduction

1.1 Preliminar considerations

Local independence is one of the key assumptions of Item Response Theory (IRT) models, and it is comprised of two parts: (i) local item independence and (ii) local individual independence [3, 27]. In the former case, the assumption entails that the individual's response to an item does not affect the probability of endorsing another item, after conditioning on the individual's ability. While in the case of the latter, the assumption considers that an individual's response to an item, is independent of another person's response to that same item, or any other [62].

The literature has shown that IRT models are not robust to the violation of local independence. The transgression of the assumption affects model parameter estimates, inflates measurement reliabilities and test information, and underestimates standard errors (see Yen [73], Chen and Thissen [9], and Jiao et al. [32]).

However, item response data arising from educational assessments often display several types of dependencies, violating the local item and/or individual independence, e.g. testlets, where items are constructed around a common stimulus [70]; the measurement of multiple latent traits within individuals [62]; cluster effects, where correlation among individuals results from the sampling and measurement mechanism used to gather the data [61]; among others. A good motivating example, that will permeate this research, is the reading comprehension sub-test, from the Peruvian public teaching career national assessment. The test is designed to measure three hierarchically nested sub-dimensions of reading comprehension: literal, inferential, and reflective abilities. Furthermore, the items are bundled together in testlets related to a common text or passage. Finally, multiple cluster effects are present, e.g. at the region, and district level, just to mention a few.

Recent studies have proposed IRT Dual Dependency Models (DDM) to deal with the testlets and individual clustering dependencies observed in the data [17, 16, 15, 32, 13, 14, 62, 8]. The majority of these representations have been developed under the Bayesian framework, and they are similar in parametrization to multilevel models. On the other hand, an almost independent line of research, the Generalized Linear Latent and Mixed Models (GLLAMM) [56, 58, 65, 59], have extended the capabilities of hierarchical models on the estimation of multiple latent traits at different hierarchical levels. These developments have been motivated mostly under the frequentist framework, and they are similar in parametrization to a Multilevel Structural Equation Model (MSEM).

While the initial sense is that both developments are independent of each other, follow-

ing their literature, one can easily notice that they share more than a resemblance. Both follow a multilevel/hierarchical multidimensional approach to account for the clustering of persons within samples and/or items within bundles (DDM), or the latent structures within the individuals (GLLAMM). However, it is important to point out that in some cases the model parametrization between the two developments differs in a way, that some of them appear to be useful only under their specific contexts. Fortunately, their integration under the Bayesian framework is not only trivial, but it can be motivated under either type of model.

The benefits of the integration revolve around two facts: (i) educational data often presents all of the aforementioned dependencies and more, as in the motivating example; and (ii) as it was hinted in the second paragraph, to reach appropriate conclusions from the parameter estimates, IRT models need to account for all of these dependencies. The latter is particularly important as, more often than not, a researcher is interested in producing inferences at the structural level of the model, i.e. how a different set of manifest variables explain the variability in the latent variables, or how the latent variables explain other manifest or latent variables, at different levels. As an example, one might be interested in finding evidence if the latent "abilities" of the teachers are explained by their initial educational conditions, i.e. if they were educated in an institute, university, or both. The main purpose of this would be to identify the type of teacher that might benefit more from the in-service training[1], offered by the national educational authorities, making the intervention cost-effective.

From the previous description, one can infer that the proposed IRT representation would be complex and highly dimensional. Moreover, as educational assessments are usually scored in a binary way (the individual either endorse or not the item), and because not all individuals are assessed by all items, the model will be estimated with sparse data. From the modeling perspective, neither of the previous points presents a challenge for the bayesian framework. However, it has long been recognized that complex parametrizations, that allow this powerful modeling schemes, introduce pathologies that make Markov Chain Monte Carlo methods (MCMC) face performance challenges [18, 19, 51, 52, 6], e.g. not achieving stationarity and/or not making a proper exploration of the posterior sampling space. This is highly relevant because, in order to make inferences about the posterior distribution of the parameters, the chains need to achieve a requirement highly related to the performance of the method: ergodicity [46], i.e. stationarity, convergence, and good mixing [45].

Throughout the bayesian IRT literature, one often finds that four solutions are offered to ensure the fulfillment of the previous requirements, and they can be classified into two broad groups: (i) solutions that involve changing the settings of the MCMC method, and (ii) solutions that involve readjusting the Bayesian model.

In the first category, we find two proposals: (a) increasing the number of iterations per chain, with large burn-in and thinning processes, and (b) designing model-specific MCMC algorithms. The easiest to implement and more prevalent in the literature is the former, e.g. Fujimoto [16] used chains with $60,000$ iterations, where $15,000$ were discarded and the remaining were thinned in jumps of 3; while Fujimoto [15] used $225,000$ iterations, with burn-in of $30,000$ and thinning with jumps of 15. Among the drawbacks of this solution are the large computational times; the user involvement in deciding the specific

[1]Intervention designed with the purpose of potentiating specific abilities in teachers that are currently part of the public teaching career.

setting for the process, which could be different for different parameters in the same model; and finally, the lack of confidence that larger chain iterations actually produce a proper posterior investigation, which in turn requires the user to refit the model multiple times [16]. On the other hand, several authors have developed high-tech MCMC algorithms that aim to optimize their performance within a particular class of models [52]. In these cases, the developers re-evaluate not only the use of the programming language, with the purpose of speeding and improving performance (e.g. Fujimoto [16]); but also the inclusion of ad-hoc model assumptions, like prior conjugancy[2] for specific parameters, predetermined transformations of the linear predictor under non-continuous outcomes, or the use of highly regularizing priors, just to mention a few. Examples of this solutions are in staple software developments like Mplus [48] or Stata [57]. It is clear from the previous that this solution is not accessible to all researchers, either because of the lack of programming skills, or the restrictive cost of access involved in acquiring the software. But more importantly, these solutions are not always applicable to a wider framework of similar models [52].

In the second category, re-adjusting the Bayesian model, we also find two proposed solutions: (a) re-write the model in an alternative parametrization, and (b) encode prior information through the prior distributions, i.e. use regularizing priors. On both solutions, the purpose is to ensure the identification of the parameters within the model, which helps to stabilize the MCMC procedure [22]. An example of the former is Fujimoto [16], who decomposed the items' discriminatory parameters into overall and specific item discriminations. For the latter, Fujimoto [17] used informative priors also for the items' discrimination parameters.

More often than not, researchers use two or more of the aforementioned solutions to reach an acceptable performance in the chains. However, as pointed out by Betancourt and Girolami [6], even the most simple hierarchical models present formidable pathologies, that no simple correction can be performed to visit the posterior distribution properly. This is true no matter the rotation/rescaling of the parameter, or the amount of data. In this context, several authors [18, 19, 51, 52, 6] showed that prior information can be included in the model, not only through the prior distributions, but also by encoding it in the model itself, changing the posterior sampling geometries, i.e. removing the dependence of the parameters on other sampled parameters, therefore favoring the performance of MCMC chains.

Given all of the above, the present research will focus on showing how easy it is to account for all of the dependencies that educational data often display, under the GLLAMM framework. Furthermore, given that only the literature related to gaussian hierarchical models have shown the benefits of changing the posterior sampling geometries, through the use of the non-centered parameterization [18, 19, 51, 52, 6], it seems sensible to provide a similar assessment for nonlinear hierarchical models, and in particular, the ones with latent stochastic processes like IRT models [52]. Finally, the research will apply the newfound knowledge to data coming from a large Teacher's standardized educational assessments from Peru.

[2] when the prior and posterior distribution belong to the same parametric family.

1.2 Objectives

As mentioned in the previous section, the present research has a three-fold purpose:

1. Motivate the Bayesian GLLAMM for binary outcomes [56, 58, 65, 59]. The representation will emphasize the modeling of multiple hierarchical latent structures and testlets. This, in turn, will effectively blur the division between the GLLAMM framework and IRT models.

2. Empirically evaluate the benefits of changing the posterior sampling geometry, in the context of the previous model. The emphasis here will be on comparing the centered and non-centered parametrizations [18, 19] in terms of performance of the chains, the parameter's recovery capacity, the retrodictive accuracy, and the ability of the model to produce appropriate inferences.

3. Apply the model and its parametrization to a real data setting. Here the emphasis will be to assess the conclusions arrived from the application of the model, and what they could imply for the educational authorities.

Given the aforementioned goals, the researcher believes the master's thesis contributes to the literature in two aspects:

1. In a theoretical and methodological sense, as the research is focused on describing a model that effectively controls for the multiple dependencies, usually observed in educational data sets; and

2. In a more practical sense, as the study will provide empirical evidence if changing the sampling posterior geometries could (could not) benefit the performance of MCMC methods and therefore the inferences, under IRT models.

Finally, it is important to mention, the computational implementation of the method will be developed in `Stan` [67] and `R` [55, 66].

1.3 Organization

Chapter 2 will motivate the GLLAMM for dichotomous outcomes, and define its components. Chapter 3 will describe the Bayesian framework, its benefits and shortcomings. Furthermore, it will outline the evidence behind the change in posterior sampling geometries, and the computational implementation of the model. Chapter 4 will show the results of an empirical simulation study designed to assess the benefits of the re-parametrization, proposed in the previous chapter. Chapter 5 will describe the instruments, the data collection process, and scales under analysis, for a large standardized educational assessment. In addition, the chapter will show the conclusions achieved by the application of the model in the said data. Finally, Chapter 6, will discuss the conclusion for the research, and it will outline the path of future research topics that can be derived from the present effort.

Chapter 2

The GLLAMM for dichotomous outcomes

The Generalized Linear Latent and Mixed Model (GLLAMM) is a framework that unifies a wide range of latent variable models. Developed by Rabe-Hesketh and colleagues [56, 58, 57, 65, 59], the method was motivated by the need of a Multilevel Structural Equation Model (MSEM) that accommodated for unbalanced data, noncontinuous responses and cross-level effects among latent variables. The authors focused its development mainly from the frequentist perspective, however, they offered a general guidance on implementing the model under the bayesian framework (see Skrondal and Rabe-Hesketh [65]).

2.1 Model motivation

Consider a large standardized assessment composed of three sub-test designed to evaluate the reading comprehension, mathematical reasoning, and pedagogical knowledge of teachers; where each sub-test has several dichotomously scored items.

Focusing on the first sub-test, the items were designed to measure only one of the three hierarchically nested sub-dimensions of reading comprehension: literal, inferential, and reflective abilities. Furthermore, it is assumed the three sub-dimensions are all that is needed to measure the reading comprehension ability, effectively making this scale, the highest level latent variable in the model, similar to a hierarchical Confirmatory Factor Analysis (CFA). Finally, the items were bundled in groups of five to a common text or passage, i.e. testlets, that provided the stimulus over which the individual is assessed. Figure 2.1 shows the path diagram of the hypothesized dimensional structure, for the hierarchical cross-classified IRT model corresponding with the instrument. The figure represents the responses of one individual on 12 items.

With the purpose of providing an easier motivation of the model, we will not consider yet the cluster effects; however, later in the presentation we will show how easy it is to introduce them in the model. Just for future reference, under this example, one expects to observe clustering effects, because individuals from different regions did not have the same educational opportunities, effectively causing differences among them at a regional level.

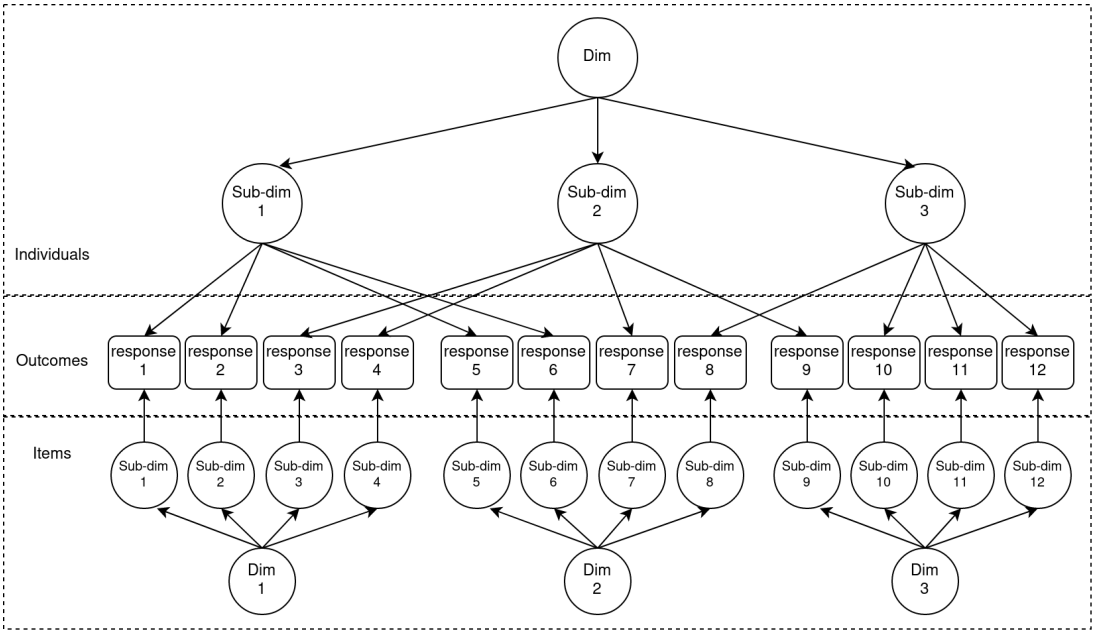

Figure 2.1: Path diagram of the dimensional structure for a hierarchical cross-classified IRT model. Squares represent dichotomous manifest variables, and circles represent latent variables. The figure is based on a reduced set of items, while the errors and scales of the latent variables are not represented. Different sub-dimensions at the individuals block represent the literal, inferential and reflective abilities, while at the items blocks represent the items' difficulties. The dimensions at the individuals block represent the reading comprehension ability, while at the items block represent the multiple testlets.

2.2 Model definition

Following Rabe-Hesketh et al. [56, 58], we continue defining the GLLAMM in two parts: (i) the response model, and (ii) the latent structure.

In case the reader is interested in outcomes different than the dichotomous case, refer to Rabe-Hesketh et al. [56, 58, 57], Skrondal and Rabe-Hesketh [65], and Rabe-Hesketh et al. [59].

2.2.1 Response model

Conditional to all regression parameters $\boldsymbol{\beta}$, loadings $\boldsymbol{\Lambda}$, latent variables $\boldsymbol{\Theta}$, and structural parameters $\boldsymbol{\Psi}$ and $\boldsymbol{\Gamma}$, i.e. $\boldsymbol{\Omega} = \{\boldsymbol{\beta}, \boldsymbol{\Lambda}, \boldsymbol{\Theta}, \boldsymbol{\Psi}, \boldsymbol{\Gamma}\}$; and the "stacked" vector of covariates for the first level ($\mathbf{X}$) and the structural part ($\mathbf{W}$); the response model can be represented by a Generalized Linear Model (GLM) [50, 44] with a distributional and a systematic part. The latter is composed of a linear predictor and a link function.

For the distributional part, the outcome y_{jkd} is modeled at the first level (level-1) by a Bernoulli probability mass function $f(\cdot)$, in the following form:

$$f\left(y_{jkd} = 1 \mid \mathbf{X}, \mathbf{W}, \boldsymbol{\Omega}\right) = \pi_{jkd}^{n}(1 - \pi_{jkd})^{1-n} \tag{2.1}$$

where individuals are indexed by $j = 1, \ldots, J$, with J representing the total number of individuals in the sample; the items are indexed by k, with d being the dimension the

items are set to measure; and n denotes the endorsement of the item in the Bernoulli trial. On the other hand, for the systematic part, the probability of endorsing the item π_{jkd} is linked to a linear predictor v_{jkd} through an inverse-link function $h(\cdot)$, in the following form:

$$P\left(y_{jkd}=1 \mid \mathbf{X}, \mathbf{W}, \mathbf{\Omega}\right)=\pi_{jkd}=h(\tau_k+v_{jkd}) \tag{2.2}$$

where τ_k is k'th item threshold, assumed to be zero for the binary case [56], while the inverse-link function can be defined in three ways:

$$h(x)=\begin{cases}\exp(x)[1+\exp(x)]^{-1} \\ \Phi(x) \\ \exp(-\exp(x))\end{cases} \tag{2.3}$$

corresponding to the logistic, standard normal $\Phi(x)$, and Gumbel (extreme value type I) cumulative distributions, respectively. It is usual to report the last in terms of link functions $g(\cdot)=h^{-1}(\cdot)$. In that case, these corresponds to the well known logit, probit and complementary log-log link functions, respectively. Finally, the linear predictor is defined by:

$$v_{jkd}=\sum_{p=1}^{P} x_{jp}\beta_p+\sum_{m=2}^{M+1}\sum_{k=1}^{K_{(m)}}\eta_k^{(m)}\alpha_k^{(m)}+\sum_{l=2}^{L+1}\sum_{d=1}^{D_{(l)}}\theta_{jd}^{(l)}\lambda_d^{(l)} \tag{2.4}$$

where β_p denotes regression parameter for the x_{jp} explanatory variable with $p=1,\ldots,P$, and P denoting the total number of level-1 response explanatory variables, e.g. time to answer the item, the number of alternatives, among others. $\eta_k^{(m)}$ is the kth item latent dimension at level m with loading $\alpha_k^{(m)}$, where $k=1,\ldots,K_{(m)}$, $K_{(m)}$ denotes the number of dimensions at level $m=2,\ldots,M+1$, and M represents the number of levels in the items block. It is important to keep in mind, the model also contemplates that not all the item are set to measure all the individual's dimension; however, we decided to not show such information in the form of an index in equation (2.4), to avoid making the notation heavier. Nevertheless, equation (2.5) is more clear on this detail, as the design block matrices specification show. Furthermore, $\theta_{jd}^{(l)}$ denotes the individual's dth latent dimension at level l with loading $\lambda_d^{(l)}$, where $d=1,\ldots,D_{(l)}$, $D_{(l)}$ represents the number of dimensions at level $l=2,\ldots,L+1$, and L denotes the number of levels in the individuals block. Notice that since the responses are modeled at the first level, the hierarchies for the latent dimensions need to start from the second level onward.

Additionally, equation (2.4) can be re-written in matrix form in the following way:

$$v_{jkd}=\mathbf{X}_j\,\boldsymbol{\beta}+\sum_{m=2}^{M+1}\boldsymbol{\eta}^{(m)}\boldsymbol{\alpha}^{(m)}\mathbf{A}_j^{(m)}+\sum_{l=2}^{L+1}\boldsymbol{\theta}_j^{(l)}\boldsymbol{\lambda}^{(l)}\mathbf{B}_j^{(l)} \tag{2.5}$$

where $\mathbf{X}_j$ represents the individual's design matrix of explanatory variables that maps the parameter vector $\boldsymbol{\beta}$ to the linear predictor; and $\mathbf{X}=[\mathbf{X}_1^T,\ldots,\mathbf{X}_J^T]^T$ the "stacked" design matrix of $\mathbf{X}_j$. Moreover, $\boldsymbol{\eta}^{(m)}=[\eta_1^{(m)},\ldots,\eta_{K_{(m)}}^{(m)}]^T$, and $\boldsymbol{\alpha}^{(m)}=[\alpha_1^{(m)},\ldots,\alpha_{K_{(m)}}^{(m)}]^T$ are the vectors of the item's latent dimensions with corresponding loadings at level m, mapped

by a block matrix $\mathbf{A}_j^{(m)}$. Similarly, $\boldsymbol{\theta}_j^{(l)} = [\theta_{j1}^{(l)}, \ldots, \theta_{jD_{(l)}}^{(l)}]^T$, and $\boldsymbol{\lambda}^{(l)} = [\lambda_1^{(l)}, \ldots, \lambda_{D_{(l)}}^{(l)}]^T$ are the vectors of the individual's latent dimensions with corresponding loadings at level l, mapped by a block matrix $\mathbf{B}_j^{(l)}$.

Finally, in order to have a more concise representation of the model, we can re-express equation (2.5) in the following way:

$$\begin{aligned} v_{jkd} &= \mathbf{X}_j\,\boldsymbol{\beta} + \boldsymbol{\eta}\,\boldsymbol{\alpha}\,\mathbf{A}_j + \boldsymbol{\theta}\,\boldsymbol{\lambda}\,\mathbf{B}_j \\ &= \mathbf{X}_j\,\boldsymbol{\beta} + \boldsymbol{\Theta}\,\boldsymbol{\Lambda}\,\mathbf{H}_j \end{aligned} \tag{2.6}$$

where $\boldsymbol{\alpha} = [\boldsymbol{\alpha}^{(2)T}, \ldots, \boldsymbol{\alpha}^{(M+1)T}]^T$ and $\boldsymbol{\lambda} = [\boldsymbol{\lambda}^{(2)T}, \ldots, \boldsymbol{\lambda}^{(L+1)T}]^T$ represent all the loadings corresponding to the items and individuals dimensions at all levels; whereas $\boldsymbol{\eta} = [\boldsymbol{\eta}^{(2)T}, \ldots, \boldsymbol{\eta}^{(M+1)T}]^T$ and $\boldsymbol{\theta} = [\boldsymbol{\theta}^{(2)T}, \ldots, \boldsymbol{\theta}^{(L+1)T}]^T$ represent the latent dimensions and sub-dimensions of items and individuals at all levels, respectively. Consequently, $\mathbf{A}_j$ and $\mathbf{B}_j$ are their mapping block matrices. Furthermore, $\boldsymbol{\Lambda} = [\boldsymbol{\alpha}^T, \boldsymbol{\lambda}^T]^T$ and $\boldsymbol{\Theta} = [\boldsymbol{\eta}^T, \boldsymbol{\theta}^T]^T$ to represent the "stacked" vector of loadings, and dimensions, respectively; and $\mathbf{H}_j$ its mapping block matrix.

To ground the excess of notation onto an example, we can use figure 2.1 as reference. In that case, we would have an empty level-1 covariates matrix $\mathbf{X}_j$, as we do not have any response explanatory variable ($P = 0$). Moreover, we have $M = 2$ levels at the items block, with $K_2 = 12$ and $K_3 = 3$. Consequently, $\boldsymbol{\eta}^{(2)} = [\eta_1^{(2)}, \ldots, \eta_{12}^{(2)}]^T$ and $\boldsymbol{\eta}^{(3)} = [\eta_1^{(3)}, \eta_2^{(3)}, \eta_3^{(3)}]^T$ denotes the items' difficulties and testlet effects, respectively. As stated in previous paragraphs, to avoid the use of even heavier notation, the item's indexes do not reflect that not all the items are set to measure all the individual's dimensions; however, the reader needs to keep that in mind. On the other hand, we have $L = 2$ levels in the individuals block, with $D_2 = 3$ and $D_3 = 1$. Therefore, $\boldsymbol{\theta}^{(2)} = [\theta_1^{(2)}, \theta_2^{(2)}, \theta_3^{(2)}]^T$ which denotes the literal, inferential and reflective abilities; whereas $\boldsymbol{\theta}^{(3)} = \theta_1^{(3)}$ represent the reading comprehension latent variable. Furthermore, since we are trying to express an IRT model, it makes sense to put some restrictions to the parameter set. In this case, we establish $\boldsymbol{\alpha}^{(2)} = -\boldsymbol{\lambda}^{(2)}$ with $\boldsymbol{\lambda}^{(2)} = [\lambda_1^{(2)}, \ldots, \lambda_{12}^{(2)}]^T$ representing the item's discriminatory parameter, where $\boldsymbol{\lambda}^{(2)} > \mathbf{0}$. This resemble to a multidimensional generalization of the linear predictor observed in the archetypical Rasch [60], or 2PL [40] models, i.e. $\lambda_d^{(2)}(\theta_{jd}^{(2)} - \eta_k^{(2)})$, where $|\lambda_d^{(2)}| = |\alpha_k^{(2)}|$ denotes the discriminatory power of item k, $\eta_k^{(2)}$ its difficulty, and $\theta_{jd}^{(2)}$ the ability of the individual at dimension d. In addition, $\boldsymbol{\lambda}^{(3)} = [\lambda_1^{(3)}, \lambda_2^{(3)}, \lambda_3^{(3)}]^T$ would represent the loadings from reading comprehension to their respective sub-dimensions; whereas $\boldsymbol{\alpha}^{(3)} = [\alpha_{11}^{(3)}, \ldots, \alpha_{15}^{(3)}, \alpha_{21}^{(3)}, \ldots, \alpha_{25}^{(3)}, \alpha_{31}^{(3)}, \ldots, \alpha_{35}^{(3)}]^T$ would represent the item-specific loadings from the testlets, usually set as $[1, \ldots, 1]^T$, indicating they explain directly the items difficulties at the lower level.

Finally, notice that through the use of $\mathbf{A}_j$ and $\mathbf{B}_j$ design block matrices, and more generally with $\mathbf{H}_j$, the model departs from the traditional multivariate framework for formulating structural models, i.e. a wide data format; and adopts a univariate approach, i.e. a long data format. The former stores the subject's repeated outcomes in a single row, with multiple response vectors and explanatory variables appended column-wise to the outcome data. The later stores the subject's repeated outcomes in a single "stacked" response vector with as many rows as there are repeated measurements, and explanatory variables appended column-wise to the outcome data, distinguished from each other, by a design block matrix.

Cluster effects

Considering the previous, we can see that modeling individual clustering just involves the addition of more random effects, to the linear predictor defined in equation (2.4):

$$\begin{aligned} v_{jkdc} &= v_{jkd} + \sum_{c=1}^{C} \delta_c \\ &= v_{jkd} + \boldsymbol{\delta}\mathbf{Z}_j \end{aligned} \tag{2.7}$$

where $c = 1, \ldots, C$, which denotes the number of clusters, v_{jkd} is defined as in equation (2.4), and $\mathbf{Z_j}$ is a design block matrix.

2.2.2 Latent structure

The structural model for the latent variables is represented in the following form:

$$\underset{(S\times 1)}{\boldsymbol{\Theta}} = \underset{(S\times S)}{\boldsymbol{\Psi}}\ \underset{(S\times 1)}{\boldsymbol{\Theta}} + \underset{(S\times Q)}{\boldsymbol{\Gamma}}\ \underset{(Q\times 1)}{\mathbf{W}} + \underset{(S\times 1)}{\boldsymbol{\zeta}} \tag{2.8}$$

where $S = K + D$, $K = \sum_m K_m$, and $D = \sum_l D_l$. Furthermore, $\boldsymbol{\Psi}$ and $\boldsymbol{\Gamma}$ are parameter matrices that map the relationship between the latent variables $\boldsymbol{\Theta}$, and the "stacked" vector of covariates $\mathbf{W}$, respectively; while $\boldsymbol{\zeta}$ is a vector of errors or disturbances. It is important to indicate that $\mathbf{W}$ considers a different set of covariates from $\mathbf{X}$, as we hypothesize they explain the variability in the latent variables at different levels.

Notice equation (2.8) is the generalization of a single-level Structural Equation Models (SEM) to a multilevel setting. However, the main difference in the GLLAMM representation is that the latent variables may vary at different levels.

Additionally, considering that $\boldsymbol{\Theta}$ has no feedback effects, is permuted, and sorted according to the levels of interest, then $\boldsymbol{\Psi}$ will be a strictly upper triangular matrix. In this regard, (i) the absence of feedback loops imply the method deals with non-recursive models, i.e. none of the latent variables are specified as both causes and effects of each other [36]; and (ii) the strictly upper triangular structure reveals the GLLAMM does not allow latent variables to be regressed on lower level latent or observed variables, implying the method deals with reflective measurement[1][5]. For a more detailed explanation on the topic see Edwards and Bagozzi [12].

However, notice that because in the IRT framework $\boldsymbol{\eta}$ and $\boldsymbol{\theta}$ should be orthogonal to each other by design, we can further decompose equation (2.8) in the following form:

$$\underset{(K\times 1)}{\boldsymbol{\eta}} = \underset{(K\times K)}{\boldsymbol{\Psi}_\eta}\ \underset{(K\times 1)}{\boldsymbol{\eta}} + \underset{(K\times Q)}{\boldsymbol{\Gamma}_\eta}\ \underset{(Q\times 1)}{\mathbf{W}_\eta} + \underset{(K\times 1)}{\boldsymbol{\zeta}_\eta} \tag{2.9}$$

$$\underset{(D\times 1)}{\boldsymbol{\theta}} = \underset{(D\times D)}{\boldsymbol{\Psi}_\theta}\ \underset{(D\times 1)}{\boldsymbol{\theta}} + \underset{(D\times Q)}{\boldsymbol{\Gamma}_\theta}\ \underset{(Q\times 1)}{\mathbf{W}_\theta} + \underset{(D\times 1)}{\boldsymbol{\zeta}_\theta} \tag{2.10}$$

where $\boldsymbol{\Psi}_\eta$, $\boldsymbol{\Psi}_\theta$, $\boldsymbol{\Gamma}_\eta$, and $\boldsymbol{\Gamma}_\theta$ are the dimension-specific parameter matrices that map the relationship between the corresponding latent variables $\boldsymbol{\eta}$ and $\boldsymbol{\theta}$; with $\boldsymbol{\zeta}_\eta$ and $\boldsymbol{\zeta}_\theta$ being the dimension-specific vector of disturbances. Finally, the matrices $\mathbf{W}_\eta$ and $\mathbf{W}_\theta$ would be the dimension-specific covariates.

[1] A latent variable is considered reflective when it is thought to be the cause of the lower level latent or manifest variables.

Considering figure 2.1 as reference, we have structural relationships $\mathbf{\Psi}$ (as described in the previous section); but we do not declare any additional covariates $\mathbf{W}$. Chapter 4 and 5 will show an implementation where we hypothesized a set of covariates explain the variability on some of the latent variables.

2.3 Model assumptions

Following Skrondal and Rabe-Hesketh [65], the framework has two main assumptions: (i) complete latent space, and (ii) local or conditional independence.

(M1) Complete latent space. The latent space is considered complete if all the latent variables, that we hypothesize affect the outcomes, are considered in the model [26]. In the GLLAMM representation, these would be all latent variables $\mathbf{\Theta}$ at levels $l > 1$ and $m > 1$, as the outcomes are modeled at the first level.

(M2) Local Independence, also known as conditional independence, is defined in the following form:

$$f\left(\mathbf{y} = \mathbf{1} \mid \mathbf{X}, \mathbf{W}, \mathbf{\Omega}\right) = \prod_{j=1}^{J} \prod_{d=1}^{D} \prod_{k=1}^{K} f\left(y_{jkd} = 1 \mid \mathbf{X}, \mathbf{W}, \mathbf{\Omega}\right) \tag{2.11}$$

where $f\left(\mathbf{y} = \mathbf{1} \mid \mathbf{X}, \mathbf{W}, \mathbf{\Omega}\right)$ denotes the level-1 conditional likelihood, and $\mathbf{y}$ the vector of all responses at the first level.

Notice the approach also assumes local or conditional independence, as their non-hierarchical IRT model counterparts. Nevertheless, its independence is conditional on all the latent dimensions and covariates, at different hierarchical levels; effectively modeling all the observed dependencies.

Finally, it is important to point out that equation (2.11) results from the union of two more specific assumptions, local item and individual independence [62, 3, 27]:

(a) **Local item independence,** which entails the individual's response to an item does not affect the probability of endorsing another item, after conditioning on the individual's ability. This is expressed in the following mathematical form:

$$f\left(y_{j..} = 1 \mid \mathbf{X}, \mathbf{W}, \mathbf{\Omega}\right) = \prod_{d=1}^{D} \prod_{k=1}^{K} f\left(y_{jkd} = 1 \mid \mathbf{X}, \mathbf{W}, \mathbf{\Omega}\right) \tag{2.12}$$

where $y_{j..} = [y_{j11}, \ldots, y_{jKD}]$ is the vector of all items for individual j and, as previously defined, $K = \sum_m K_m$ and $D = \sum_l D_l$.

(b) **Local individual independence,** which entails that an individual's response to an item is independent of another person's response to that same item. The assumption is expressed in the following form:

$$f\left(y_{.kd} = 1 \mid \mathbf{X}, \mathbf{W}, \mathbf{\Omega}\right) = \prod_{j=1}^{J} f\left(y_{jkd} = 1 \mid \mathbf{X}, \mathbf{W}, \mathbf{\Omega}\right) \tag{2.13}$$

where $y_{.kd} = [y_{1kd}, \ldots, y_{Jkd}]$ is the vector of all individuals endorsing item k from dimension d.

Chapter 3

Bayesian estimation

The practical use the GLLAMM developed in chapter 2 requires the estimation of all the items and individuals' dimensions, loadings, regression and structural parameters. These can be obtained within two frameworks: the frequentist and bayesian.

The current chapter center its attention on describing the bayesian estimation procedure, using the Markov Chain Monte Carlo method (MCMC). For a full development of GLLAMM under the frequentist framework refer to Rabe-Hesketh and colleagues [56, 58, 65, 59].

3.1 Benefits and shortcomings

3.1.1 Why Bayesian?

The reasons on why bayesian statistics is attractive to perform the parameters' estimation of any model, and especially for the GLLAMM developed in chapter 2, are:

1. It is built on a simulation-based estimation method, therefore, it can handle all kinds of priors and data-generating processes [14]. This is especially useful with highly complex and over-parameterized models, where other methods are unfeasible or work poorly [2, 35].

2. While the likelihood functions are used to define the posterior sampling distributions, they can also be used in a generative way. The likelihood for the data, and priors for the parameters, form the basis to produce samples from the posterior distribution. However, they can also be used to simulate observations, allowing us to test the ability of the method/data to recover the parameters of interest [45].

3. The bayesian estimates are at least as good as its frequentist counterparts [2, 72, 30]. This is true when the method uses uninformative 'flat' priors. However, because the procedure allow us to integrate prior knowledge about the parameters, beyond the observed responses, it can produce results even in scenarios where the Maximum Likelihood methods (ML) have issues of non-convergence or improper estimation [65, 14, 45]. Examples of such are:

   (a) when we have small sample sizes.

(b) when individuals have null scores or aberrant response patterns [27, 1]. The latter happens when examinees answer some relatively difficult and discriminating items correctly, while answering some of the easiest incorrectly.

(c) when parameters need to be confined to a permitted space, e.g. the estimation of positive unique factors variances [43].

(d) when we need to estimate parameters under sparse data, where the asymptotic theory is unlikely to hold [14];

3.1.2 Are you sure there's nothing wrong?

Off course the bayesian framework has shortcomings, among them:

1. It exposes the user to somewhat-arbitrary decisions about the running of the chains, in order to ensure a proper performance, e.g. how many iterates does the chain need to achieve precise estimates?, what is the right size for the burn-in and warm-up phases?, how should the thinning procedure be performed, if any?, should we follow the same procedure for all parameters of interest?, among others [65].

2. The user can include all type of information through the priors distributions, making their elicitation convenient for manipulation.

3. Multiple options are available to evaluate the performance of the method, and most of them are visual, making it hard to assess if a proper posterior investigation have been made [23]. More specifically, the user has multiple options to assess if the chain achieves the three requirements of a good performance: stationarity, convergence, and good mixing [45].

4. The procedure makes it hard to discover parameters' lack of identification [65]. Inadequate mixing of the chain could lead us to think unidentified parameters have been estimated with precision, when in fact what we have are 'flat' posteriors [33].

5. Oftentimes the posterior sampling geometry of the model makes it hard to find proper solutions for the parameter space, no matter the rotation/rescaling of the parameter, or the amount of data [6]. This is especially true in complex hierarchical models.

6. The greater the complexity of the model, the harder it is to communicate/share the implementation with other scientist. This is especially true, when researcher re-parameterize the model to solve the previous shortcoming [45].

7. The procedure usually requires more time to achieve a proper solution, compared to the classical methods. This is especially true in models with high complexity [68, 63].

Although some of the previous shortcomings have made the Bayesian procedure a "controversial" implementation, most of them already have acceptable solutions.

For the first point, a popular approach is to use a large number of iterates, burn-in and thinning processes. This is mostly applicable under the Metropolis-Hastings and Gibbs sampling algorithms. However, recent solutions, like the Hamiltonian Monte Carlo

(HMC) [6], is less reliant on selecting these features, as it implements a different sampling mechanism (see section 3.3.1).

On the second point, the literature has indicated that priors can be used to reflect subjective beliefs[1]. However, as McElreath [45] indicated, because the priors can be considered a part of the model assumptions, they can be chosen, evaluated, and revised as any other component of the model, through the use of prior predictive simulations and/or sensitivity analysis (see section 3.3.5).

About the third point, the literature on bayesian analysis acknowledge that the visual assessment of stationarity and convergence is easier, and these procedures usually has additional support from statistics like `Rhat` [22]. On the contrary, a visual evaluation of 'good' mixing remains as a hard task. Recent approaches have taken a more proactive stance on the latter, seeking to increase the possibility of a well mixed chain by changing the posterior sampling geometry of the model [51, 52, 6, 45] (see section 3.4).

On the fourth point, the most common solution is to use regularizing priors, i.e. priors that are more 'skeptical' of wider parameter spaces [45]. However, it is important to mention, there are scenarios where one achieves poor parameter estimates, even in the presence of 'enough' data and regularizing priors, but this shortcoming is also applicable to the classical estimation procedures, e.g. the estimation of the variance parameters in random effects models [65].

About the fifth point, as it was mentioned in previous paragraphs, a recent approach to solve the issue is to change the posterior sampling geometry of the model. This means to re-parameterize the model in a way that removes the dependence of the parameters on other sampled parameters; or even, produce sample mechanisms located in a continuous between a centered (CP) and non-centered parametrization (NCP), e.g. Interleaved HMC (iHMC) and/or Variational Inference (VI) [18, 19, 51, 52, 6, 25] (see section 3.4).

Finally, the sixth and seventh points can be considered the 'price' a scientist has to pay, to be able to fit models that conforms better with the observed data generating processes. Although, recent developments are striving to improve upon reducing the latter, i.e. the required running time.

3.2 Bayesian GLLAMM for dichotomous outcomes

3.2.1 Posterior distribution

Denoting $\mathbf{Y}$ as the observed data and $\mathbf{\Omega} = \{\boldsymbol{\beta}, \mathbf{\Lambda}, \mathbf{\Theta}, \mathbf{\Psi}, \mathbf{\Gamma}\}$, i.e. all the parameters declared in section 2.2.1 and 2.2.2, the posterior distribution is obtained using the Bayes theorem, in the following way:

$$P(\mathbf{\Omega} \mid \mathbf{Y}) = \frac{P(\mathbf{Y} \mid \mathbf{\Omega})\ P(\mathbf{\Omega})}{\int P(\mathbf{Y} \mid \mathbf{\Omega})\ P(\mathbf{\Omega})\ d\mathbf{\Omega}} \tag{3.1}$$

this is possible since the Bayesian approach makes no distinction between latent variables and parameters. All of them are considered random quantities [65]. Furthermore, since inference only requires representing the likelihood of the data $P(\mathbf{Y} \mid \mathbf{\Omega})$ and the prior

[1] see McElreath [45] (Chapter 2, pp. 36), Rethinking section "Prior, prior pants on fire", for an interesting stance contrary to this statement.

distribution $P(\boldsymbol{\Omega})$; and because the denominator is just a (hard to calculate) constant, the posterior can be determined up to a normalizing constant, without loss of generality:

$$P(\boldsymbol{\Omega} \mid \mathbf{Y}) \propto P(\mathbf{Y} \mid \boldsymbol{\Omega})\ P(\boldsymbol{\Omega}) \tag{3.2}$$

3.2.2 Prior distributions

Similar to Patz and Junker [53], we use an independent distributional structure for the joint priors of the parameters. Therefore, at the highest level of the GLLAMM we have:

$$\begin{aligned} P(\boldsymbol{\Omega}) &= P(\boldsymbol{\beta}, \boldsymbol{\Lambda}, \boldsymbol{\Theta}, \boldsymbol{\Psi}, \boldsymbol{\Gamma}) \\ &= P(\boldsymbol{\beta})\ P(\boldsymbol{\Lambda})\ P(\boldsymbol{\Theta})\ P(\boldsymbol{\Psi})\ P(\boldsymbol{\Gamma}) \\ &= P(\boldsymbol{\beta})\ [P(\boldsymbol{\alpha})\ P(\boldsymbol{\lambda})]\ [P(\boldsymbol{\eta})\ P(\boldsymbol{\theta})]\ [P(\boldsymbol{\Psi}_\eta)\ P(\boldsymbol{\Psi}_\theta)]\ [P(\boldsymbol{\Gamma}_\eta)\ P(\boldsymbol{\Gamma}_\theta)] \end{aligned} \tag{3.3}$$

However, giving the hierarchical and cross-classified structure of the model, the lower level priors will also depend on further parameters[2], e.g. variances and covariances of the latent variables. These are known as *hyper-parameters*, and their prior distributions as *hyper-priors*. Because of the previous, we are not going to detail their lower level representation, as they are highly dependent on the specifics of the model.

Section 3.3.5 details the process of prior predictive investigation for prior elicitation. On the other hand, chapter 4 and 5 will show the elicitation of priors, in the context of a simulated and real data set, respectively.

3.2.3 Likelihood

Following Rabe-Hesketh et al. [56], the likelihood function is build in a recursive way. First, we replace the structural model (2.8) into the linear predictor (2.6):

$$v_{jkd} = \mathbf{X}_j\boldsymbol{\beta} + (\mathbf{I} - \boldsymbol{\Psi})^{-1}\left[\boldsymbol{\Gamma}\mathbf{W} + \boldsymbol{\zeta}\right]\ \boldsymbol{\Lambda}\ \mathbf{H}_j \tag{3.4}$$

Second, the linear predictor and inverse-link function (2.3) are used to construct the systematic part of the response (2.2), i.e. its expected value. Here we use a logit link for pedagogical purposes:

$$\pi_{jkd} = \frac{\exp(\tau_k + v_{jkd})}{1 + \exp(\tau_k + v_{jkd})} = \frac{\exp(v_{jkd})}{1 + \exp(v_{jkd})} \tag{3.5}$$

Third, the expected value is used in the distributional part (2.1), in the following form:

$$f\left(y_{jkd} = 1 \mid \mathbf{X}, \mathbf{W}, \boldsymbol{\Omega}\right) = \pi_{jkd}^{n}(1 - \pi_{jkd})^{1-n} \tag{3.6}$$

Fourth, considering assumptions (M1) and (M2) described in section 2.3, we produce the level-1 likelihood:

$$f\left(\mathbf{y} = \mathbf{1} \mid \mathbf{X}, \mathbf{W}, \boldsymbol{\Omega}\right) = \prod_{j=1}^{J}\prod_{d=1}^{D}\prod_{k=1}^{K} f\left(y_{jkd} = 1 \mid \mathbf{X}, \mathbf{W}, \boldsymbol{\Omega}\right) \tag{3.7}$$

[2]Because of this sequential model specification, it is said the priors also have a "hierarchical" structure.

Fifth, we define the marginal likelihood at levels l and m, conditional on the latent variables at levels $l+1$ and $m+1$, in the following form:

$$f_{(m)}^{(l)}\left(\mathbf{y}=\mathbf{1}\mid\mathbf{X},\mathbf{W},\boldsymbol{\Omega}\right)=\int\left[\prod f_{(m-1)}^{(l-1)}\left(\mathbf{y}=\mathbf{1}\mid\mathbf{X},\mathbf{W},\boldsymbol{\Omega}\right)\right]\;P(\boldsymbol{\Theta}_{(m)}^{(l)})\,d\boldsymbol{\Theta}_{(m)}^{(l)} \tag{3.8}$$

where the product inside the brackets is over all units in level $(l-1)$ and $(m-1)$, the first level is defined as in equation (3.7), and $P(\boldsymbol{\Theta}_{(m)}^{(l)})$ are the prior distributions for the latent variables at level l and m, with $\boldsymbol{\Theta}_{(m)}^{(l)}=[\boldsymbol{\eta}^{(m)},\boldsymbol{\theta}^{(l)}]$. Finally, we define the likelihood as:

$$\mathcal{L}(\mathbf{X},\mathbf{W},\boldsymbol{\Omega})=\prod_{m=2}^{M+1}\prod_{l=2}^{L+1}f_{(m)}^{(l)}\left(\mathbf{y}=\mathbf{1}\mid\mathbf{X},\mathbf{W},\boldsymbol{\Omega}\right) \tag{3.9}$$

and the log-likelihood as:

$$\ell(\mathbf{X},\mathbf{W},\boldsymbol{\Omega})=\log\mathcal{L}(\mathbf{X},\mathbf{W},\boldsymbol{\Omega}) \tag{3.10}$$

3.2.4 Model identification

As indicated by Rabe-Hesketh et al. [56], in order to fully specify the model and provide a scale for the latent variables, we have to make assumptions for either the distribution of the disturbances $\boldsymbol{\zeta}$, or the distribution of one or more of the latent variables $\boldsymbol{\Theta}$. On the other hand, as point out by Fujimoto [15] and several others authors, we could also set restrictions for one or more of the loadings in $\boldsymbol{\Lambda}$. The last is more prevalent in frequentist software packages like Winsteps [39].

Furthermore, as it is hinted by equation (3.9), it is assumed the latent variables at different levels are independent from each other, whereas latent variables at the same level may present dependency. Therefore, when the assumption is coherent under the context of a model, we will presume the latent variables at the same level have a multivariate normal distribution with a mean and covariance structure determined by equations (2.9) and (2.10). In the case of the covariance matrices, these are determined by the covariance of the specific disturbances $\boldsymbol{\zeta}$.

Chapter 4 and 5 will show the model identification strategy in the context of a simulated and real data set, respectively.

3.3 Computational implementation

3.3.1 Hamiltonian Monte Carlo

For years the state-of-the-art platform for Bayesian statistical modeling have been the BUGS[3] project, with its WinBUGS and OpenBUGS implementations [42, 41]. However, a more recent participant JAGS[4] [54] has gain traction, due to its cross-platform capabilities. In any case, both platforms are not that different form each other. Both use two of the most popular and successful algorithms for performing MCMC: the Metropolis-Hastings [46, 28] and Gibbs Sampling [24] algorithms.

[3]Bayesian inference Using Gibbs Sampling
[4]Just Another Gibbs Sampling

In general lines, the Metropolis algorithm follows a two-step procedure: (i) it produces a new proposal for a parameter's value, and (ii) it evaluates the proposal against the current, accepting the new value if it is more likely under the posterior distribution. On the other hand, the Gibbs sampling uses a similar procedure, but it gains efficiency by exploiting the knowledge behind the target distribution. The latter improvement catapulted the Gibbs sampler into the workhorse of high dimensional Bayesian computing.

However, both methods remain as highly random procedures, and because of it, they still have important limitations. The most important of them all is that they do not yield independent and identically distributed samples (*iid*). This in turn means, the exploration of the posterior is made through a series of (sometimes highly dependent) values, causing the MCMC chain to converge to the target distribution only in the long run, i.e. when the number of iterations approach infinity ($t \to \infty$) [22]. It is important to mention that while the non-fulfillment of *iid* samples is a critic of all simulation based method, in no algorithms is more relevant than in the previous two.

In this context, multiple approaches have been proposed to ensure an appropriate exploration of the posterior distribution, even in the face of non-*iid* samples, and oftentimes, the aforementioned software packages use them in conjunction. The first consist on improving the sampling mechanism by using a Hybrid MCMC method. A good example of this is the "Metropolis-within-Gibbs" algorithm [47]. The second consist on setting a warm-up phase, that seeks to adapt the proposal distributions to favor the exploration of the posterior. Finally, the third consist on defining the number of iterates, burn-in and thinning processes in a way, that it reduces the serial dependencies in the samples.

Nevertheless, the aforementioned solutions, while useful, do not solve all issues related to the MCMC chain's performance, as these can remain biased in subtle ways, that are more harder to identify [45]. Therefore, researchers have decided instead to propose new improved algorithms.

Is in this context were the Hamiltonian Monte Carlo or Hybrid Monte Carlo (HMC) was proposed by Duane et al. [11]. While the full representation of method is out of the scope of this research, we can still make a high-level description of its process, with purpose of point out its benefits and shortcomings.

Assuming we have continuous distributions and the partial derivatives of the log-density function (the gradient) exists, the method iterates between the updating of two vector spaces of a "particle" [49, 45]. First, it samples new values for the particle's momentum vector, independent of the current values of its position vector, i.e. it gives the particle a random "flick". In the second iteration, using Hamiltonian dynamics, a Metropolis update is performed to propose a new position vector for the particle. In this sense, one can say the HMC runs a small physics simulation, where the position of the particle denotes the current parameter values in the Markov chain, the momentum denotes the proposed path to a new set of values, and the log-posterior provides the "friction-less" surface over which the particle moves, where the gradient defines its curvature [45].

In principle, because HMC produce "more informed" proposals, it will accept all of them. In practice, however, HMC uses a rejection procedure that inform us when the method was not able to maintain the "energy" of the system, producing a bad numeric approximation, i.e. a divergent transition (see section 3.4 for an example). For more on the background theory on Hamiltonian Dynamics and specifics about the HMC algorithm refer to Neal [49], and Betancourt and Girolami [6].

All of this just tells us that HMC is a highly complex algorithm. However, as McElreath

[45] pointed out, the Gibbs strategy got its improvement over the Metropolis-Hastings by being less random, not more. And in that sense, HMC pushes this principle to a greater length, but manages to improve the efficiency of the MCMC procedure. This is especially true in cases where ordinary Metropolis or Gibbs sampling cannot make a proper exploration of the parameter space, e.g. with highly correlated parameters or with models with hundreds or thousands of parameters, like complex hierarchical models. Moreover, as HCM is more efficient, it requires less number of iterations to make a proper posterior investigation [45, 22]

Off course, this comes with a cost. The methods is more computationally intensive than the previous. However, because of the efficiency improvement and the need of less number of iterations, the method requires less computer time in total, even when each individual sample needs more.

3.3.2 Where can I find this magical software?

`Stan` [67] is the software package that will provide us with the machinery of HMC. Furthermore, the results produced from the software will be analyzed with `R` [55, 66], and its integration packages.

3.3.3 What about burn-in and thinning?

Since HMC uses a different approach and produces more informative proposals, setting specific burn-in and thinning processes are no longer necessary.

However, the method does require to perform a warm-up procedure to adapt sampling, and to tune in two parameters of the algorithm: the number of steps (`leapfrogs`), and the `step size` [67]. These parameters are used to make a discrete approximation of the momentum vector, i.e. the path of the particle [49, 6].

Notice, this phase is similar to a procedure performed in JAGS, where the proposal distribution gets "tuned-up" to improve the posterior investigation. However, the similarities end there. The warm-up samples in HMC are not representative of the target posterior distribution, no matter how long the iterations continue [45]. Finally, after the warm-up is finished, and assuming the adaption was successful, the method will sample directly from the target distribution.

In the current research, we will use a total $3,000$ effective iterations, coming from 3 chains of $2,000$ iterations each, where $1,000$ of them will be spend on warm-up, and no thinning procedure will be performed. Notice this departs (by far) from most of the literature on bayesian IRT models, e.g. Fujimoto [16] used chains with $60,000$ iterations, where $15,000$ were discarded and the remaining were thinned in jumps of 3; while Fujimoto [15] used $225,000$ iterations, with burn-in of $30,000$ and thinning with jumps of 15; just to mention a few.

3.3.4 Initial starts

Since the method requires starting values for the parameters, the author decided to allow the software to sample these from the priors defined in the model.

3.3.5 Prior elicitation

Multiple studies have pointed out the benefits of using mildly informative priors to reduce over-fitting, while allowing the model to learn regular features of the data, e.g. McElreath [45], Gelman et al. [21] and Jaynes [31]. In contrast, the literature has emphasized the unintended consequences on the posterior, resulting from using highly uninformative priors, e.g. Seaman et al. [64], and Gelman [20].

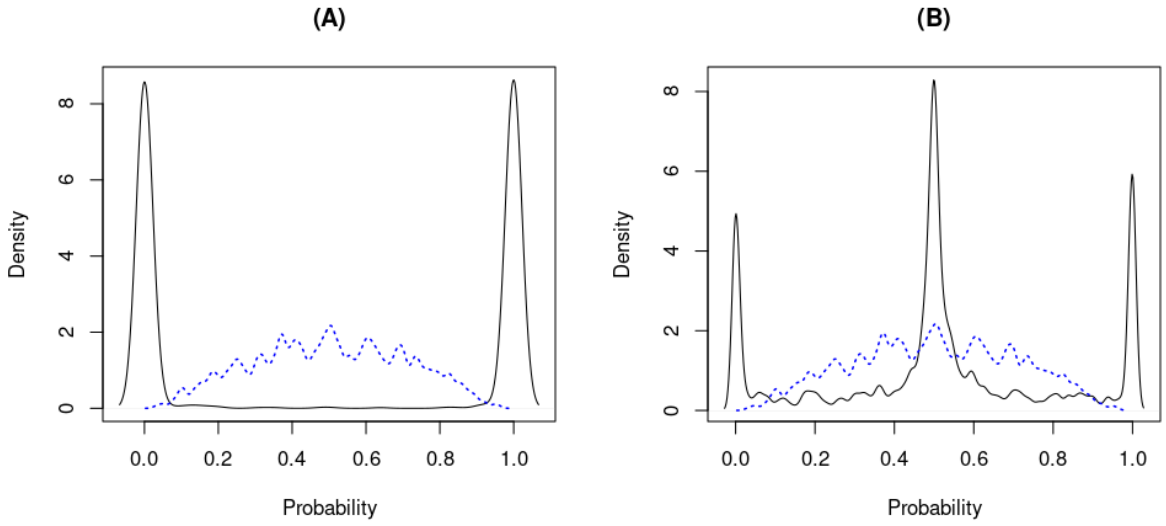

Figure 3.1: Prior predictive simulation. Examples of uninformative and mildly informative priors. The panels show the density for the probability of a binary outcome under: (A) uninformative prior on equation (3.11) and mildly informative prior on equation (3.13); and (B) uninformative prior on equation (3.12) and mildly informative prior on equation (3.14). Uninformative (black continuous lines), and mildly informative (blue discontinuous lines).

In our case, as stated in section 3.1.2, because priors can be considered a part of the model assumptions, they can be chosen, evaluated, and revised as any other component of the model [45]. In that sense, we motivate the use of mildly informative priors though a simple illustration. Consider an example similar to one outlined by Seaman et al. [64] and McElreath [45], where a simple latent variable model is defined as follows:

$$\begin{aligned} \theta &\sim N(0, 100) \\ \text{logit}(p) &= \theta \end{aligned} \tag{3.11}$$

where θ denotes a latent dimension, and p denotes the probability of a binary outcome, defined by the logit-link function. For pedagogical purposes, we did not declared the distribution of the response, as we wanted to assess what the prior implied for its expected value/probability.

Panel (A) from figure 3.1 shows that the uninformative prior in equation (3.11) forces an unintended assumption onto the expected value of the outcome (black continuous line). The model is now highly 'skeptical' of probabilities that are not zero or one. This means

that we are effectively using highly regularizing priors, but opposite on how they are supposed to be used, i.e. to avoid visiting non-likely parameter spaces.

The scenario gets equally bad with the presence of hyper-parameters, and even less extreme assumptions. Consider the following example:

$$\begin{aligned} v &\sim \log N(0, 3) \\ \theta &\sim N(0, v) \\ \text{logit}(p) &= \theta \end{aligned} \tag{3.12}$$

where v is distributed as a log-normal distribution ($v > 0$). Panel (B) from figure 3.1 shows that the uninformative priors in equation (3.12) forces another unintended assumption on the expected value of the outcome (black solid line). The model is now highly 'skeptical' of probabilities that are not $p = \{0, 0.5, 1\}$. Again is like using regularizing priors but "in reverse".

Luckily, these extreme priors can be overcome with enough data. However, that does not discard the need to address the issue, as much of the time, researchers do not have a clear definition of how much is "enough" data.

In that sense, mildly informative priors can help alleviate the concern. Mildly/weakly informative priors are prior distributions that are not supplying any controversial information, are consistent with the likelihood, and are strong enough to pull the data away from inappropriate inferences [22]. To illustrate its use, consider a mildly informative prior on equation (3.11):

$$\begin{aligned} \theta &\sim N(0, 1) \\ \text{logit}(p) &= \theta \end{aligned} \tag{3.13}$$

Notice now θ has a prior that is consistent with common IRT applications, and its distribution does not force the expected value of the outcome to be as extreme, as in the uninformative case. Panel (A) from figure 3.1 shows the prior probability of the outcome can now be in the full $[0, 1]$ range, where extreme probability values are assumed less likely (blue discontinuous line vs black solid line).

Similarly, in the presence of hyper-parameters we can set mildly informative priors in the following form:

$$\begin{aligned} v &\sim \log N(0, 0.5) \\ \theta &\sim N(0, v) \\ \text{logit}(p) &= \theta \end{aligned} \tag{3.14}$$

where the blue discontinuous line on panel (B) of figure 3.1 shows its transformation into the expected value of the outcome.

One can imagine the procedure of prior elicitation will get more complicated with a larger number of parameter and hyper-parameters. Therefore, the use of prior predictive simulation will be a tool of high value for complex models, as in our current implementation. See section 4.5.3 for an example on the use of prior predictive investigation, under simulated data.

3.4 To center or not to center

As pointed out by Betancourt and Girolami [6], even the most simple hierarchical models present formidable pathologies, that no simple correction can be performed to visit the posterior distribution properly. This is true no matter the rotation/rescaling of the parameter, or the amount of data.

A great example of this, is what McElreath [45] dubbed the Devil's funnel (depicted in figure 3.3), where the author shows that you do not need a complex model to start observing these issues. Consider the following joint distribution:

$$\begin{aligned} v &\sim N(0, 3) \\ \theta &\sim N(0, \exp(v)) \end{aligned} \tag{3.15}$$

where v is the *hyper-parameter* of θ, and θ is dependent on the samples of v. The latter is what the literature dubbed as the centered parametrization (CP).

This joint distribution might seem familiar, as bayesian hierarchical model practitioners often use it to ensure the standard deviation remains into the permitted parameter space ($\sigma \geq 0$). Similar types of requirements permeate IRT models. Moreover, it seems that any MCMC procedure would not have any issues exploring the joint distribution of these parameters, as they are normally distributed and there are only two of them, but we would be wrong.

For pedagogical purposes, the example was run without data, as the author wanted to emphasize the pathologies are present even before we feed the data to our model. However, its easy to extrapolate that these issues will remain present when data is available. The reader can find the `stan` and `jags` implementations for these examples in Appendix B.1.1. The `stan` implementation replicates the example stated by Betancourt and Girolami [6] and McElreath [45].

Figure 3.2 show the chains resulting from implementing the model on equation (3.15), through HMC (`stan`). As one can see from the figure, the joint posterior distribution is not explored properly. The chains show no sign of achieving ergodicity [46], i.e. they do not show stationarity or convergence (top panels), nor a good mixing (middle and bottom panels). This is further supported by the `Rhat`, and effective sample sizes estimates `n_eff` [22]. The `Rhat` for all parameters did not properly approach the threshold of one (most of them were even above 1.05), indicating the chains did not achieve convergence: the between-chain variability was larger than the within-chain variability. Furthermore, the effective samples sizes were 34 and 293 for v and θ, respectively; indicating the values of the chains remained highly correlated (see bottom panels). This is striking, as one expects effective sample sizes closer to the effective number of iterations ($3,000$, coming from 3 chains with $1,000$ samples each, after warm-up).

3.4.1 Wasn't HMC the solution to this?

HMC is a powerful MCMC algorithm, and its benefits stand the test of other scenarios, as detailed by multiple authors [45, 22]. However, the issue with this example goes a little bit farther than what the algorithm can actually overcome, and this is true also for the Metropolis and Gibbs sampling algorithms. Nevertheless, we will see HMC still manages to be efficient within the constrains of the problem.

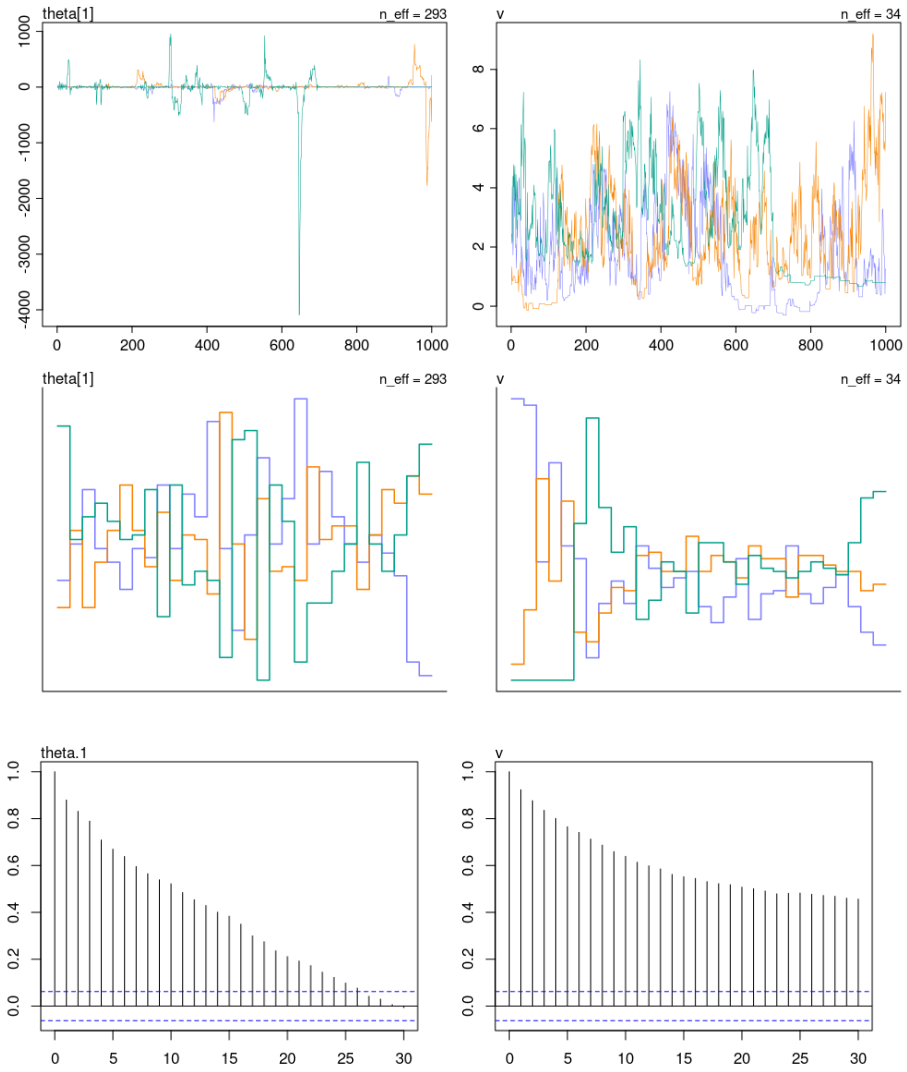

Figure 3.2: The Devil's funnel. Centered Parametrization. (Top) Trace plots for the iterations on three chains. (Middle) Trank plots for the same data. (Bottom) Autocorrelation Functions (ACF) plot for the iterations.

Figure 3.3 shows the posterior sampling geometry of the model in equation (3.15), for the HMC algorithm (`stan`, left) and Metropolis/Gibbs sampler (`jags`, right), respectively.

The first thing to notice is the complexity of the joint posterior geometry, with a steep funnel shape as the values of v gets smaller and smaller. This makes sense, as the exponential of larger negative values, force the standard deviation of θ to be narrower and narrower around its mean (zero).

The second thing to notice is that HMC (left panel) still manages to successfully explore the steep parameter space of v and θ. This can be observed through the blue points that are scattered in most of the geometry, albeit some parts are less visited (area where $\theta > 0$). This does not happen under the Gibbs sampler. In fact, `jags` did not managed to escape the narrow funnel shape, no matter the number of iterations used in the adaptation, burn-in and sampling procedures[5] (panel B). Moreover, as shown in figure A.1 (appendix), the

[5] `stan` used 3 chains with $1,000$ iterations for adaptation, and $1,000$ iterations for sampling. `jags`

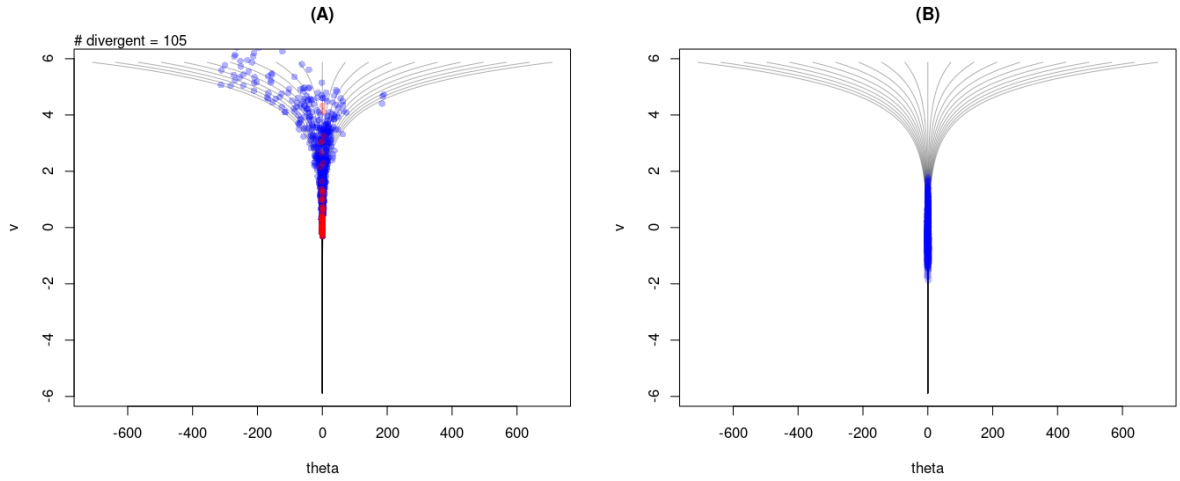

Figure 3.3: Posterior sampling geometry. Centered Parametrization. (A) HMC exploration, blue points are accepted samples, red points denote divergent transitions. (B) Metropolis/Gibbs exploration, blue points are samples.

traceplots indicate the chains are "healthy", ignoring the clear lack of exploration of the joint distribution.

Finally, although HMC did not fully explore the joint distribution, at least the method let us know which parts of it did not manage to visit appropriately (see red points indicating divergent transitions). The last is important, because HMC gives us the opportunity to identify where can we improve our parametrization, something that is not available on Metropolis or Gibbs.

3.4.2 So, how can we solve this?

There are three proposed solutions for the previously explained issue: (i) adapt the HMC warm-up, (ii) use regularizing priors, and (iii) change the posterior sampling geometry. This section will explain briefly the first, outline the geometrical perspective of the second (as its results are well studied); and advocate for the benefits of the third, in the context of IRT models.

Adapt warm-up

As explained in previous sections, the HMC requires a warm-up phase to adapt the parameters' sampling. In it, two meta-parameters are "tuned-in" (`leapfrogs` and `step size`); while a rejection criterion (`adapt_delta`) remains constant. As we recall, HMC follows a rejection procedure that inform us when the method was not able to maintain the "energy" of the system, producing a divergent transition; and in the core of this procedure is the rejection criterion.

used 3 chains with 5,000 iterations for adaptation, 5,000 iterations for burn-in, and 5,000 iterations for sampling.

As it is pointed out by McElreath [45], divergent transitions do not damage directly our posterior distribution approximation, but they do hurt indirectly, as the region where divergent transitions occur is hard to properly explore.

Therefore, to reduce the impact of the divergent transitions, the user can increase the criterion's threshold above the default values (`adapt_delta`= 0.95), resulting in slower but more confident exploration of the posterior distribution.

Finally, in regards to this solution, two important point should be kept in mind. First, the solution is method specific, as HMC is the only algorithm that uses this parameter. And second, the solution forces the method to make a slower posterior investigation, therefore requiring the user to set longer chains to achieve ergodicity.

Regularizing priors

As previously mentioned, several authors have shown the benefits of using weakly regularizing priors in diverse types of models [45, 21, 31]. However, few of them have described how the use of regularizing priors work from a geometrical perspective. Consider the use of a regularizing prior on equation (3.15):

$$\begin{aligned} v &\sim N(0, 1) \\ \theta &\sim N(0, \exp(v)) \end{aligned} \tag{3.16}$$

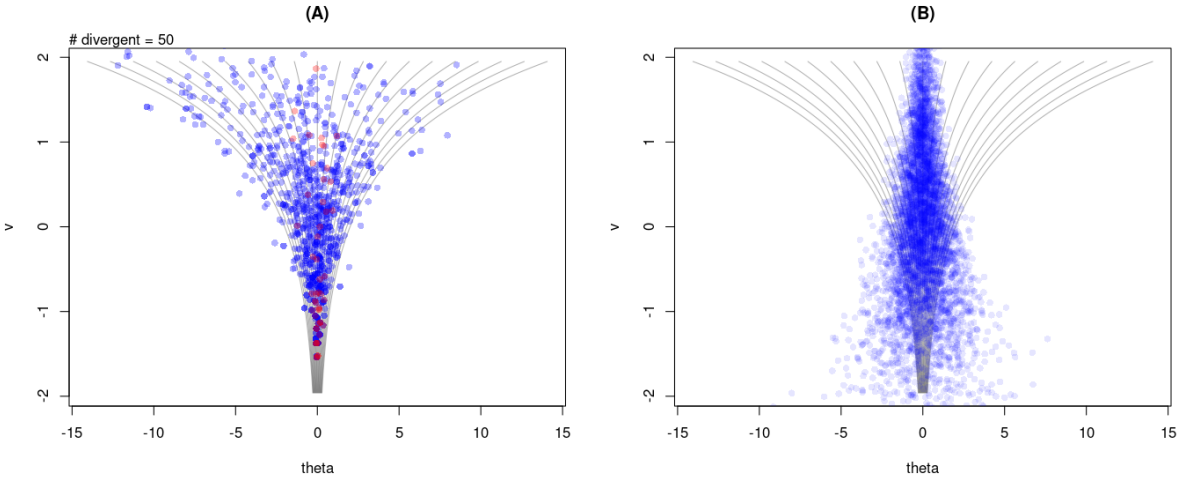

Figure 3.4: Posterior sampling geometry. Centered Parametrization with mildly informative priors. (A) HMC exploration, blue points are accepted samples, red points denote divergent transitions. (B) Metropolis/Gibbs exploration, blue points are samples.

Figure 3.4 shows the joint posterior sampling geometry of equation (3.16), for the HMC algorithm (`stan`, left) and Metropolis/Gibbs sampler (`jags`, right), respectively.

The first thing to notice from the figure is that the geometry available for exploration seems more broad. However, what is actually happening is that by adding information about v, we are "zooming" into the posterior geometry observed in figure 3.3, according to the range of the new priors. This further benefits the exploration of the posterior, as

extreme ranges don't have to be visited from the start, e.g. the narrow funnel in the range $[-6, -2]$ of v from figure 3.3.

The second thing one can notice is that HMC does not "loose time" exploring sections of the posterior distribution that are not needed. Surprisingly in contrast, the Gibbs sampler explore non-useful sections and leaves the useful ones unexplored. Again, this is true no matter the number of iterations used in the adaptation, burn-in and sampling procedures[6]. Moreover, the chains in figure A.2 (appendix) seem to be ergodic, ignoring the clear lack of exploration of the joint distribution, as in the previous example.

Third, we see that using regularizing priors reduced the number of `stan`'s divergent transitions from 105 (in the previous example) to 50. Although we still observe them in the steepest part of the geometry. Such information is not provided by `jags`.

Finally, we notice a mild improvement on the trace, tranks and ACF plots in figure 3.5, compared to figure 3.2 in the previous example.

Non-centered parametrization

It would seem that, increasing the regularizing power of the priors is the answer to get rid of the sampling issues raised by a complex posterior sampling geometry. However, there is a limit on the information a prior can contain, without imposing strong assumptions about the parameters (see section 3.3.5 for an example). Furthermore, sometimes researchers want the information on the data dominates the posterior parameter space. In those cases, imposing even mildly informative priors is a solution that is out of the picture.

In this context, Betancourt and Girolami [6] indicated that prior information can be included in the model, not only through the prior distributions, but also by encoding it in the model itself, taking advantage of the hierarchical structure explicitly, i.e. change the posterior sampling geometries.

Following Papaspiliopoulos et al. [51, 52], and Betancourt and Girolami [6], under the Bayesian framework, a change in geometry consist on a model re-parameterization that seeks to remove the dependence of parameters on other sampled parameters, therefore favoring the performance of the MCMC chains. This is what the literature has dubbed as the non-centered parametrization (NCP).

In light of previous example, changing the posterior sampling geometry means to modify equation (3.15) in the following way:

$$\begin{aligned} v &\sim N(0, 3) \\ z &\sim N(0, 1) \\ \theta &= \exp(v)\, z \end{aligned} \tag{3.17}$$

notice v has the original assumed variability, but now θ is defined in a way, that is no longer sample dependent on v, i.e. we no longer sample θ directly but z, and transform it back.

The motivation for this re-parametrization have seeds in the calculation of the standard score (z-sore). Notice that if $\theta \sim N(\mu_\theta, \sigma_\theta)$ where $\sigma_\theta = \exp(v)$, then $(\theta - \mu_\theta)/\sigma_\theta = z$ and $z \sim N(0, 1)$. Using this process in reverse, we notice $\theta = \mu_\theta + \sigma_\theta z = \exp(v)\ z$ when $\mu_\theta = 0$, then $\theta \sim N(0, \exp(v))$. As the previous, there are transformation that can be

[6] the author used the same settings as in the previous example.

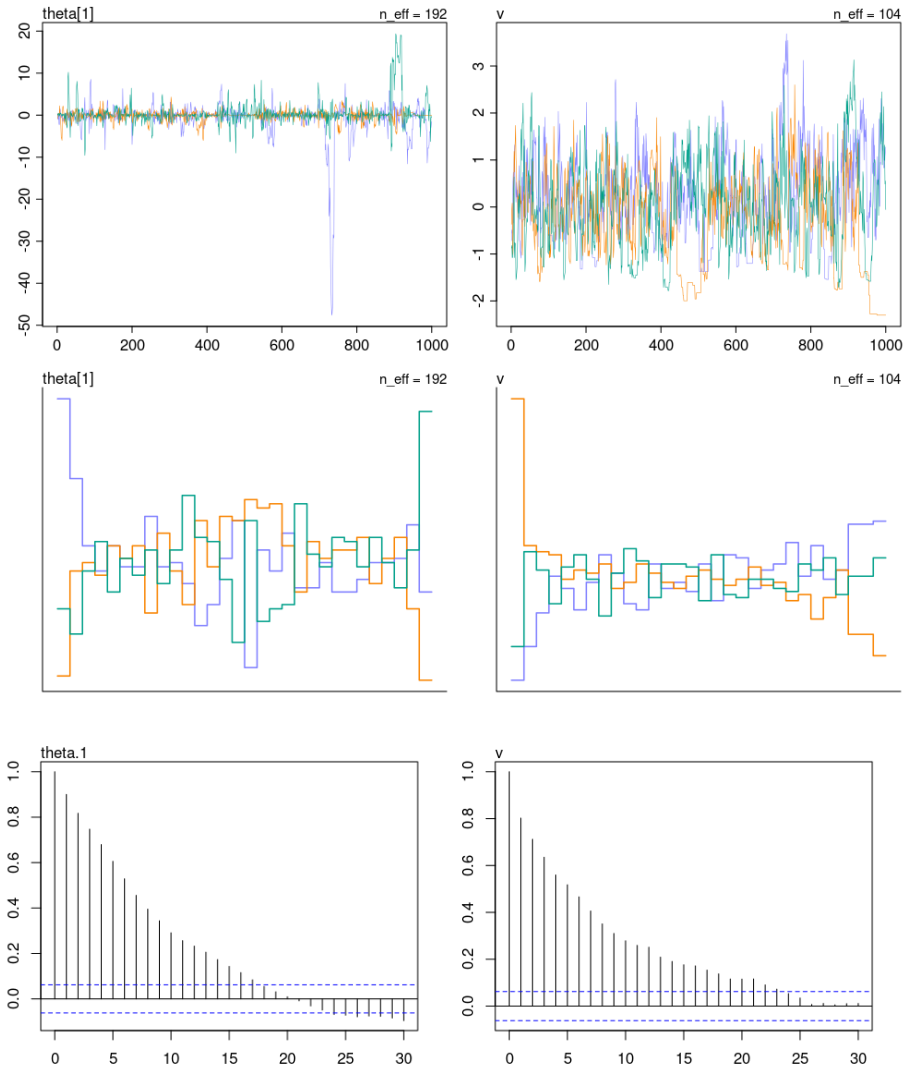

Figure 3.5: The Devil's funnel. Centered Parametrization with mildly informative priors. (Top) Trace plots for the iterations on three chains. (Middle) Trank plots for the same data. (Bottom) Auto-correlation Functions (ACF) plot for the iterations.

done for other distributions, and they can even be extended to the multivariate normal case, through the use of the Cholesky decomposition [45].

Figure 3.6 show the posterior sampling geometry for the HMC and Gibbs sampler, under the re-parametrization. Notice both algorithms manage to explore more successfully the joint distribution, although the HMC does it a bit better on the extremes of v. Moreover, the HMC shows no divergent transitions, meaning the posterior is "visited" without issues. Evidence of the latter can be seen on figure 3.7, where the chains show clear signs of stationarity, convergence and good mixing.

Finally, Papaspiliopoulos et al. [52] indicated that the success of the NCP strategy is largely dependent on the specifics of the model and data, i.e. neither CP or NCP is more "optimal" than the other. The author pointed out that the natural CP showed better performance when conditional conjugacy was present, i.e. when the posterior distribution belonged to the same parametric family as the likelihood or prior distribution. On the

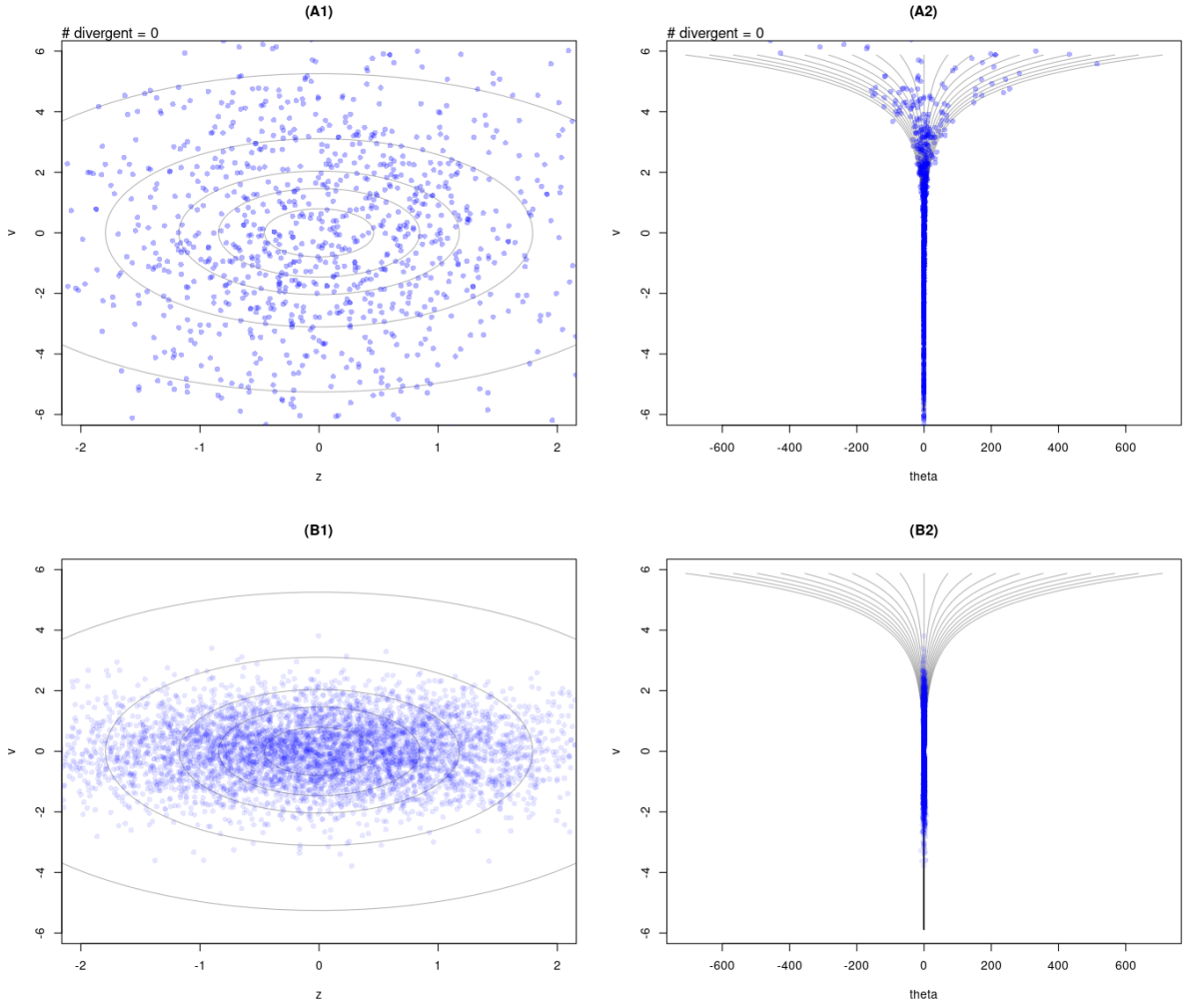

Figure 3.6: Posterior sampling geometry. Non-Centered Parametrization. (A1, B1) geometry defined by the independent parameters v and z. (A2, B2) Original sampling geometry defined by v and θ. (A1, A2) HMC algorithm (`stan`). (B1, B2) Metropolis/Gibbs sampler (`jags`). Blue points are accepted samples. Red points denote divergent transitions.

other hand, the NCP worked as its complement, excelling when the previous requirement was not present. Therefore, due to this complementary behavior, it makes sense that recent advancements on the topic are focused on producing sample mechanisms located in a continuous between CP and NCP, as greater performance improvements can be obtained by leveraging on both, e.g. Interleaved HMC (iHMC) or Variational Inference (VI) [18, 19, 51, 52, 25].

Chapter 4 and 5 will show the application of NCP in the context of an IRT model.

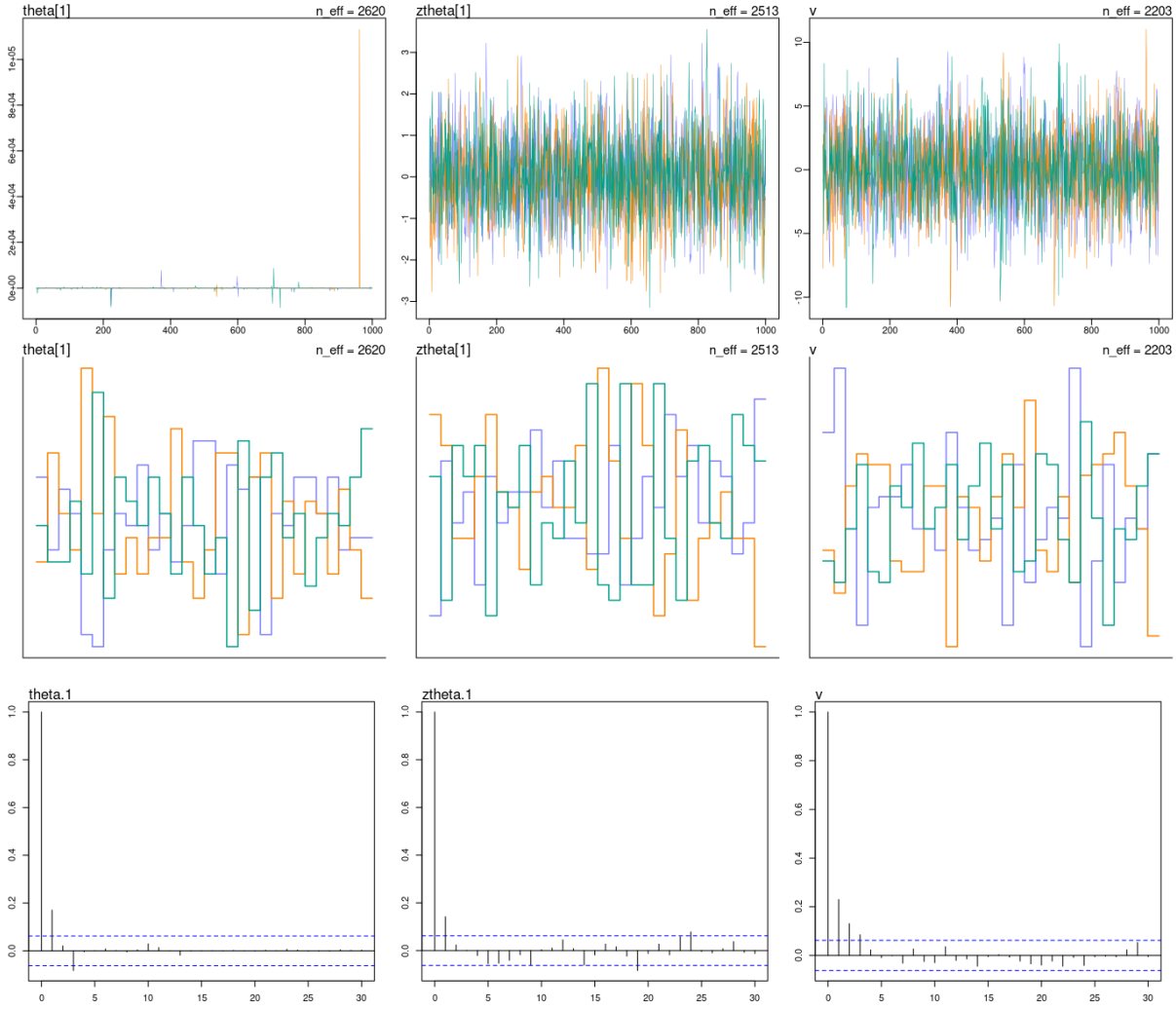

Figure 3.7: The Devil's funnel. Non-centered Parametrization. (Top) Trace plots for the iterations on three chains. (Middle) Trank plots for the same data. (Bottom) Autocorrelation Functions (ACF) plots for the iterations.

Chapter 4

Simulation Study

4.1 Objectives

The simulation study was conducted to assess three attributes of the bayesian implementation of the GLLAMM for dichotomous outcomes:

1. **Performance.** The study assessed the performance of the MCMC chains in terms of achieving ergodicity, under the centered (CP) and non-centered parametrization (NCP), respectively.

2. **Recovery capacity.** The study evaluated the capacity to recover the parameters of interest, e.g. structural regression parameters, latent variables and loadings. However, it centered its focus on the recovery of the structural regression parameters, as they are highly relevant for making appropriate inferences at the individual level.

3. **Retrodictive accuracy.** The study appraised the capacity of the implementation to retrodict the data of interest, according to a set of aggregating dimensions.

4.2 Conditions

In order to investigate the previous three objectives, a full factorial design would need to consider a rather large number of experimental conditions (48). The number of conditions would result from considering the following factor levels: (i) 2 levels related to the value of the rejection criteria for the HMC method, i.e. a default `adapt_delta`= 0.95 or `adapt_delta`= 0.99, (ii) 2 levels related to the use/no use of regularizing priors, (iii) 2 levels for the parametrization of the model, CP and NCP, (iv) 3 simulated sample sizes of interest, and (v) 2 levels related to the models of interest.

However, because bayesian practitioners usually use the centered parametrization in conjunction with other solutions to reach acceptable levels of performance (and the practice is fairly extended in the literature), we believe a large set of the aforementioned conditions can be trimmed out in favor of a more realistic comparison. Therefore, we decided to use a fractional factorial design with $3 \times 2 \times 2 = 12$ experimental conditions, where only the sample sizes, parametrization, and models of interest were manipulated. On the other hand, we decided to maintain the remaining two experimental factors on levels close to realistic implementations, that is, we used weakly regularizing priors with

`adapt_delta`= 0.99, where the former decision was substantively justified in sections 3.3.5 and 3.4.2.

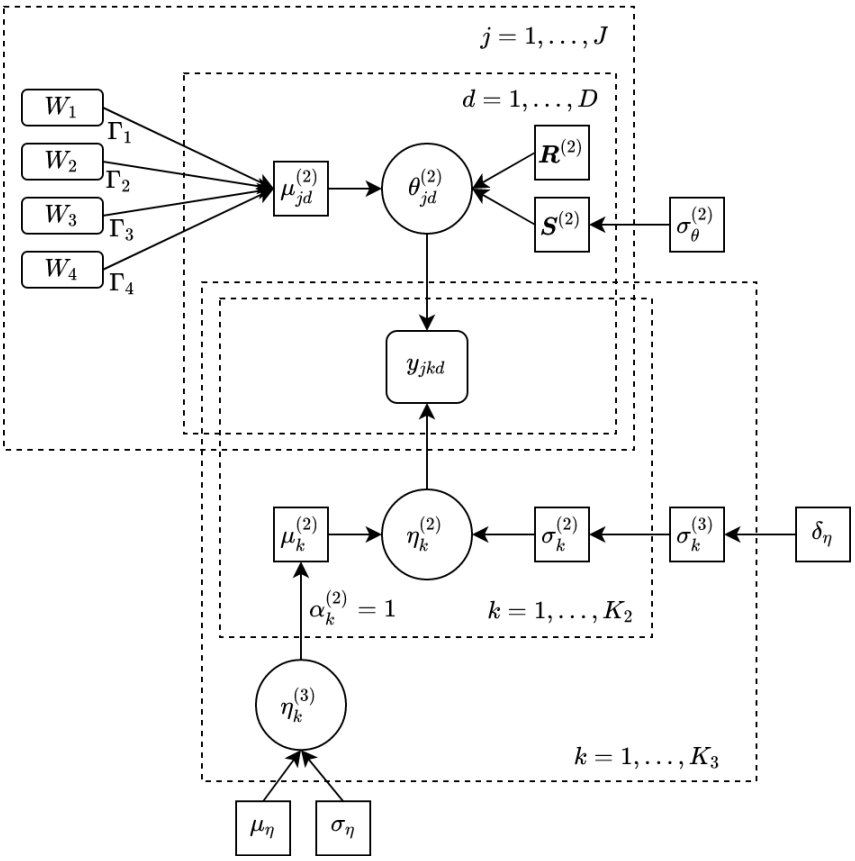

Figure 4.1: Directed Acyclig Graph (DAG). First-order latent variable model (FOLV). Circles represent latent variables. Squares represent parameters or parameters for priors. Large squares represent nesting in specific units.

Therefore, first, the author selected three different samples sizes to generate the data under analysis: 500, 250, and 100. The literature on IRT models present several implementations with samples sizes above 250, however, few present samples lower than that. The author decided to use a sample size of a 100 to fill in this gap. Moreover, the decision was also supported by the notion that the change of the posterior sampling geometries could benefit the performance and recovery capacity of the implementation, under this setting.

Second, as expected, the author used two parametrization of the models: CP and NCP. To the author's knowledge, the IRT literature has not evaluated the change of posterior sampling geometries, as an alternative to improve the performance of the bayesian implementation of said models. The study is set to fill in part of this gap.

Third, the author evaluated the performance, recovery capacity and retrodictive accuracy of a first- and second-order latent variable models; FOLV and SOLV, respectively.

Consequently, ten (10) data sets were generated for each study condition, following the algorithm in section 4.3. Each data set resembled responses to 25 binary scored items, conforming to the SOLV model defined in figure 4.2. The model was motivated by the hypothesized structure of the reading comprehension sub-test, from the Peruvian public

teaching career national assessment (see chapter 5). Finally, the latent structure, structural regression parameters, and loadings remained unchanged throughout the simulation replicas, to reduce experimental error [34].

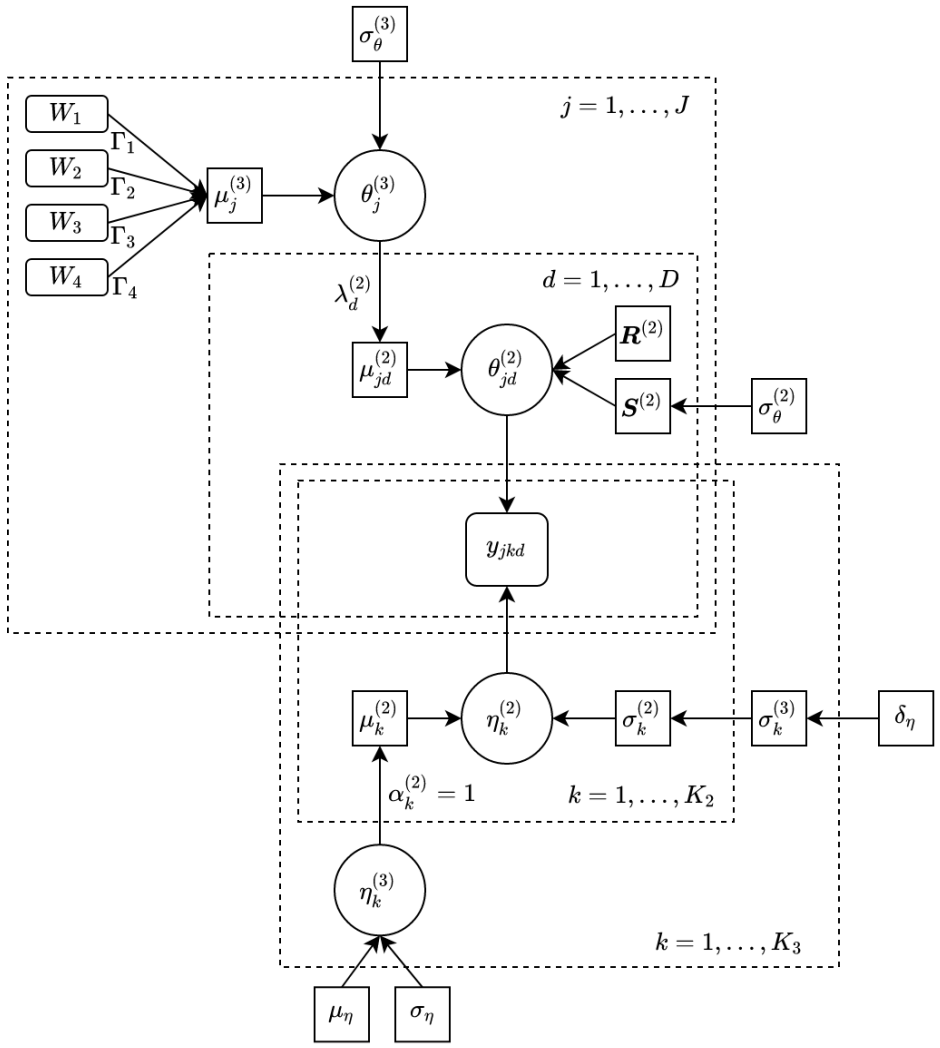

Figure 4.2: Directed Acyclig Graph (DAG). Second-order latent variable model (SOLV). Circles represent latent variables. Squares represent parameters or parameters for priors. Large squares represent nesting in specific units.

4.3 Algorithm

Each data replication was simulated following a six-step procedure. First, the author randomly simulated a collection of pseudo-covariates $\mathbf{W}_\theta = [W_1, W_2, W_3, W_4]$, motivated by a similar information set present in the reading comprehension sub-test. The generated covariates were: (i) a binary "gender" variable (W_1), describing males and females, (ii) an integer "age" variable (W_2) with range $[30, 65]$, the latter corresponding to the peruvian age of retirement from the public teacher career, (iii) a three-level categorical "education" variable (W_3), indicating the type of education the individual received: institute only, university only, or both; and finally, (iv) a four-level categorical "experience" variable

(W_4), denoting the individual's years of work experience, where the higher the category, the higher the years of experience. Associated with these, the author defined their structural regression parameters $\mathbf{\Gamma}_\theta = [\Gamma_0, \Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4]$ where: (i) $\Gamma_0 = 0$, indicating the absence of an intercept; (ii) $\Gamma_1 = [\gamma_m, \gamma_f] = [0, 0.5]$, for males and females, respectively; (iii) $\Gamma_2 = -0.02$, indicating the individuals loose ability with age, in a linear manner; (iv) $\Gamma_3 = [\gamma_{io}, \gamma_{uo}, \gamma_b] = [-0.5, 0.5, 0]$, assuming individuals with university degree have better ability levels, followed by individuals with both educations, and individuals with institute degrees; and lastly (v) $\Gamma_4 = [\gamma_{0y}, \gamma_{5y}, \gamma_{10y}, \gamma_{11+y}] = [-0.5, 0, 0.35, 0.5]$, implying experience has decreasing returns on abilities.

Second, the study simulated the second- and first-order latent variables, corresponding to the reading comprehension ability and its three sub-dimensions: literal, inferential and reflective. Reading comprehension ($\theta_j^{(3)}$) was generated from a normal distribution $N(\mu_j^{(3)}, \sigma_\theta^{(3)})$, with $\mu_j^{(3)} = \mathbf{\Gamma}_\theta \, \mathbf{W}_\theta$, that is, the linear combination of the simulated covariates and its corresponding structural regression parameters, and $\sigma_\theta^{(3)} = 0.5$. On the other hand, the three sub-dimensions were generated from a multivariate normal distribution $MVN(\boldsymbol{\mu}_j^{(2)}, \mathbf{\Sigma}^{(2)})$, with a mean vector $\boldsymbol{\mu}_j^{(2)} = [\mu_{j1}^{(2)}, \mu_{j2}^{(2)}, \mu_{j3}^{(2)}] = [\lambda_1^{(2)}\theta_j^{(3)}, \ \lambda_2^{(2)}\theta_j^{(3)}, \ \lambda_3^{(2)}\theta_j^{(3)}]$, loadings $\boldsymbol{\lambda}^{(2)} = [\lambda_1^{(2)}, \lambda_2^{(2)}, \lambda_3^{(2)}] = [0.95, 0.95, 0.95]$, and a factored covariance matrix $\mathbf{\Sigma}^{(2)} = \mathbf{S}^{(2)} \cdot \mathbf{R}^{(2)} \cdot \mathbf{S}^{(2)}$; where $\mathbf{S}^{(2)} = \boldsymbol{\sigma}_\theta^{(2)}\mathbf{I}$ is a diagonal standard deviation matrix with $\boldsymbol{\sigma}_\theta^{(2)} = [0.5, 0.5, 0.5]$, whereas $\mathbf{R}^{(2)} = \mathbf{I}$ is an identity correlation matrix, implying the simulated sub-dimensions are independent, after accounting for the reading comprehension.

Third, the author defined five (5) common stimulus or texts for the items, where the mean difficulty for the texts $\boldsymbol{\eta}^{(3)} = [\eta_1^{(3)}, \eta_2^{(3)}, \eta_3^{(3)}, \eta_4^{(3)}, \eta_5^{(3)}] = [-1.50, -0.75, 0, 0.75, 1.50]$; whereas the deviation from said mean difficulties were $\sigma_k^{(3)} = 0.5$ for all texts.

Fourth, 25 items were randomly generated from independent normal distributions $N(\mu_k^{(2)}, \sigma_k^{(2)})$, with $\mu_k^{(2)} = \boldsymbol{\eta}^{(3)}\boldsymbol{\alpha}^{(2)}\mathbf{A}$ and $\sigma_k^{(2)} = \sigma_k^{(3)}$; where $\boldsymbol{\alpha}^{(2)} = \mathbf{1}$, indicating the difficulty of the common stimulus directly explained the difficulty of the items, and $\mathbf{A}$ was a block design matrix that maps the items to its corresponding passage. Lastly, the items' measured dimensions were also generated at random, for each replica.

Fifth, the author calculated the linear predictor v_{jkd} and probability of endorsing an item π_{jkd} according to equations (2.2), (2.3), and (2.5), respectively. The probabilities were calculated using the logistic inverse-link function.

Sixth and last, the outcome y_{jkd} was simulated from a Bernoulli distribution as in equation (2.1), with a probability of success calculated as in the previous step.

The code associated with the full simulation process can be found in Appendix B.2.1.

4.4 Evaluation criteria

As stated in the objectives, the study was set to evaluate the performance, recovery capacity, and retrodictive accuracy of the bayesian implementation.

First, to assess the performance of the MCMC chains, in terms of achieving stationarity, convergence and good mixing, the author followed the usual bayesian approach. The approach involved the visual evaluation of trace plots, for stationarity and convergence (more useful to establish lack of thereof); and trank and autocorrelation plots (ACF), for good mixing. Moreover, the assessment of convergence and mixing were supported by the *potential scale reduction factor* (`Rhat`) and *effective sample size* (`n_eff`) statistics

developed by Gelman et al. [22] (pp. $284 - 287$).

Second, to evaluate the recovery capacity for all the parameters $\boldsymbol{\Omega} = \{\boldsymbol{\beta}, \boldsymbol{\Lambda}, \boldsymbol{\Theta}, \boldsymbol{\Psi}, \boldsymbol{\Gamma}\}$, we used the between replica root mean squared error (RMSE_B), i.e. the extent of the deviation the posterior means exhibited from the true generating values, in all replicate simulations. The RMSE_B for each parameter of interest was defined as follows:

$$\text{RMSE}_B\left(\eta_k^{(m)}\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{\eta}_{krs}^{(m)} - \eta_k^{(m)}\right)^2} \tag{4.1}$$

$$\text{RMSE}_B\left(\theta_{jd}^{(l)}\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{\theta}_{jdrs}^{(l)} - \theta_{jd}^{(l)}\right)^2} \tag{4.2}$$

$$\text{RMSE}_B\left(\Gamma_w\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{\Gamma}_{wrs} - \Gamma_w\right)^2} \tag{4.3}$$

$$\text{RMSE}_B\left(\lambda_d^{(l)}\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{\lambda}_{drs}^{(l)} - \lambda_d^{(l)}\right)^2} \tag{4.4}$$

$$\text{RMSE}_B\left(\rho_{dq}^{(2)}\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{\rho}_{dqrs}^{(2)} - \rho_{dq}^{(2)}\right)^2} \tag{4.5}$$

where the "hat" parameters with index rs described the parameter's posterior sample $s = 1, \ldots, S$, with $S = 3,000$; and replica $r = 1, \ldots, R$, with $R = 10$. Additionally, $\rho_{dq}^{(2)}$ denoted the dimensions' correlation parameters in matrix $\boldsymbol{R}^{(2)}$, with $q > d$, where $d = 1, \ldots, 3$ and $q = 1, \ldots, 3$. Notice in the case of the FOLV model, in order to asses the recovery capacity of the correlations, we were forced to calculate an approximate implied correlation structure between sub-dimensions, resulting from having a miss-specified model. The calculation was perform through the use of Wright's tracing rules [5].

Finally, since IRT models are known to be invariant to the shift of the linear predictor [4, 7], i.e. the addition/subtraction of a constant to the abilities/difficulties results in the same probability value, it could happen that we observe substantial differences in the recovery of the parameters, that not necessarily implies a classification error [72]. Therefore, to avoid a mistake in the model's assessment of fit, the current research will use posterior predictive checks, i.e. we use the parameters' posterior distribution to asses how good is the retrodictive accuracy of the implementation.

The retrodictive accuracy was measured by the root mean squared error of the responses' predictive proportion $\hat{p}$, versus the observed proportion p, according to a set of aggregating dimensions. However, since there is prediction uncertainty within and

between replicas, we defined two measures:

$$\overline{\text{RMSE}}_W\left(p_j\right) = \frac{1}{R}\sum_{r=1}^{R}\sqrt{\frac{1}{S}\sum_{s=1}^{S}\left(\hat{p}_{jrs} - p_j\right)^2} \tag{4.6}$$

$$\text{RMSE}_B\left(p_j\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{p}_{jrs} - p_j\right)^2} \tag{4.7}$$

where $\overline{\text{RMSE}}_W$ and RMSE_B denotes the average within and between prediction root mean squared error, respectively; while $j = 1, \ldots, J$ defined the individual's index, $S = 3,000$ and $R = 10$. Notice from the previous equations that we obtain two measures of deviations from the true proportion, per individual.

In a similar manner, we calculated the statistics for each item, text, dimension, and even per each individual and simulated covariate combination. Such statistics were defined as follows:

$$\overline{\text{RMSE}}_W\left(p_k\right) = \frac{1}{R}\sum_{r=1}^{R}\sqrt{\frac{1}{S}\sum_{s=1}^{S}\left(\hat{p}_{krs} - p_k\right)^2} \tag{4.8}$$

$$\text{RMSE}_B\left(p_k\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{p}_{krs} - p_k\right)^2} \tag{4.9}$$

$$\overline{\text{RMSE}}_W\left(p_l\right) = \frac{1}{R}\sum_{r=1}^{R}\sqrt{\frac{1}{S}\sum_{s=1}^{S}\left(\hat{p}_{lrs} - p_l\right)^2} \tag{4.10}$$

$$\text{RMSE}_B\left(p_l\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{p}_{lrs} - p_l\right)^2} \tag{4.11}$$

$$\overline{\text{RMSE}}_W\left(p_d\right) = \frac{1}{R}\sum_{r=1}^{R}\sqrt{\frac{1}{S}\sum_{s=1}^{S}\left(\hat{p}_{drs} - p_d\right)^2} \tag{4.12}$$

$$\text{RMSE}_B\left(p_d\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{p}_{drs} - p_d\right)^2} \tag{4.13}$$

$$\overline{\text{RMSE}}_W\left(p_{jw}\right) = \frac{1}{R}\sum_{r=1}^{R}\sqrt{\frac{1}{S}\sum_{s=1}^{S}\left(\hat{p}_{jwrs} - p_{jw}\right)^2} \tag{4.14}$$

$$\text{RMSE}_B\left(p_{jw}\right) = \sqrt{\frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{S}\sum_{s=1}^{S}\hat{p}_{jwrs} - p_{jw}\right)^2} \tag{4.15}$$

where $k = 1, \ldots, 25$ items; $l = 1, \ldots, 5$ texts; $d = 1, \ldots, 3$ dimensions; and $w = 1, \ldots, 4$ simulated covariates.

4.5 Parameter estimation

4.5.1 Likelihood, priors and hyper-priors

As stated in previous sections, we analyzed two models, that is, the FOLV model depicted in figure 4.1, and the SOLV model depicted in figure 4.2. In this section, we proceed to enumerate the likelihood functions, and the full set of priors and *hyper-priors* used, for the centered and non-centered parametrizations, respectively.

First, for the centered parametrization (CP), as stated in equations (3.4), (3.5), and (3.6), the distributional and systematic part of both models was defined as follows:

$$y_{jkd} \sim \text{Bernoulli}(\pi_{jkd}) \tag{4.16}$$

$$\text{logit}(\pi_{jkd}) = v_{jkd} \tag{4.17}$$

$$v_{jkd} = \theta^{(2)}_{jd} - \eta^{(2)}_{k} \tag{4.18}$$

Notice the linear predictor can be considered a multilevel-multidimensional extension of the well known Rasch IRT model [60]. The first-order latent variables were defined as follows:

$$\boldsymbol{\theta}^{(2)}_{j} = \left[\theta^{(2)}_{j1}, \theta^{(2)}_{j2}, \theta^{(2)}_{j3}\right] \tag{4.19}$$

$$\boldsymbol{\theta}^{(2)}_{j} \sim \text{MVNormal}\left(\boldsymbol{\mu}^{(2)}_{j}, \boldsymbol{\Sigma}^{(2)}\right) \tag{4.20}$$

where $\boldsymbol{\Sigma}^{(2)}$ was defined by a factorization of the variances and correlations structures, in the following form:

$$\boldsymbol{\Sigma}^{(2)} = \boldsymbol{S}^{(2)} \cdot \boldsymbol{R}^{(2)} \cdot \boldsymbol{S}^{(2)} \tag{4.21}$$

$$\boldsymbol{S}^{(2)} = \boldsymbol{\sigma}^{(2)}_{\theta}\mathbf{I} = \begin{pmatrix} \sigma^2_1 & 0 & 0 \\ 0 & \sigma^2_2 & 0 \\ 0 & 0 & \sigma^2_3 \end{pmatrix} \tag{4.22}$$

$$\boldsymbol{R}^{(2)} = \begin{pmatrix} 1 & \rho_{12} & \rho_{13} \\ \rho_{21} & 1 & \rho_{23} \\ \rho_{31} & \rho_{32} & 1 \end{pmatrix} \tag{4.23}$$

while $\boldsymbol{\mu}^{(2)}_{j}$ was defined in two ways, depending on the model we were fitting. For the FOLV model, the mean vector was defined as follows:

$$\boldsymbol{\mu}^{(2)}_{j} = \left[\mu^{(2)}_{j1},\ \mu^{(2)}_{j2}, \mu^{(2)}_{j3}\right] \tag{4.24}$$

$$\mu^{(2)}_{jd} = \Gamma_0 + \Gamma_1 W_{1j} + \Gamma_2(W_{2j} - W_{2\text{min}}) + \Gamma_3 W_{3j} + \Gamma_4 W_{4j} \tag{4.25}$$

where Γ_p and W_{jp} are defined as in section 4.3. Notice the structural regression parameters were not dimension specific. On the other hand, for the SOLV model, $\boldsymbol{\mu}^{(2)}_{j}$ was defined by:

$$\boldsymbol{\mu}^{(2)}_{j} = \left[\mu^{(2)}_{j1},\ \mu^{(2)}_{j2}, \mu^{(2)}_{j3}\right] \tag{4.26}$$

$$\boldsymbol{\lambda}^{(2)} = \left[\lambda^{(2)}_{1},\ \lambda^{(2)}_{2}, \lambda^{(2)}_{3}\right] \tag{4.27}$$

$$\mu^{(2)}_{jd} = \lambda^{(2)}_{d}\ \theta^{(3)}_{j} \tag{4.28}$$

where d denoted the index of the individual's dimension. Lastly, because the SOLV contemplates an additional level, corresponding to the reading comprehension ability, an additional set of parameters was defined as follows:

$$\theta_j^{(3)} \sim \text{Normal}\left(\mu_j^{(3)}, \sigma_\theta^{(3)}\right) \tag{4.29}$$

$$\mu_j^{(3)} = \Gamma_0 + \Gamma_1 W_{1j} + \Gamma_2 (W_{2j} - W_{2\text{min}}) + \Gamma_3 W_{3j} + \Gamma_4 W_{4j} \tag{4.30}$$

Turning now our attention to the items' parameters $\eta_k^{(2)}$, both models had:

$$\eta_k^{(2)} \sim \text{Normal}\left(\mu_k^{(2)}, \sigma_k^{(2)}\right) \tag{4.31}$$

$$\mu_k^{(2)} = \boldsymbol{\eta}^{(3)}\mathbf{A} \tag{4.32}$$

$$\sigma_k^{(2)} = \boldsymbol{\sigma}^{(3)}\mathbf{A} \tag{4.33}$$

where $k = 1, \ldots, 25$ items, $\boldsymbol{\eta}^{(3)} = [\eta_1^{(3)}, \eta_2^{(3)}, \eta_3^{(3)}, \eta_4^{(3)}, \eta_5^{(3)}]$ represented the texts difficulties, $\boldsymbol{\sigma}^{(3)} = [\sigma_1^{(3)}, \sigma_2^{(3)}, \sigma_3^{(3)}, \sigma_4^{(3)}, \sigma_5^{(3)}]$ the text deviations from such difficulties; and $\mathbf{A}$ was a block design matrix that mapped the items to its corresponding passage. In addition, $\eta_k^{(3)}$ and $\sigma_k^{(3)}$ were distributed as follows:

$$\eta_k^{(3)} \sim \text{Normal}\left(\mu_\eta, \sigma_\eta\right) \tag{4.34}$$

$$\sigma_k^{(3)} \sim \text{Exponential}\left(\delta_\eta\right) \tag{4.35}$$

Lastly, we declare the remaining prior and *hyper-priors* in the following form::

$$\boldsymbol{R}^{(2)} \sim \text{LkjCorrelation}(2) \tag{4.36}$$

$$\Gamma_{1c} \sim \text{Normal}(0, 0.5) \tag{4.37}$$

$$\Gamma_2 \sim \text{Normal}(0, 0.5) \tag{4.38}$$

$$\Gamma_{3c} \sim \text{Normal}(0, 1) \tag{4.39}$$

$$\Gamma_{4c} \sim \text{Normal}(0, 0.5) \tag{4.40}$$

where c denotes the categories inside the covariate, different for each variable, while the LKJCorrelation$(\cdot)$ function denotes the Lewandowski, Kurowicka, and Joe prior correlation distribution [38]. Notice from equation (4.21), and the aforementioned prior, that the current research departs from the usual setting of prior distributions for covariances matrices. As it is point out by Depaoli [10], the literature on bayesian MSEM/CFA/IRT have favored the use of the Inverse-Wishart distribution, as a prior for covariances matrices, based on its conjugancy properties. However, the assumptions derived from its use, that is, the priors are equally informative for variances and covariances, is sometimes restrictive. Therefore, following McElreath [45], we believe that by factoring the covariance matrix into variances and correlations, allow us gain control on the preliminary assumptions set to those parameters. Moreover, it makes easier to identify when the researcher is using the unit variance identification scheme (UVI, see next section); and allow us to transition easily to the non-centered parametrization (NCP).

Second, as it was outlined in section 3.4.2, to identify parameters that can be transformed into the NCP, we needed to recognize equations where the parameters' sampling

procedure is dependent on other sampled parameters. A careful inspection of the preceding parametrization revealed that equations (4.20), (4.29), and (4.31) fulfilled this characteristic. Therefore, under the NCP, equation (4.31) was re-defined as follows:

$$\eta_k^{(2)} = \mu_k^{(2)} + \sigma_k^{(2)} z_k^{(2)} \tag{4.41}$$

$$z_k^{(2)} \sim \text{Normal}(0, 1) \tag{4.42}$$

where $\mu_k^{(2)}$ and $\sigma_k^{(2)}$ are defined as in equations (4.32) and (4.33), respectively; while $z_k^{(2)}$ was sampled from independent standard normal distributions, one for each item corresponding to a text passage.

Moreover, equation (4.20) was re-defined as follows:

$$\boldsymbol{\theta}_j^{(2)} = \boldsymbol{\mu}_j^{(2)} + \boldsymbol{S}^{(2)} \cdot \boldsymbol{L}_\Sigma^{(2)} \cdot (\mathbf{z}_j \mathbf{I}) \tag{4.43}$$

$$\mathbf{z}_j = [z_{j1}, \ldots, z_{jd}]^T \tag{4.44}$$

where $\boldsymbol{\mu}_j^{(2)}$ is defined as in equation (4.24), $\boldsymbol{S}^{(2)}$ is defined as in equation (4.22), and $\mathbf{I}$ is an identity matrix. Additionally, we had:

$$z_{jd} \sim \text{Normal}(0, 1) \tag{4.45}$$

$$\boldsymbol{L}_\Sigma^{(2)} \sim \text{LKJCorrelationCholesky}(2) \tag{4.46}$$

where z_{jd} was sampled from independent standard normal distributions (one for each dimension and individual); while $\boldsymbol{L}_\Sigma^{(2)}$ denoted the Cholesky factorization of the correlation matrix $\boldsymbol{R}^{(2)}$. A Cholesky factorization is a decomposition of any positive-definite matrix $\mathbf{A}$, into the product of a lower triangular matrix and its conjugate transpose, i.e. $\mathbf{A} = \boldsymbol{L} \cdot \boldsymbol{L}^T$. Notice the prior distribution of the decomposition LKJCorrelationCholesky$(\cdot)$, implied the same assumption as before, that is, $\boldsymbol{R}^{(2)} = \boldsymbol{L}_\Sigma^{(2)} \cdot \boldsymbol{L}_\Sigma^{(2)T} \sim \text{LKJCorrelation}(2)$[1].

Furthermore, as in the previous cases, equation (4.29) also shows sampling dependence. Therefore, the equation was re-defined as follows:

$$\theta_j^{(3)} = \mu_j^{(3)} + \sigma_\theta^{(3)} z_j \tag{4.47}$$

$$z_j \sim \text{Normal}(0, 1) \tag{4.48}$$

where $\mu_j^{(3)}$ is defined as in equation (4.30), and z_j was sampled from independent standard normal distributions, one for each individual. Finally, the remaining equations defined under the CP maintain their description under the NCP.

4.5.2 Identification

In order to fully specify the model and provide a scale for the latent variables, we decided to use the UVI scheme, that is, to set the scale of the higher-order dimension and subdimensions to one, $\sigma_\theta^{(3)} = 1$ and $\mathbf{S}^{(2)} = \boldsymbol{\sigma}_\theta^{(2)} \mathbf{I}$ with $\boldsymbol{\sigma}_\theta^{(2)} = [1, 1, 1]^T$. The previous effectively turned the covariance matrix $\boldsymbol{\Sigma}^{(2)}$ into a correlation, i.e. $\boldsymbol{\Sigma}^{(2)} = \boldsymbol{R}^{(2)}$ (see equation 4.21). Moreover, we identified the texts' and items' by setting $\mu_\eta = 0$, $\sigma_\eta = 1$, $\delta_\eta = 2$.

[1] See Stan Development Team. [67] for a detailed explanation on the implementation.

Notice we needed to set the scales for all the individuals' dimensions equal to one, because otherwise the latent structure would be under-identified. In either model, at the level of the first-order latent variables (level-2), we would have $(3 \times 4)/2 = 6$ pieces of information available, corresponding to the variances and covariances of the latent variables. Then, the information would need to estimate 10 parameters, i.e. 3 loadings, 3 variances corresponding to the first–order latent variable, 1 variance for the second-order latent variable (if present), and 3 correlations among sub-dimensions, making that portion of the model under-identified. Therefore, by restricting the variances to one, we have enough information to estimate other parameters of interest (correlations and loading, if present). Additionally, this is further justified by the fact that latent variables have no inherent unit in which they are measured, consequently, setting their scale is an arbitrary decision [5].

4.5.3 Prior predictive investigation

As it was outlined in section 3.3.5, we will use prior predictive investigation to assess and visualize the consequences of our prior assumptions. The implications will be assessed from two perspectives: the IRT and the outcome space perspective. For the former, we will investigate what the set of priors imply for the item characteristic curves (ICC), and item information function (IIF). For the latter, we will examine how the prior assumption translate onto the outcome space.

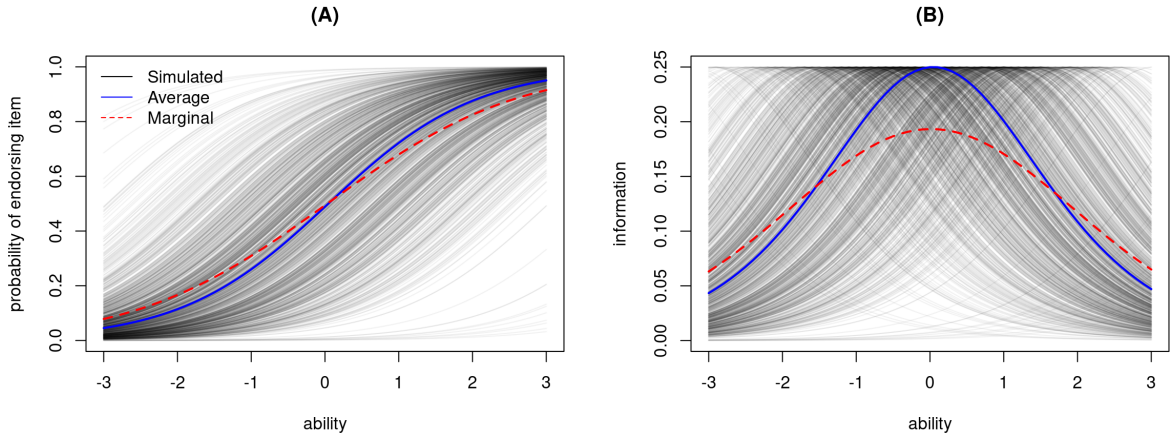

Figure 4.3: First-order latent variable model (FOLV). (A) Item Characteristics Curve, ICC. (B) Item Information Function, IIF.

First, it will useful to define the ICC and IIF functions. Hambleton and Swaminathan [26] defined the ICC as the mathematical function that relates the probability of success on an item, to the ability measured by that same item. In our current implementation, such function would correspond to the systematic part of the GLLAMM, as defined in equations (2.2) and (2.3):

$$\text{ICC} = \pi_{jkd} = \frac{\exp(v_{jkd})}{1 + \exp(v_{jkd})} \tag{4.49}$$

where the linear predictor v_{jkd} contains the items' difficulties $\eta_k^{(2)}$ and the individuals' sub-dimensions/abilities $\theta_{jd}^{(2)}$, as defined in equation (2.4). As previously mentioned, the ICC of our current implementation can be considered as the multilevel-multidimensional extension to the Rasch's ICC.

On the other hand, the same authors defined the IIF as a function that measured the amount of information provided by an item. In this setting, much similar to the field of information theory, "information" is understood as the reduction of uncertainty resulting from using an item to measure the ability of an individual. Given the above, the IIF for the GLLAMM was defined in a similar manner as the Rasch counterpart, in the following form:

$$\text{IIF} = \pi_{jkd}\left(1 - \pi_{jkd}\right) \tag{4.50}$$

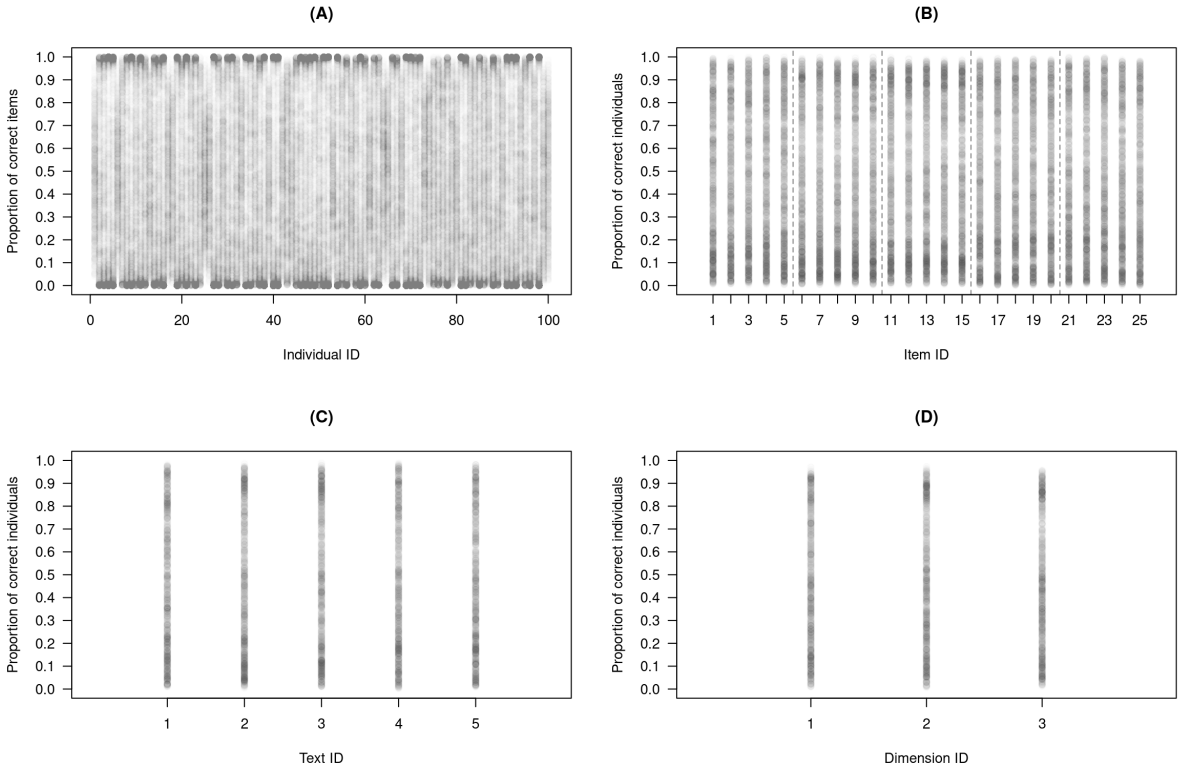

Figure 4.4: First-order latent variable model (FOLV). Aggregated endorsement rate per: (A) individuals, (B) items, (C) text or passage, and (D) measured dimension.

Figure 4.3 shows the ICC and IIF for the FOLV model, resulting from the assumptions integrated by the priors detailed in the previous section. From panel (A), we can notice the difficulty of the items match a wide range of the individuals' abilities (see multiple black solid lines). This means the model expect items' difficulties to be in any part of the continuous range of abilities, with no restriction. Furthermore, the priors allow the model to obtain information about individuals located in the full significant range of the abilities, as seen in panel (B) of the same figure. All of this just means that no unintended

assumption has "creep" into the model, at least from the IRT perspective. A similar result can be seen for the SOLV model (figure A.4, appendix).

Finally, from figure 4.4 and 4.5, we notice that no unintended assumption has been translated to the outcome space of the FOLV model, either. Panel (A) from figure 4.4 shows the individuals' proportion, of correctly answered items, can be present in the full range of possibilities. Furthermore, panel (B) of the same figure shows the proportion of individuals endorsing an item, for any item, can also be present in the full range of possibilities. Lastly, panels (C) and (D) convey similar information aggregated by texts and dimensions. On the other hand, all panels in figure 4.5 revealed the prior distributions did not force any apriori tendencies in the covariates. A similar result can be seen for the outcome space of the SOLV model (figures A.5 and A.6, appendix).

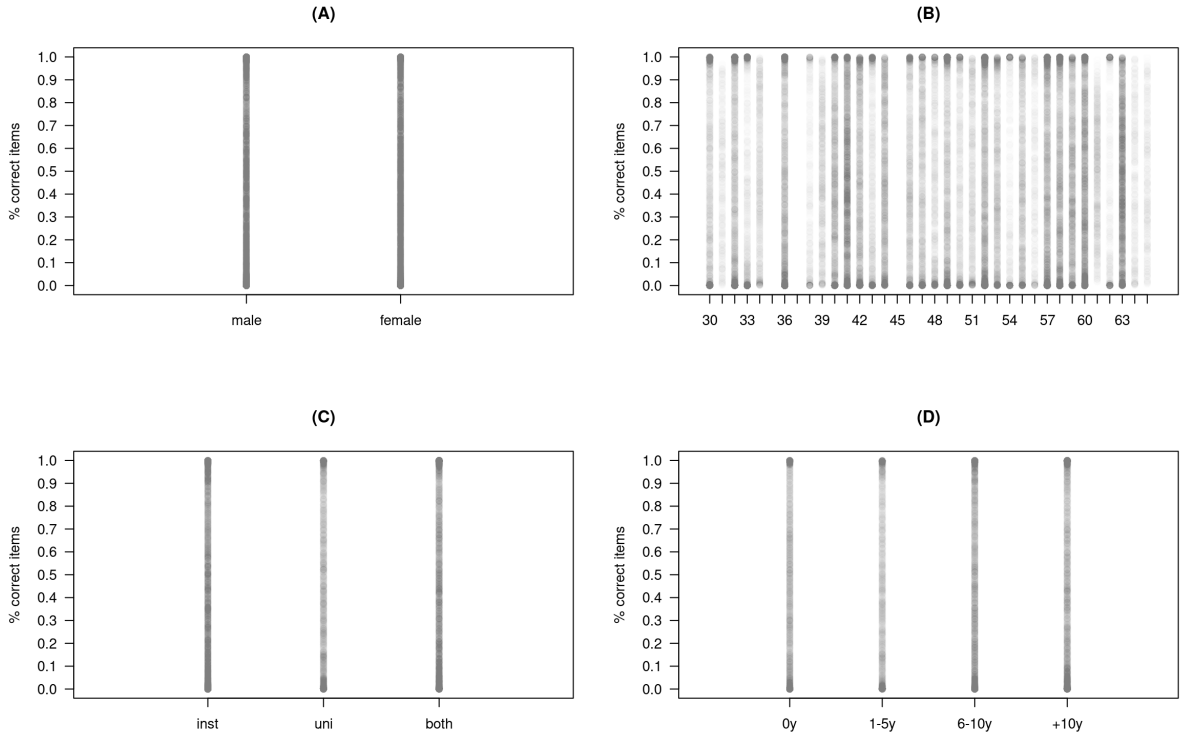

Figure 4.5: First-order latent variable model (FOLV). Aggregated endorsement rate per simulated covariate: (A) gender, (B) age, (C) education, and (D) experience.

4.6 Results

4.6.1 Chain performance

First, the CP and NCP chain performance for the texts' mean difficulties is reported in figures 5.3 and 5.4, respectively. The figures correspond to replica number two of the FOLV model, with a simulated sample size of 100.

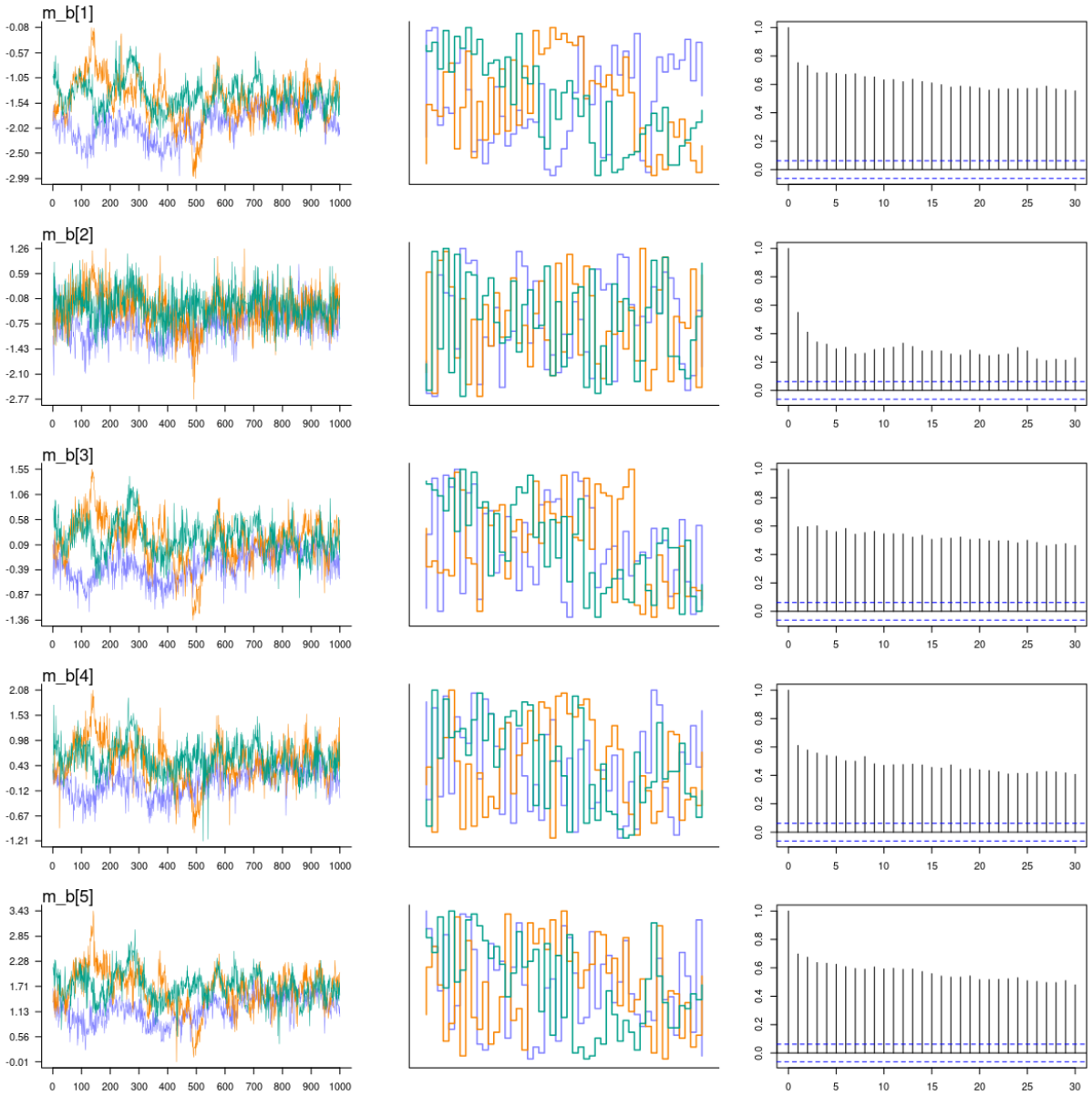

Figure 4.6: First-order latent variable model (FOLV). Sample size 100, replica number 2. Centered parametrization. Mean difficulty per text: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

From the figures, one can easily notice the parameters under the CP did not show signs of achieving ergodicity. The chains showed a clear lack of stationarity and convergence (left panels). Moreover, the iterations did not explore the posterior distribution appropriately, as the several divergent transitions hinted, indicating a lack of good mixing, e.g. the top trace and trank plots show chains that slowly meander through specific areas of the posterior distribution. In addition, the ACF plots reveal the chain iterations were highly auto-correlated, another sign of bad mixing. All of this is further confirmed by panels (A) and (C) from figure 5.5, where we can see the CP has low effective samples sizes (`n_eff`) and higher than appropriate `Rhat` values, compared to the NCP.

In contrast, the parameters from the NCP are closer to achieve ergodicity. The trace

plots show chains that seem to be stationary and convergent. The trank plots show a rapid exploration of the full range of posterior distribution, with no divergent transitions on sight. And finally, the ACF plots showed the iterations were less auto-correlated. Again, this is further confirmed by figure 5.5, where the NCP show greater `n_eff` and appropriate `Rhat` values, versus the CP.

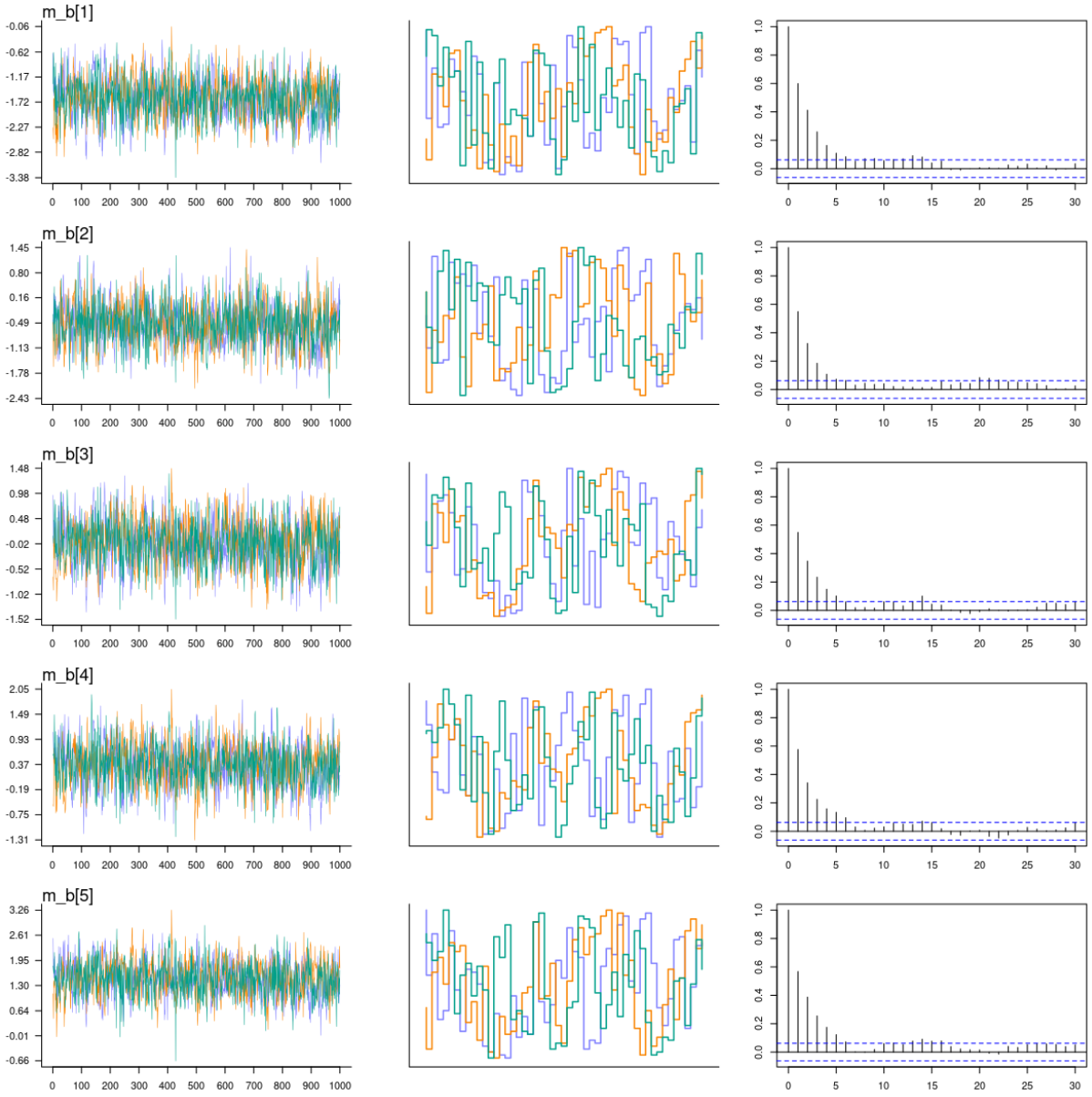

Figure 4.7: First-Order latent variable model (FOLV). Sample size 100, replica number 2. Non-centered parametrization. Mean difficulty per text: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

On the other hand, figures A.7 and A.8 (appendix) show the the CP and NCP trace, trank, and ACF plots for the the difficulties' deviations of the texts. At a simple glance, the plots seem to indicate that both parametrizations achieve a similar level of ergodicity. However, from panels (A) and (C) in figure 5.5, we can notice that while both

parametrization show chains with `Rhat`< 1.05, for some parameters, the CP has lower effective sample sizes than the NCP counterpart. The latter just indicates the NCP has less auto-correlated chains, and therefore a bit better mixing.

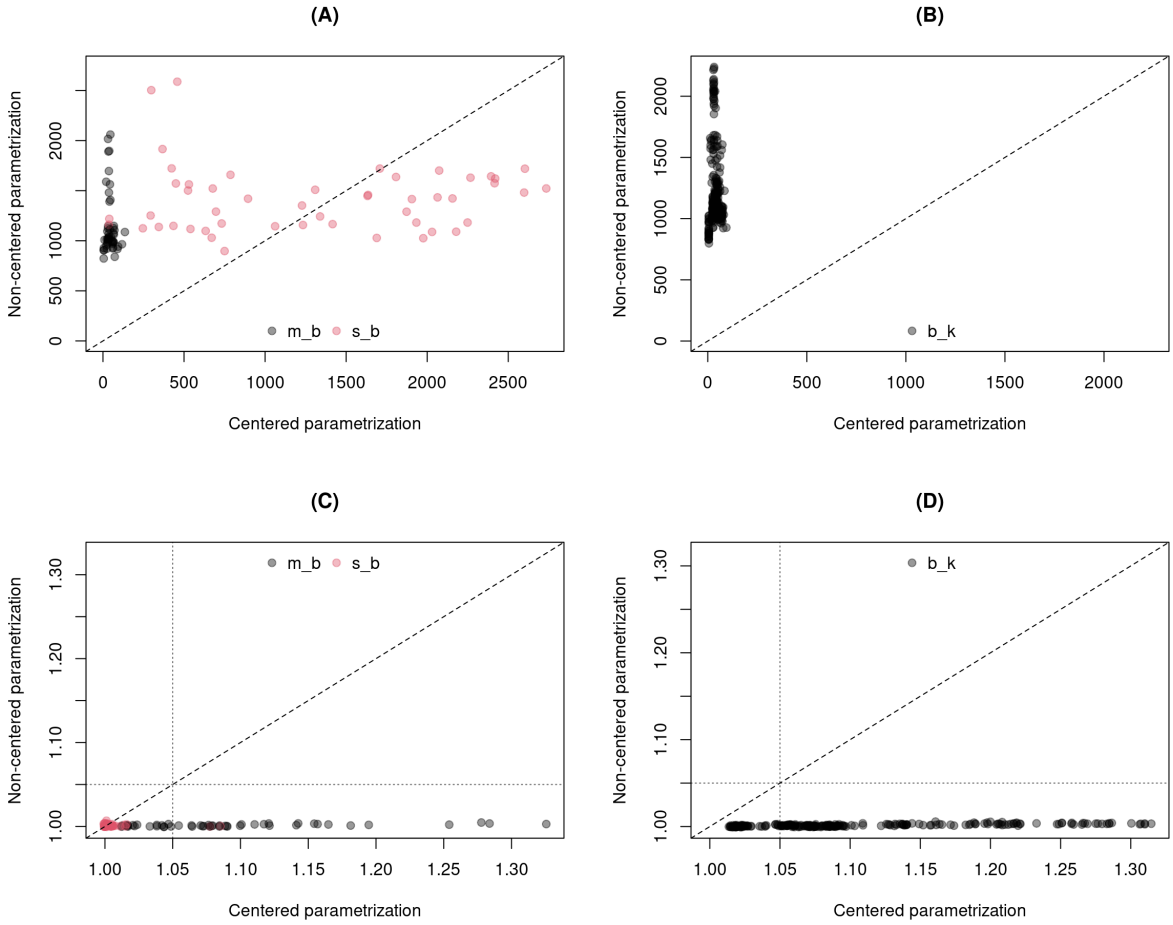

Figure 4.8: First-order latent variable model (FOLV). Sample size 100, all replicas. CP and NCP comparison plot. (A) `n_eff` for texts' mean difficulty and standard deviation of difficulty. (B) `n_eff` for items's difficulties. (C) `Rhat` for texts' mean difficulty and standard deviation of difficulty. (D) `Rhat` for items's difficulties. Diagonal discontinuous line describes equality between CP and NCP. Vertical and horizontal discontinuous lines is set at `Rhat`= 1.05.

Furthermore, figures A.9 and A.10 (appendix) show the chain performance for the CP and NCP item difficulties. The trace, trank and ACF plots show a similar pattern as the one outlined for the difficulty of the texts, albeit more extreme. The CP chains did not achieved stationarity, convergence, nor good mixing; whereas the NCP does. This is further confirmed by panels (B) and (D) of figure 5.5, where one can see observe the NCP has significantly higher effective sample sizes and `Rhat`< 1.05, in contrast to the CP.

Second, the CP chain performance for the reading comprehension sub-dimensions are reported in figures A.11, A.13, A.15 (appendix). The figures correspond to several replicas of the FOLV model, with a simulated sample size of 100. Similar to the preceding texts'

difficulties, the chains also showed a lack of ergodicity, e.g. the left panels of figure A.15 show chains that did not explore the posterior distribution appropriately (see straight lines in the trace plots). This is further confirmed by panels (A) through (D) in figure A.27, where the CP has low effective sample sizes, while an important proportion of the parameters is above the recommended `Rhat` threshold. In contrast, the NCP chains for the sub-dimensions achieve ergodicity, as it is shown in figures A.12, A.14, A.16, and confirmed by the `n_eff` and `Rhat` values in figure A.27.

Third, similar visualizations for the CP and NCP structural regression parameters are shown in appendix figures A.17 and A.18. By a careful inspection of the CP trace plots, one can notice we still have chains that are "stuck" in specific part of the posterior distribution (see green lines). As a result, the CP registered lower effective samples sizes than the NCP, as figure A.28 confirms. However, neither of those things prevented the CP achieving convergence, as the `Rhat` in the same figure confirms. In contrast, the NCP parameters seem to achieve stationarity, convergence, and good mixing without any issues, with higher effective sample sizes and `Rhat` values close to one.

Lastly, the performance assessment plots for the sub-dimensions' correlations of the FOLV model are shown in appendix figures A.19 and A.20. In both parametrizations, we observe the correlations seem to achieve stationarity and convergence. However, it also seems they do not manage to make a proper investigation of the posterior. This is further confirmed by panels (B) and (D) from figure A.28, where we see observe the parameters' chains remain below the `Rhat` recommended threshold, indicating convergence; although, the effective sample sizes were low, indicating highly auto-correlated iterations.

So far we have only compared the CP and NCP performance of the FOLV model, for specific replicas, and a simulated sample size of 100. However, after the inspection of the trace, trank, ACF and comparison plots[2], we notice the preceding patterns of stationarity, convergence and mixing extends to all replicas, simulated sample sizes and parametrizations of the FOLV model. Furthermore, the aforementioned patterns were also observed in a similar parameter set the SOLV shares with the FOLV model.

For the remaining parameters in the SOLV model, that is, the loadings and reading comprehension higher-order latent variables, we continue observing the same recurring patterns. Appendix figures A.21 and A.22 show the NCP loadings have chains that seem slightly more stationary, convergent, and well mixed than its CP counterpart. This is further confirmed by panels (B) and (D) from figure A.29. In a similar fashion, figures A.25 and A.26 show the higher-order latent variable is significantly more ergodic under the NCP than the CP. Figure A.30 confirms the hypothesis, by showing it has larger effective sample sizes, with `Rhat` values always close to one.

Finally, panel (A) of figure A.31 show the pattern of ergodicity for the regression parameters is slightly improved under the SOLV model. The chains show larger effective sample sizes under the NCP versus the CP, indicating less correlation among the iterations, and therefore, better mixing. Ultimately, the patterns for this set of parameters was also consistent across replicas and simulated sample sizes.

Consequently, all of the above just show the NCP largely improved the performance of the MCMC chains under our implementation, for all models and almost all of the parameters. It is important to point out, the result did not extend to the sub-dimensions' correlation parameters, where no large difference in performance was observed between

[2]To inspect the remaining trace, trank, ACF, and comparison plots follow the github accompanying page detailed in Appendix A.2.2.

the CP and NCP. On this matter, the issue was not related to a lack of identification of said parameters, as we were careful to ensure the fulfillment of this requirement (see next section for more evidence).

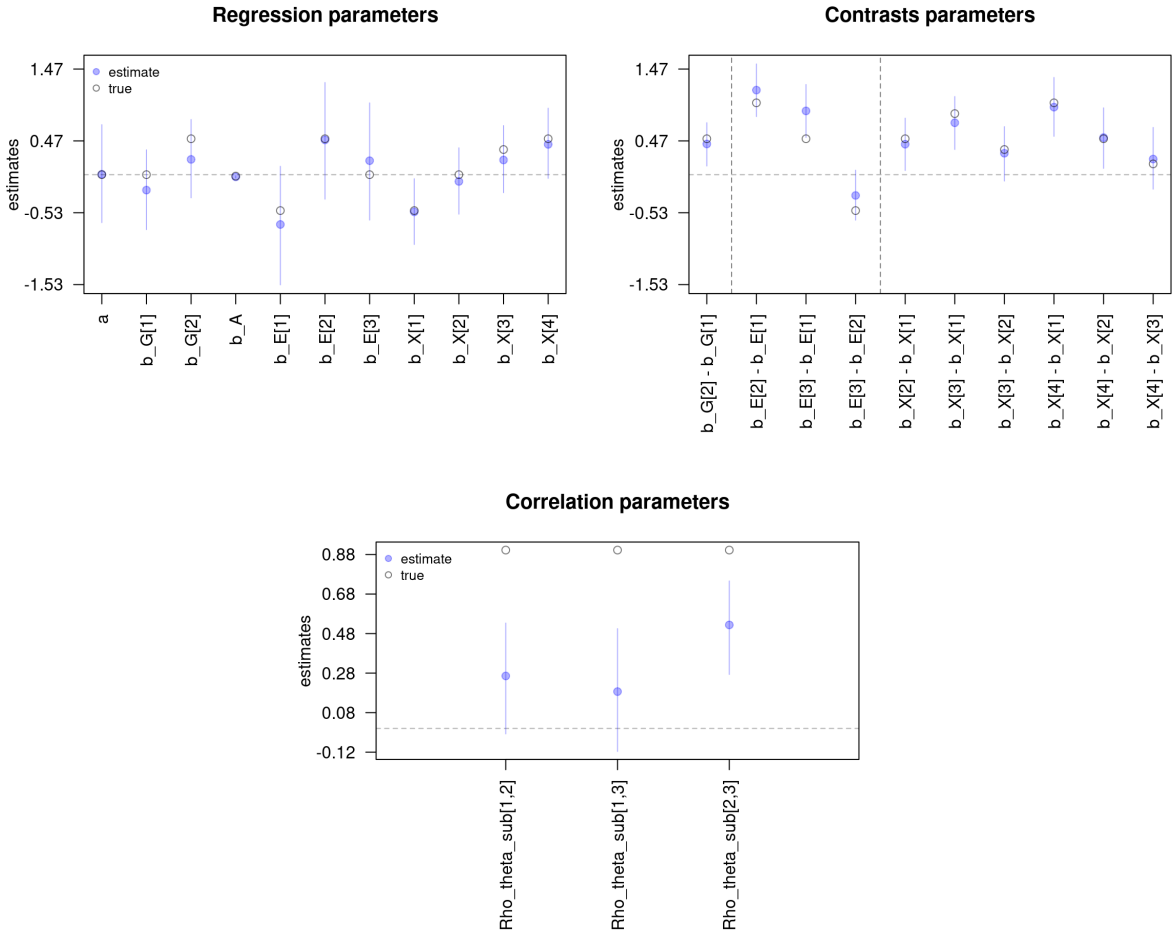

Figure 4.9: First-order latent variable model (FOLV). Centered parametrization. Sample size 100, replica 1. Regression, contrast, and correlation parameters. The "true" correlation parameters were calculated using Wright's tracing rules, and corresponds to an approximate unconditional correlation.

4.6.2 Recovery capacity

Figures 4.9 and 4.10 show the CP and NCP recovery of the regression, contrast, and correlation parameters for the first replica of the FOLV model, with a sample size of 100. From both figures we notice the regression and contrast parameters were estimated appropriately. The true values were located within the compatibility interval of the estimates. Moreover, no remarkable difference is observed between the CP and NCP implementations. This result extends across all replicas and sample sizes, as the recovery plots[3] confirm. Furthermore, table A.4 (appendix) reveal the RMSE_B of the structural regression parameters under NCP were consistently lower than the CP counterpart, but the

[3]To inspect the remaining recovery plots follow the github accompanying page detailed in Appendix A.2.3.

differences were negligible. In a similar fashion, table A.5 show a similar pattern in the contrast parameters, between the CP and NCP; however, we also observe the contrasts were recovered with better precision with larger sample sizes, no matter the parametrization.

On the other hand, in the same figures, we notice the CP and NCP correlation parameters were estimated similarly far from the approximate unconditional correlations. The unconditional correlations were calculated using Wright's tracing rules [5] using the simulated loadings $\lambda_d^{(2)} = 0.95$, therefore, the approximate correlation was $0.95 \times 0.95 = 0.9025$. However, the comparison is misleading because the correlation parameters produced by the model were not unconditional. Recall the FOLV model used regression covariates to explain some of the variability in the sub-dimensions, therefore, the reported correlations are the residual correlations, after controlling for the aforementioned covariates. In that sense, we notice a non-negligible (and sometimes significant) amount of correlation remains present among the sub-dimensions, apparently due to the miss-specification of the model. Again, this behavior is consistent across replicas, and samples sizes, with no discernible difference among parametrizations (see table A.6, appendix).

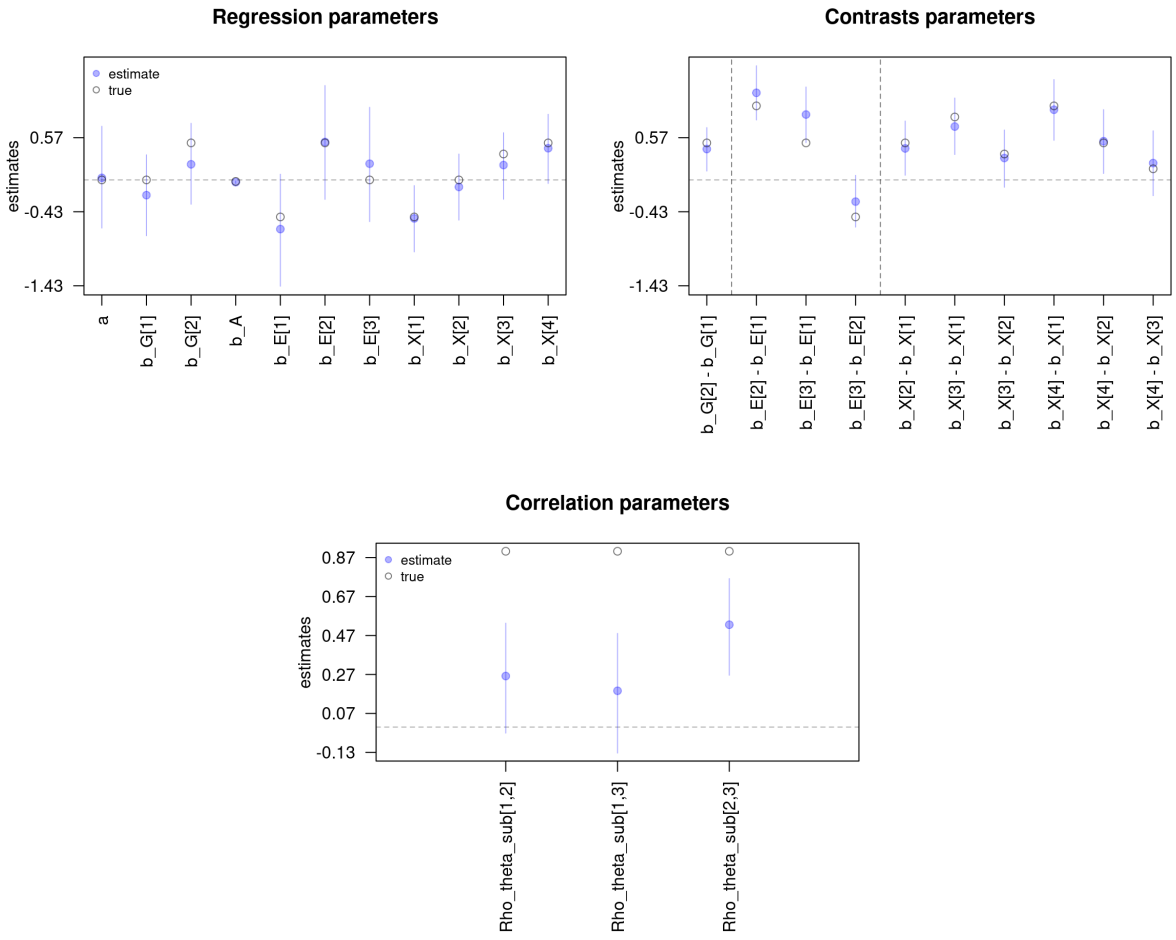

Figure 4.10: First-order latent variable model (FOLV). Non-centered parametrization. Sample size 100, replica 1. Regression, contrast, and correlation parameters. The "true" correlation parameters were calculated using Wright's tracing rules, and corresponds to an approximate unconditional correlation.

Furthermore, figure A.32 show the CP and NCP items, text, and text deviation difficulty estimates. As with the previous patterns of recovery, no remarkable difference was observed among the different parametrizations. On both scenarios, the true items and text difficulties were within the compatibility intervals of the bayesian implementation, while the text difficulties and deviations were estimated with greater precision. As in the previous estimates, the patterns were consistent across replicas, and samples sizes, with no discernible difference among parametrizations (see tables A.8 and A.9, appendix).

A behavior somewhat similar is observed for the SOLV model. Figures A.33 and A.34 (appendix) show the CP and NCP regression, contrast, correlation and loading parameters. From the figures we notice the regression parameter were estimated equally well by both parametrizations. However, we notice in general, the model had a harder time to estimate the regression, contrast and correlation parameters, compared to the FOLV model. In the case of the structural regression parameters, table A.14 show the SOLV model had larger levels of errors for sample sizes of 500. For the contrasts, on the other hand, table A.15 reveals the model estimated all parameters with higher error. This is particularly true for the contrast related to the education covariate. Moreover, one would think that, given the model is now correctly specified, the estimated correlations could be close to the true simulation parameters, but we would be wrong; and table A.16 hides this fact. From a careful inspection of the recovery plots[4], we can notice the SOLV model estimated non-negligible positive correlations for some of replicas with sample sizes greater than 100. This just indicates that for larger sample sizes, some residual correlation remains present, even when the model considers a higher order latent variable. One explanation for this could be that the model severely underestimated the loadings. The true values for the parameters were not within any of the compatibility intervals (that is why we registered higher levels of RMSE_B in table A.17). Moreover, one could further argue that this is due to a lingering parameters' lack of identification, however, the evidence from the pairs plot[5] does not seem to support the hypothesis. The pairs plots show loadings and correlation with low levels of interdependence, contrary to the "narrow ridge" pattern expected under the lack of identification hypothesis. Ultimately, in concordance with the FOLV model, these patterns of recovery were consistent across replicas, samples sizes, with no discernible difference between parametrizations.

Finally, appendix A.2.3 provides the FOLV and SOLV tables of the RMSE_B for the latent individuals' dimensions. As with the preceding parameters, no discernible difference in the recovery capacity was observed between the CP and NCP. The results were consistent across replicas and sample sizes, as the several tables confirm.

Nevertheless, it is surprising to observe the CP and NCP have a similar recovery capacity, as one could argue that in the case of simulation based estimation methods, like MCMC, the recovery capacity is highly tied to the performance of the procedure (conditional on the number of chosen iterations). The reader can recall from the previous section, that under the CP, multiple parameter chains fail to achieve ergodicity, therefore, jeopardizing our ability to make valid inferences about them.

One explanation for this could be that both parametrizations achieved what is called a *local convergence* [10], that is, the chains appear to be stable in the range of the iterations (stability observed at least in the NCP). However, given the consistent results across

[4]Not shown in the document. Refer to the corresponding github accompanying page detailed in Appendix A.2.3.

[5]Same as previous footnote.

replicas, the researcher is led to think that even with all the issues present under the CP, the chains managed to visit the posterior distribution in a way, that allowed the method to produce a proper estimation of the parameters. Moreover, we are also led to think, the preceding patterns of recovery capacity are the result of using the HMC algorithm with a higher rejection criteria (`adapt_delta`= 0.99) and weakly regularizing priors. The previous chapter has described the benefits of these factors separately, so it is sensible to assume that used in conjunction, they could benefit the posterior exploration, and therefore, the recovery capacity of the method. Further investigations manipulating these conditions could be of relative importance.

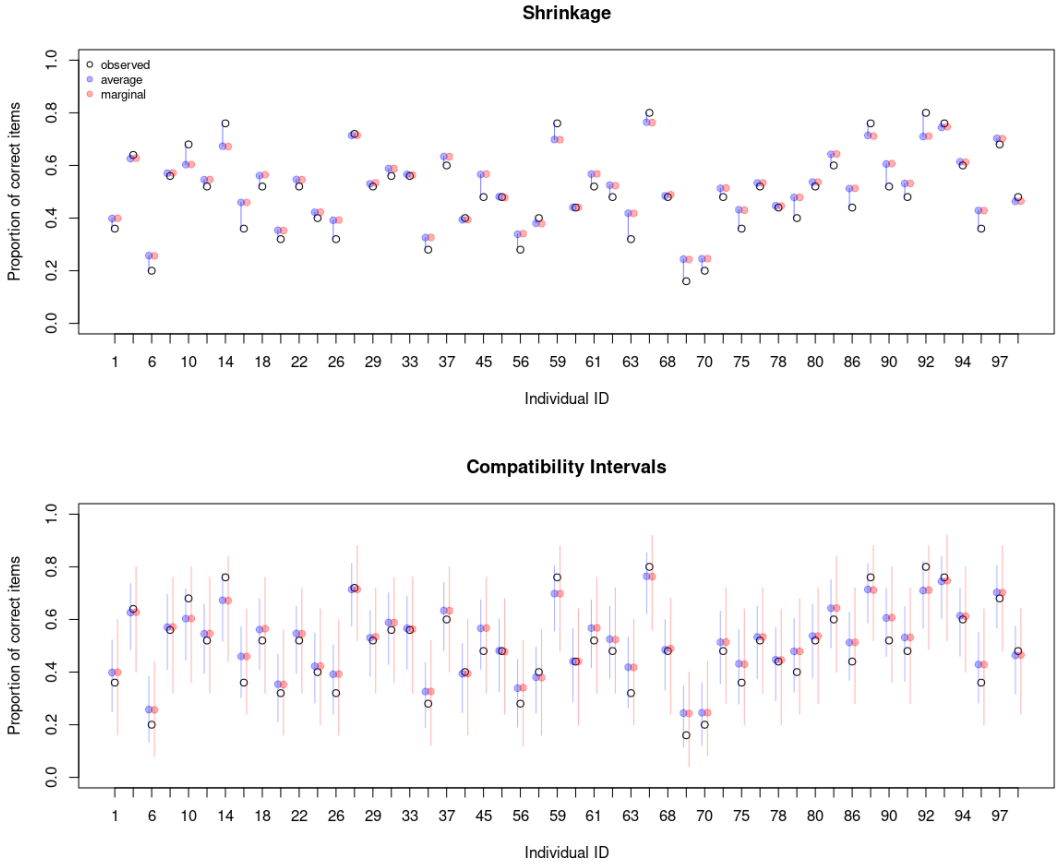

Figure 4.11: First-order latent variable model (FOLV). Sample size 100, replica number 4. Non-centered parametrization. Individual predictive plot.

4.6.3 Retrodictive accuracy

Figure 4.11 shows the true, average, and marginal predicted proportions, on a random sample of individuals, of the fourth replica from the NCP FOLV model with a sample size of 100. The plot also depicts the corresponding shrinkage and compatibility intervals.

From the figure we notice the model manages to capture rather well the traits of the data, while avoiding its exact replication. We observe the true proportion of endorsed items, among individuals, are within the compatibility intervals of the average and marginal predictions, i.e. the true simulated values for the proportions are close to

the predicted mean and outcomes produced by the model. However, we observe the average and marginal predictions show some level of shrinkage, that is, the estimates are "pulled" towards the average true proportion across individuals, in greater or less quantity. This results from the complex pooling of information across individuals, items, texts and dimensions; that allows the model to negotiate between predicting all individuals with the average proportion (under-fitting) or predicting each individual by its own proportion (over-fitting). On the other hand, figure 4.12 only confirms the model manages to capture the traits of the data, as the predicted proportions per individual and covariate combination, are replicated rather well. Similar results can be observed on the predictive accuracy of the model aggregated by dimensions, items, and texts, in figures A.36, A.37, and A.38, albeit the predicted proportion are closer to the true values.

On the other hand, for the same model, figures A.39 and A.40 (appendix) show the true, average, and marginal items characteristic curves, and item information functions. From the figures, we notice the model manages to reproduce rather well the true ICC and IIF curves. This means that the model allow us to correctly recover the item's psychometric characteristics, a trait of high relevance for the development of instruments.

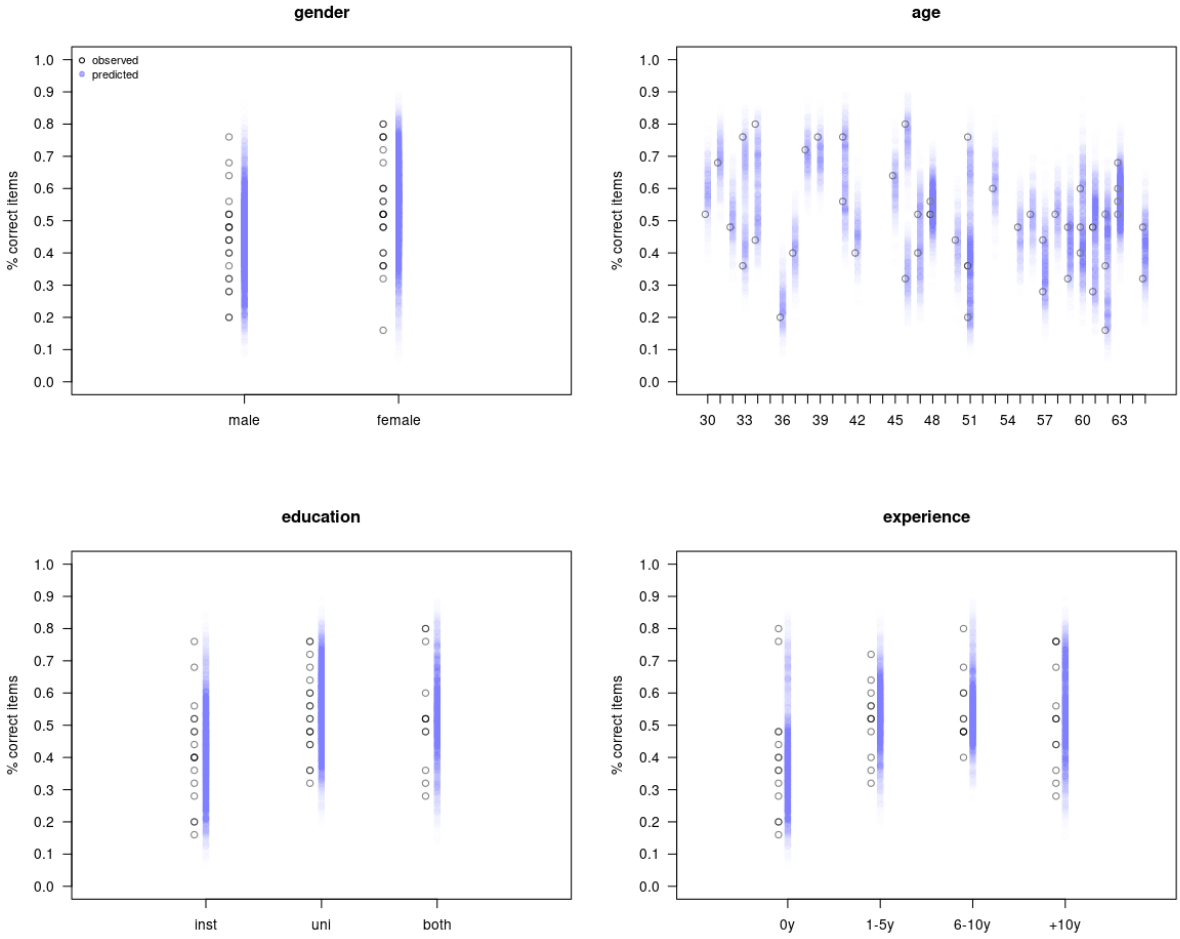

Figure 4.12: First-order latent variable model (FOLV). Sample size 100, replica number 4. Non-centered parametrization. Individual predictive plot per covariate.

Figures A.41 through A.47 (appendix) show the SOLV model obtained similar results. Moreover, a careful inspection of similar plots for all replicas, sample sizes, and parametrizations[6] revealed the preceding patterns of retrodictive accuracy are consistent across the conditions, with no apparent difference between the CP and NCP. This results also resonates with the patterns obtained in the previous section.

Finally, tables 4.1 and 4.2 further confirm that no difference in the individuals' retrodictive accuracy is observed between the CP and NCP, under the FOLV or SOLV model. Moreover, they reveal the predictive uncertainty is higher within rather than between the replicas, that is, the models predicted the data in a similar manner across replicas. A similar behavior is observed for the items' retrodictive accuracy, as tables A.21 and A.22 show.

		Sample	$\overline{\text{RMSE}}_W$			RMSE_B		
	Parametrization	size	mean	min	max	mean	min	max
1	CP	100	0.113	0.106	0.120	0.041	0.018	0.058
2	CP	250	0.114	0.107	0.127	0.041	0.023	0.070
3	CP	500	0.112	0.104	0.125	0.041	0.015	0.076
4	NCP	100	0.113	0.106	0.121	0.041	0.018	0.059
5	NCP	250	0.114	0.107	0.125	0.041	0.023	0.068
6	NCP	500	0.112	0.105	0.126	0.041	0.015	0.077

Table 4.1: First-order latent variable model (FOLV). Centered and non-centered parametrization. Within and between replicas individual predictive RMSE.

		Sample	$\overline{\text{RMSE}}_W$			RMSE_B		
	Parametrization	size	mean	min	max	mean	min	max
1	CP	100	0.112	0.106	0.117	0.034	0.015	0.050
2	CP	250	0.112	0.107	0.122	0.035	0.018	0.059
3	CP	500	0.111	0.104	0.122	0.036	0.015	0.068
4	NCP	100	0.112	0.106	0.117	0.034	0.015	0.050
5	NCP	250	0.112	0.106	0.121	0.035	0.018	0.057
6	NCP	500	0.111	0.105	0.122	0.036	0.014	0.068

Table 4.2: Second-order latent variable model (SOLV). Centered and non-centered parametrization. Within and between replicas individual predictive RMSE.

4.6.4 Time

Although the assessment and comparison of the MCMC procedure's running time is not one of the main goals of the study, the researcher believes it is useful to report them, as they could serve as a benchmark of comparison for future developments.

Tables 4.3 and 4.4 report the CP and NCP MCMC running time, for the FOLV and SOLV model, respectively. As the reader can recall, we decided to use 10 replicas for each combination of model, parametrization and sample size. On each replica, the posterior

[6]To inspect the prediction plot follow the github accompanying page detailed in Appendix A.2.4.

distribution of the parameters was approximated through an HMC procedure with 3 chains (run in parallel), with 2,000 iterations per chain. From the tables, we can see the NCP was faster than the CP for the smallest sample sizes (100 and 250). However, the differences in running time got smaller as the sample size increased. This was true for both type of models.

The NCP being slightly faster than the CP is important, as the non-centered parametrization is more complex, and requires the sampling of more parameters than the centered counterpart. This just means that improving the performance of the MCMC, through a more complex model as the NCP, does not come with a cost on running time.

			Time (min.)		
	Parametrization	Sample	mean	min	max
1	CP	100	4.80	1.89	7.56
2	CP	250	7.85	5.60	16.57
3	CP	500	19.15	17.20	22.53
4	NCP	100	1.94	1.74	2.23
5	NCP	250	7.18	6.78	8.40
6	NCP	500	22.38	20.12	28.62

Table 4.3: First-order latent variable model (FOLV). Running time statistics.

			Time (min.)		
	Parametrization	Sample	mean	min	max
1	CP	100	4.04	1.39	8.55
2	CP	250	7.13	5.22	11.49
3	CP	500	16.14	14.45	19.73
4	NCP	100	1.87	1.62	2.28
5	NCP	250	5.92	5.20	6.79
6	NCP	500	16.58	13.37	18.26

Table 4.4: Second-order latent variable model (SOLV). Running time statistics.

Chapter 5

Application

5.1 Objectives

Considering the GLLAMM model developed in previous chapters, its application on a real data set had a four-fold purpose:

1. **Evaluate the performance of the parametrizations.** We assessed if changing the posterior sampling geometry benefited the performance of the MCMC method.

2. **Evaluate the retrodictive accuracy.** We wanted to assess how well the model retrodicted the data, and what was the evidence in favor of any of the proposed models.

3. **Assess the psychometric properties.** We had a special interest in determine how difficult the items were, and in what part of the abilities measurement range they were located.

4. **Test research hypothesis.** We were interested on testing hypothesis about the explanatory power a set of covariates had on the latent dimensions, and what were the implications of these, for the educational authority's policy decision making.

5.2 Instrument

The evaluation instrument was selected from the Peruvian public teaching career national assessment. The large standardized test took place during 2017, an it allowed the winner to obtain an appointment, or temporal hiring, into the public teaching career of Peru.

The instrument was composed of 90 multiple-choice question, with four alternatives per item. The items were scored on a dichotomous scale, that is, only one of the four alternatives was the correct one. Moreover, the test was organized in three sub-tests designed to evaluate the reading comprehension, mathematical reasoning, and pedagogical knowledge of the applicants. Given the extension and differences between the sub-tests, we decided to focus on the reading comprehension portion, composed of the first 25 items of the instrument.

The reading comprehension sub-test was designed to evaluate the teacher's ability to reconstruct the meaning of different types of texts, presented in diverse formats. The

sub-test had items designed to measure only one of the three hierarchically nested sub-dimensions of reading comprehension: literal, inferential, and reflective abilities. The literal ability items centered its focus on assessing the teacher's capability to locate explicit information on the texts. The inferential items assessed the teacher's ability to integrate the information in texts, with the goal of inferring its theme, purpose or implicit logic relationships. Lastly, the reflective items evaluated the teacher's abilities to critically reflect about the content and structure of texts.

Finally, besides being nested in dimensions, the items were bundled in groups of five, to a common text or passage, that provided the stimulus over which the individual was assessed, i.e. the items were testlets.

5.3 Data

The data set was accessed through the proper legal requirement of open information to the Ministry of Education of Peru (MINEDU). The data was anonymized and transferred through digital mean to the researcher.

Finally, given the large amount of individuals exposed to the aforementioned evaluation (approximately $195,000$), a simple random sample of $2,000$ individuals was taken.

5.4 Hypothesis

On it core, statistical models are neat association engines. Through their use, one is able to detect associations between variables, and estimate their effects. However, when a researcher find itself in the position of trying to determine what are the consequences of intervening on a variable, i.e. infer causes, statistical models are never sufficient. Information outside the data, related to the causal hypothesis between the variables, is always required [45]. Nevertheless, since much of time the statistical endeavor has to do with produce understanding that leads to generalization and application; there must be a reasonable way to state our hypothesis, and to think formally about causal inference.

Luckily, as McElreath [45], Hernán and Robins [29], and several other authors indicate, Graphical Causal Models (GCM) can come to the rescue. The simplest and yet powerful GCM is the Directed Acyclic Graph (DAG). A DAG is a heuristic model that contains information that is not purely statistical, but unlike a detailed statistical model, the graph allow us to deduce which variable relationships can provide valid causal inferences. However, abide by the "no-free lunch" rule, the causal inferences produced under a DAG are only valid if the assumed DAG is correct. On the latter, one would think the aforementioned caveat is an insurmountable critic of the tool, but one would forget that any statistical analysis hinges on assumption, and a DAG is just a tool to make assumptions more transparent. A full description of the GCM's formal theory is beyond the scope of this work. However, if the reader is interested in the topic, he/she can refer to McElreath [45] or Hernán and Robins [29] for introductory level descriptions, and beyond, about the use of GCM for the statistical endeavor.

So, for our current research interest, figures 5.1 and 5.2 show the DAGs of the application's first- and second-order latent variable model (FOLV and SOLV, respectively). The figures aim to reflect the hierarchies and hypothesized dimensional structure of the instru-

ment, while also describing the assumed qualitative relationships among the structural covariates.

A careful inspection of both figures reveal they are similar to the ones implemented in the previous chapter. Therefore, the definitions of the likelihoods, priors, and *hyper-priors* will be similar in nomenclature and even in distributional assumptions, as the simulated counterparts (see appendix B.3). Nevertheless, the previous cannot be fully extended to the structural covariates, as one can notice the models have a different set of variables than their simulated counterparts. Consequently, in this section we proceed to define them and outline their assumed causal hypothesis.

It is important to emphasize, the hypothesis presented in this section, and the results presented in section 5.5.4, will take a diagnostic perspective, that is, we will emphasize on what factors does the educational authorities will need to give priority, to ensure a fair and equitable development of teachers.

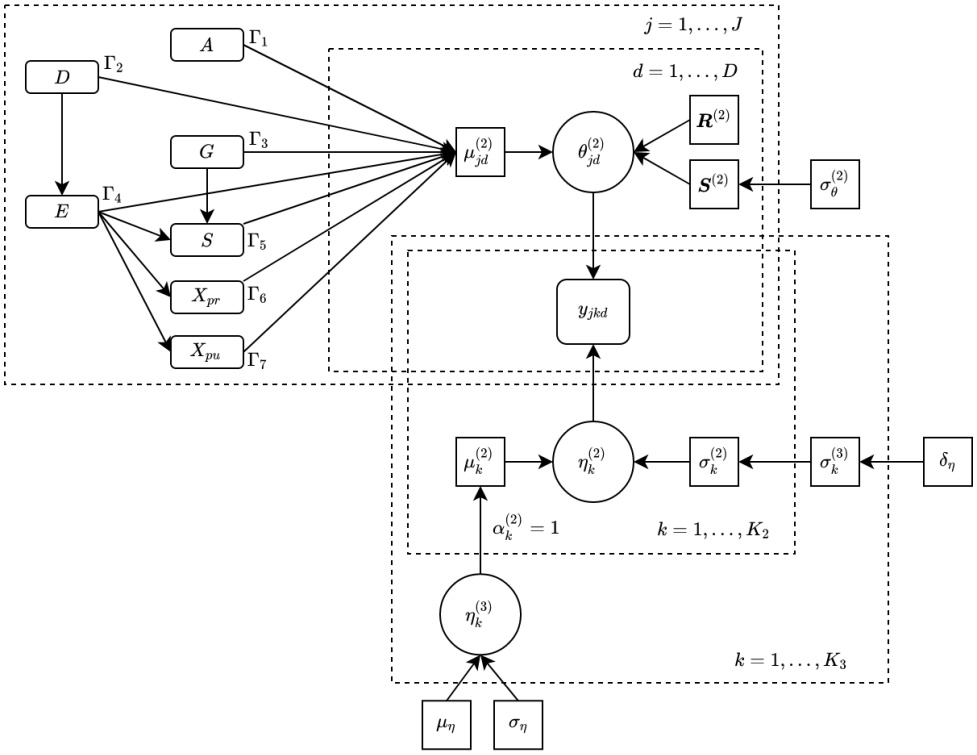

Figure 5.1: Directed Acyclig Graph (DAG). Application's first-order latent variable model (FOLV). Circles represent latent variables. Squares represent parameters or parameters for priors. Large squares represent nesting in specific units.

First, we assumed the reading abilities and/or sub-dimensions were affected by age (A). In this particular case, age was used as a proxy for the applicant's style of teaching. As it is point out by the current National Basic Regular Educational Curriculum of Peru[1], approximately forty years ago, an individual was considered literate if he/she had acquired the basic knowledge of reading, writing, and performing mathematics; while having a preliminary exposure to trades's dexterity and abilities. Much of this has been changing

[1]National Basic Regular Educational Curriculum (2017). URL: `http://www.minedu.gob.pe/curriculo/pdf/curriculo-nacional-2016-2.pdf`

throughout the years, and although reading and writing abilities remained important, the criteria to determine if a person is literate, now goes beyond assessing if an individual knows the basic of how to read, write, or apply mathematics. Therefore, it is sensible to assume, age could inform us if a participant has a style of teaching based on the previous requirements of literacy (what we dubbed as the "old" curricula). The idea is that older applicants, having developed most of their educational careers under the "old" curricula, now being assessed under the new one, would register lower levels of reading comprehension. However, since the evolution from the "old" to the new curricula is not a binary scenario, i.e. you either are in one or the other; but rather a continuous evolution, we reflect this idea by allowing age explain the abilities on a continuous manner, more specifically, a linear one. Off course, since age is associated with several other factors, e.g. cognitive development, we understand the previous hypothesis would not be the only one explaining the outcome. However, we believe that by measuring this effect, albeit proxy, educational authorities could be able to target in-training services to individuals with these characteristics.

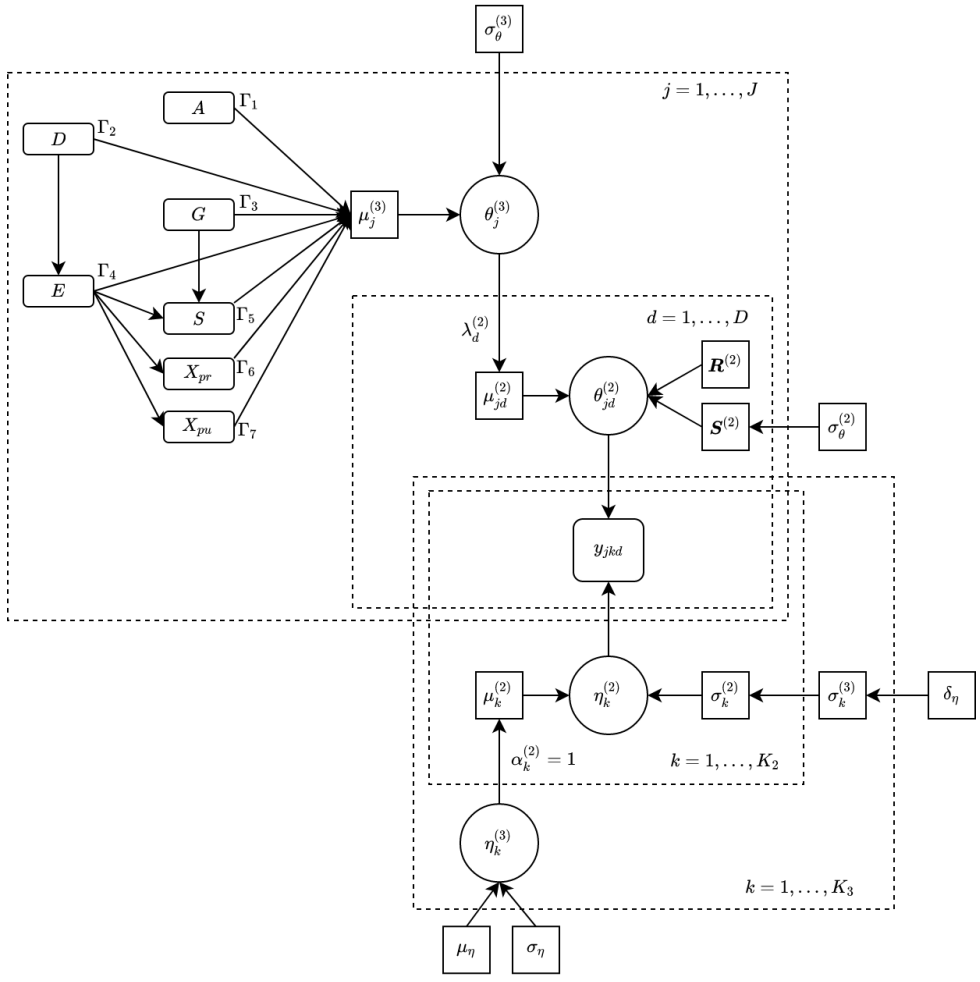

Figure 5.2: Directed Acyclig Graph (DAG). Application's second-order latent variable model (SOLV). Circles represent latent variables. Squares represent parameters or parameters for priors. Large squares represent nesting in specific units.

Second, we assumed disability (D) causally explained the levels of reading compre-

hension and/or its sub-dimensions. It is not hard to imagine an scenario where this relationship is true, and a good example of this, is people with vision disabilities. As individuals with vision disabilities are forced to learn completely different set of tools and strategies to be able to read and write, e.g. learn Braille or receive text-to-speech support, the possibility that this additional requirements have hindered the individual's development of the reading comprehension abilities is high. Moreover, the possibility that these effect are reflected on their evaluation outcomes is also highly likely. At this point, it is important to indicate that MINEDU provide the means for people with vision disabilities, to be evaluated under equitable conditions, e.g. using evaluation instruments written on Braille or providing text-to-speech support. However, we believe the effects of the disability could offset the effects of the conditions set in place to ensure an equitable evaluation, but we do not know by how much. Furthermore, although the justification to include this variable in the model have been guided by the previous example, one cannot discard that other types of disabilities could also had an impact on the development of the reading comprehension abilities. In this sense, we believe the educational authority only benefits from obtaining this type of evidence, to either target in-training services for these individuals, or to improve further their evaluations conditions, to ensure their participation on the educational workforce.

Third, we assumed the type of education an applicant received (E) affected the reading comprehension abilities, where type of education was a categorical variable that denoted if an individual received his/her pedagogical training from an institute, university, or both. The assumptions behind the inclusion of the variable derives from incidental evidence about the quality of training on pedagogical institutes. In Peru, there is silent, but widely accepted notion that most pedagogical institutes produce teacher with low levels of pedagogical abilities. Multiple reason have been considered, that is, a poorly devised or outdated curricula of teaching; their lack of funding; the fact that mostly institutes are specialized on training people with disabilities; or the fact that because institutes are easier to enter and less expensive than private universities, people coming the lower tiers of income (another proxy of educational opportunities and development), select institutes to begin their education. Considering the previous, it could be of interest for the educational authority to set this incidental evidence on a more quantitative ground, again with the purpose of providing remedial measures.

At this point, it is important to indicate that from the previous paragraph statements, one can easily assume a backdoor path exists between the education and disability variables (see the path $D \rightarrow E$ on the model's figures). A backdoor path is just a DAG's path that outlines a relationship between two variables, that if remains uncontrolled for, can confound the variables' effects on an outcome. In our specific case, this just means that, in order to obtain unbiased estimates for the effects of education and/or disability, both variables need to considered in the model, to close any of the backdoor paths.

Fourth, we assumed the educational career experience, measured in years of public or private teaching (X_{pu} and X_{pr}, respectively), affected the reading comprehension abilities. In this case, the causal assumption goes in line with the idea that with more teaching experience, the teacher had more opportunities to develop key aspects of the reading comprehension abilities. However, as it happened previously, we believe a backdoor path exist between these covariates and the educational variable. The intuition of these relationships derives from the same incidental evidence relating the type and quality of the pedagogical training. One can imagine, if the pedagogical training in institutes is perceived as of

lesser quality, the opportunities a teacher can access, if any, will also be of less quality. We believe the latter is especially true in the private sector, where this perception can also be translated in less years of experience. However, since we are interest in estimating the unbiased and independent effects of education and experience, both set of variables need to be included in the model, as they already are.

Fifth, we assumed the broad definition of teaching specialty (S) also explained causally the reading comprehension abilities and/or its sub-dimensions. Specialty was a categorical variable denoting if a teacher had the credentials to instruct at the early childhood, primary, or secondary educational levels. The intuition behind its inclusion, lies in the idea that instructing students that require more complex levels of written and oral communication, could end up benefiting the reading comprehension abilities of teachers.

Notice from the model's figures, that we additionally assumed, there is a backdoor path between the specialty and gender variables. In this case, this relationship only tries to reflect that it is more likely that women choose to specialize on the early childhood and primary educational levels. However, it is important to indicate that gender only enters the model as a control variable, following a common practice in the literature.

Therefore, after declaring our hypothesis, we proceed to declare the prior model assumptions related to the aforementioned covariates. We state our current lack of knowledge about the effects of the variables, by setting weakly informative priors, in the following way:

$$\Gamma_1 \sim \text{Normal}(0, 0.5) \tag{5.1}$$
$$\Gamma_{2c} \sim \text{Normal}(0, 0.5) \tag{5.2}$$
$$\Gamma_{3c} \sim \text{Normal}(0, 0.5) \tag{5.3}$$
$$\Gamma_{4c} \sim \text{Normal}(0, 1) \tag{5.4}$$
$$\Gamma_{5c} \sim \text{Normal}(0, 0.5) \tag{5.5}$$
$$\Gamma_{6c} \sim \text{Normal}(0, 0.5) \tag{5.6}$$
$$\Gamma_{7c} \sim \text{Normal}(0, 0.5) \tag{5.7}$$

where as in the previous chapter, c denotes the categories inside the covariate, different for each variable.

Finally, by no means we state the proposed model is the "correct" model to analyze the data. This is particularly true, if we consider that several unobserved important variables are absent from the analysis, e.g. if the individual followed training courses beyond their initial education, or something as simple as, the amount of hours the individual spends on reading. Moreover, this is aggravated by the fact that a good portion of the variables used in the analysis come from self-reported measures, like experience and education. However, we believe that giving the constraints of the data, the model provides a sensible depiction of the causal hypothesis among the available variables.

5.5 Results

5.5.1 Parametrization performance

Figures 5.3 and 5.4 show the item parameters' trace, trank, and ACF plots for the centered and non-centered parametrization of the FOLV model. From a quick inspection of the

figures, we realize the items' parameter did not achieved ergodicity under the CP, in contrast to the NCP. This result if further confirmed by figure 5.5, where we observe the items parameters had effective sample sizes below 100 under the CP, while above $1,500$ under the NCP. Moreover, the parameters' chains registered `Rhat` values well above the recommended threshold under CP, but maintained values close to one under the NCP.

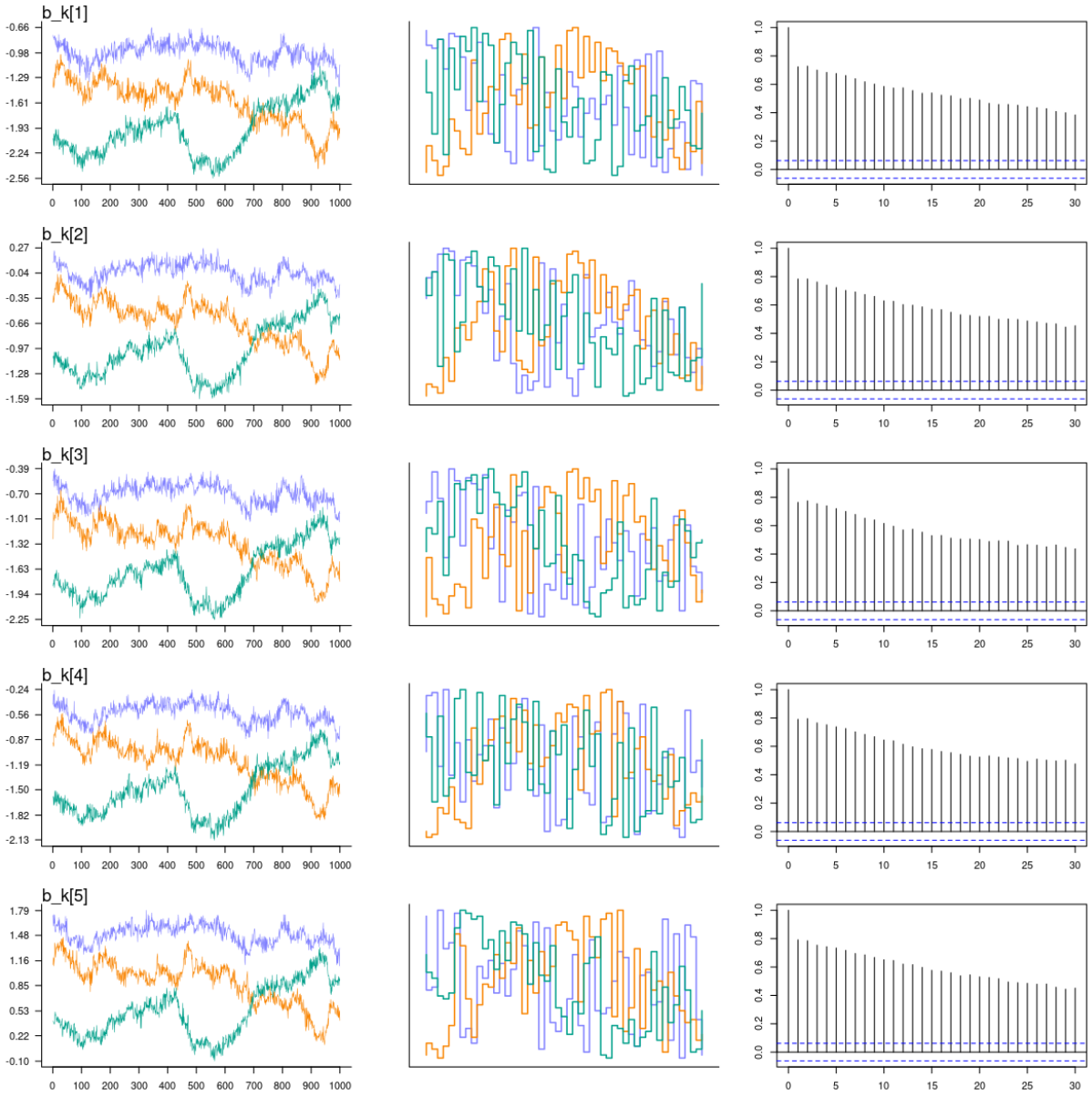

Figure 5.3: Application's first-order latent variable model (FOLV). Centered parametrization. Items difficulty: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

A similar behavior was registered for the texts difficulties and deviations, and individuals abilities. From the inspection of the appropriate figures[2], we notice the CP did not achieved stationarity, convergence, nor good mixing, albeit the patterns were less extreme

[2]To inspect the figures refer to the github accompanying page detailed in Appendix A.3.

than in the previous scenario. In contrast, the NCP achieved ergodicity without any issues. Again, this was further confirmed by the comparison of the `n_eff` and `Rhat` values between the two parametrizations.

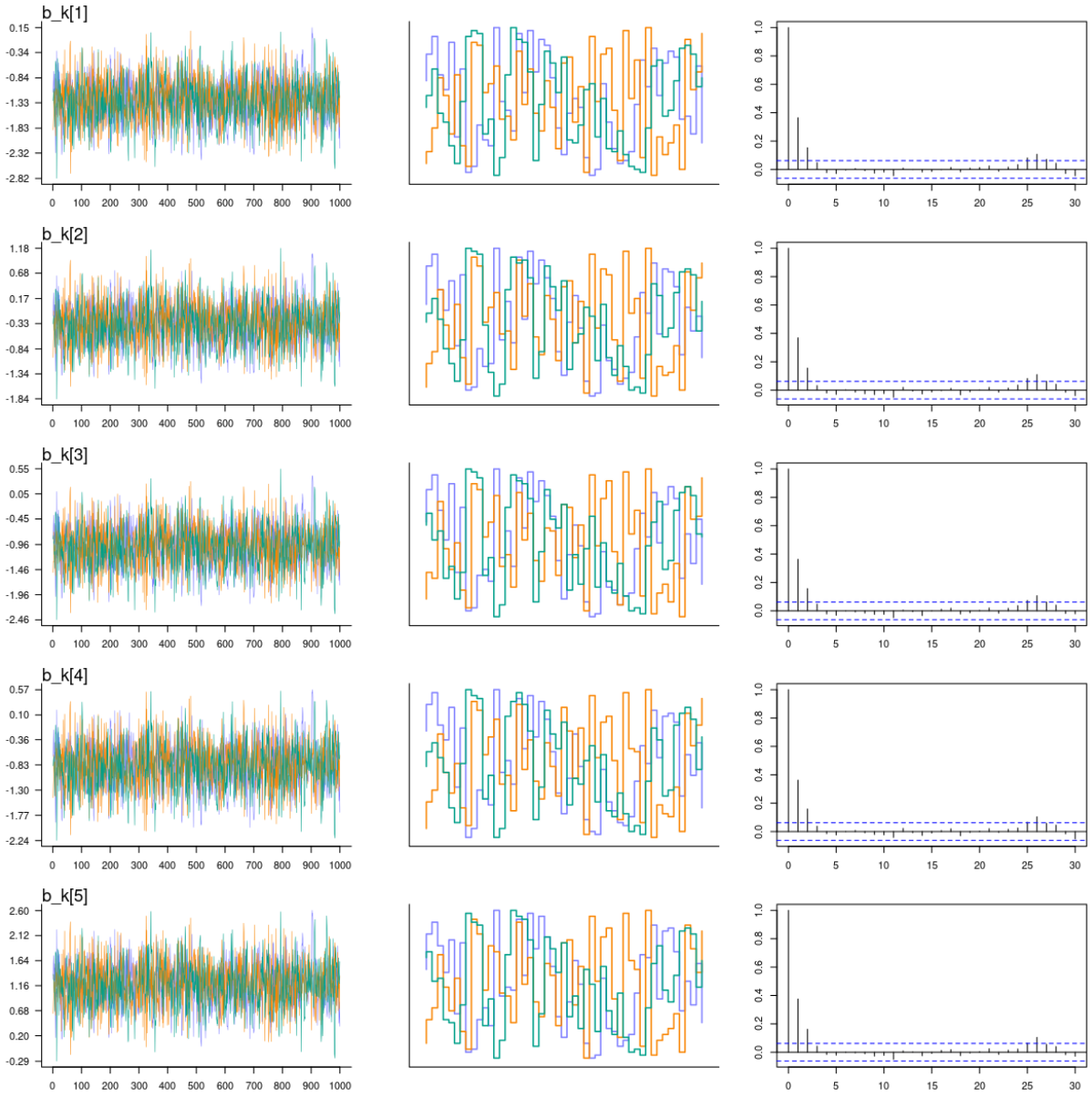

Figure 5.4: Application's first-Order latent variable model (FOLV). Non-centered parametrization. Items difficulty: (Left) trace plot, (Middle) trank plot, (Right) autocorrelation plot.

Moreover, both parametrizations had a hard time exploring the posterior distribution of the sub-dimensions correlations. Inspecting the appropriate figures (not shown), we noticed the CP registered effective sample sizes below 100, while the NCP barely reached sample sizes above 300. This, in conjunction with the trace, trank, and ACF plots further confirmed the chains did not achieved ergodicity.

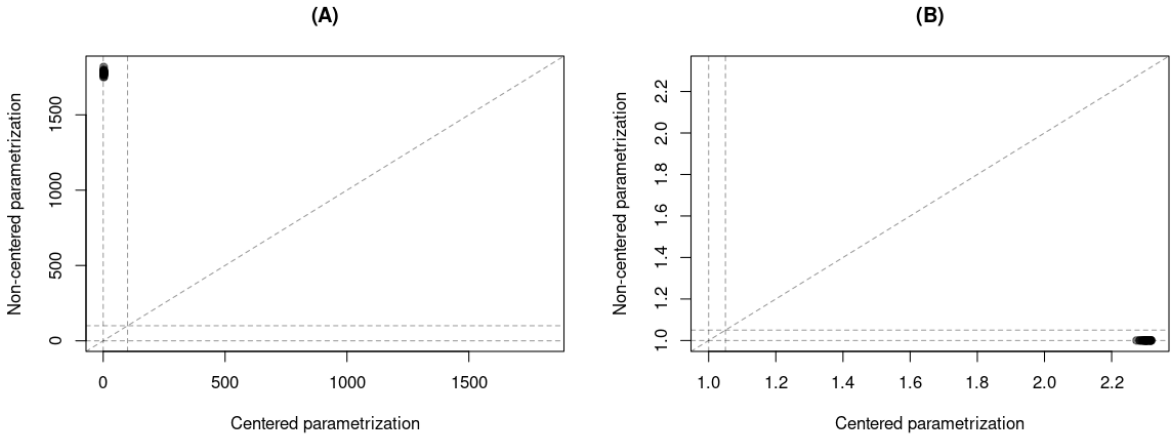

Figure 5.5: Application's first-order latent variable model (FOLV). CP and NCP comparison plot. (A) `n_eff` for items' difficulties. (B) `Rhat` for items' difficulties. Diagonal discontinuous line describes equality between CP and NCP. Vertical and horizontal discontinuous lines set in A corresponds to `n_eff`= 100. Vertical and horizontal discontinuous lines set in B corresponds to `Rhat`= 1.05.

Strikingly similar patterns were observed in the same set of parameters the SOLV shares with the FOLV model. Part of the evidence of such statement can be found in figures A.48 and A.49 (appendix). Moreover, the performance patterns of the loadings were similar to the ones observed for the correlations.

All of the preceding results resonate with the performance patterns obtained in the previous chapter. However, in contrast with the results of the simulation study, some of the structural regression parameters (not shown) registered better performance under the CP, rather than the NCP. Among these set of covariates were age, disability, and both of the experience variables. Nevertheless, the performance improvement was not unanimous, as the parameters showed "healthier" chains with larger effective sample sizes, but `Rhat` values above the recommended threshold, indicating a lack of convergence. This result resonates with Papaspiliopoulos et al. [52], who stated that the success of the NCP strategy was largely dependent on the specifics of the models and data. Moreover, it is surprising this pattern of behavior was not replicated on the SOLV model, where all of the structural parameters seemed to be more ergodic under the NCP. The latter lead us to think, the previous patterns resulted from the use of a possibly miss-specified model.

In conclusion, given the results obtained in this section, and section 4.6.1 of the previous chapter, conditional on the selected number of iterations, the non-centered parametrization was the only one to ensure the GLLAMM parameters attained ergodicity.

5.5.2 Retrodictive accuracy

Much similar to what was the observed in section 4.6.3 of the simulation study, we notice the model manages to capture the traits of the data, while avoiding its exact replication. Figure 5.6 show the individuals' true proportion of endorsed items, which are within the compatibility intervals of the average and marginal predictions, that is, the observed values for the proportions were close to the predicted mean and outcomes produced by

the model. Similarly, we also observe the average and marginal predictions show some level of shrinkage, i.e. the estimates were "pulled" towards the average true proportion across individuals, in greater or less quantity. As explained in the simulation chapter, this results from the complex pooling of information across individuals, items, texts and dimensions; that allows the model to negotiate between predicting all individuals with the average proportion (under-fitting) or predicting each individual by its own proportion (over-fitting).

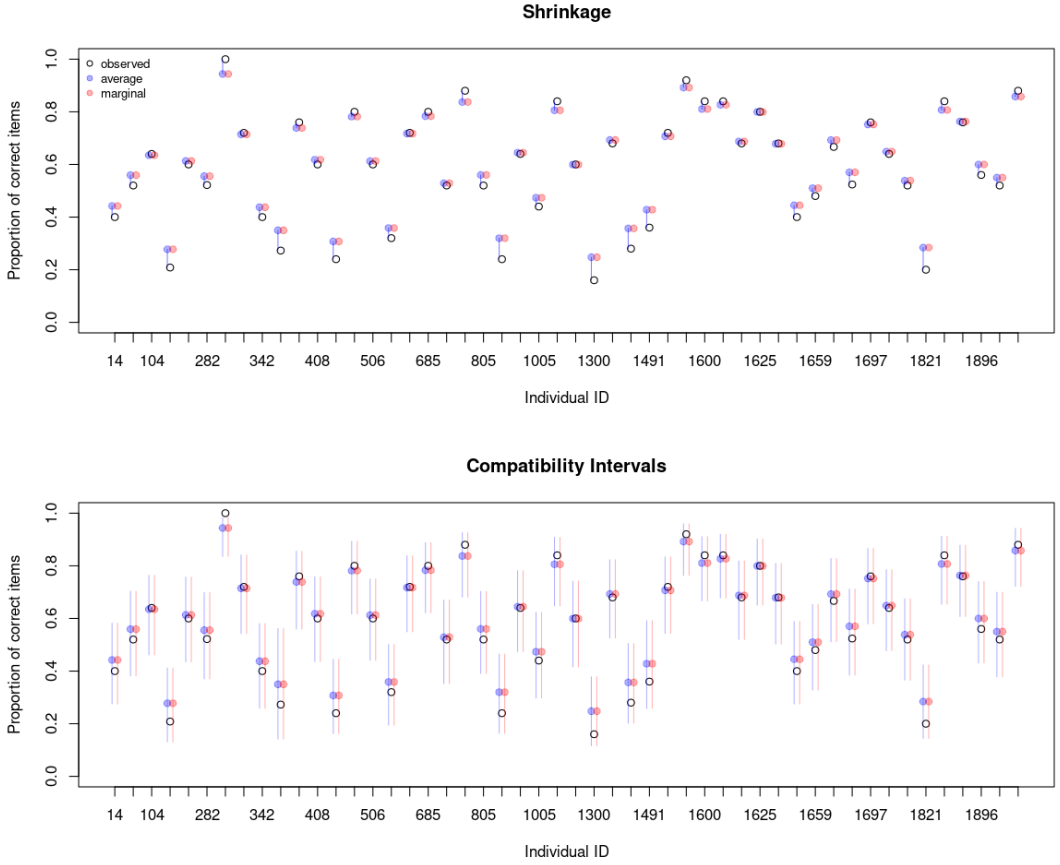

Figure 5.6: First-order latent variable model (FOLV). Non-centered parametrization. Individual predictive plot.

In a similar fashion, inspecting the appropriate figures (not shown), we can say the retrodictive accuracy of the FOLV model extends to the predicted proportions per individual and covariate combination, as well as proportions aggregated by dimensions, items, and texts[3]. Furthermore, the SOLV model managed to do a similar job in all of the previous descriptions, as it is evident from figure 5.6.

Finally, regarding the retrodictive accuracy and model fit, what remains to determine is if the statistical evidence favors one model over the other. This is particularly important, if we notice the SOLV model have approximately $2,000$ more parameters than the FOLV, that is, the reading comprehension abilities of the individuals (not the sub-dimensions). In that sense, a lurking fear of over-fitting the data with the SOLV model is present.

[3]To inspect the figures refer to the github accompanying page detailed in Appendix A.3.

Additionally, since we implemented our model in a Bayesian framework, the uncertainty of the model fit now has two faces, one resulting from the uncertainty of the parameters, and one coming from to uncertainty of the outcome's likelihood. Therefore, a proper assessment of a model fit would needs to consider these two facets. Lastly, since comparing models based on their training data fit could lead us to wrongful conclusions, we need a method that evaluates our models out-of-sample or through cross-validation, i.e. on data not used to train the model's parameters.

Luckily, as McElreath [45] indicate, is a benign fact of the statistical theory, that we do not need to re-fit the model multiple times to obtain approximately valid cross-validation measures. Moreover, he continues, is also useful that there are statistics that uses all the information of a "model's distribution" to asses its fit, and this measures are called information criteria.

It is safe to say that the theory behind the Kullback-Leibler divergence (KL divergence) [37], and the field of of information theory in general, from which the information criteria derives, is way out of the scope of the current research. However, the reader can rest assure the information criteria fulfills all the previous requirements to make the proper comparison of our models, more specifically the Pareto-smoothed importance sampling cross-validation (PSIS) [69] and the Widely Applicable Information Criterion (WAIC) [71]. Additionally, the reader should keep in mind that as these measures try to approximate the out-of-sample KL divergence, is not the absolute magnitude that matters, but the difference among them.

	Model	Parametrization	WAIC	lppd	penalty
1	FOLV	CP	54,632.4	-25,429.7	1,886.5
2	FOLV	NCP	54,631.7	-25,427.6	1,888.3
3	SOLV	CP	54,610.3	-25,348.4	1,956.7
4	SOLV	NCP	54,614.7	-25,337.9	1,969.4

Table 5.1: Model fit. Widely Applicable Information Criterion (WAIC).

	Model	Parametrization	PSIS	lppd	penalty
1	FOLV	CP	54,657.4	-27,328.7	1,904.4
2	FOLV	NCP	54,656.9	-27,328.5	1,898.1
3	SOLV	CP	54,627.3	-27,313.7	1,940.1
4	SOLV	NCP	54,642.5	-27,321.2	1,990.2

Table 5.2: Model fit. Pareto-smoothed importance sampling cross-validation (PSIS).

Tables 5.1 and 5.2 show the WAIC and PSIS statistics, respectively. From the tables we notice all models and parametrizations seem to achieve similar levels of out-of-sample fit. Moreover, the penalties of the statistics indicate all the models have similar levels of over-fitting risk, i.e. all models have a similar risk of encoding too much of the data. As previously mentioned, this is important, because the difference between one model and the other is about $2,000$ parameters.

Consequently, from the results, wa can safely say that both models produced similar encodings of the data. Therefore, the decision of choosing one over another rest now on a more theoretical ground.

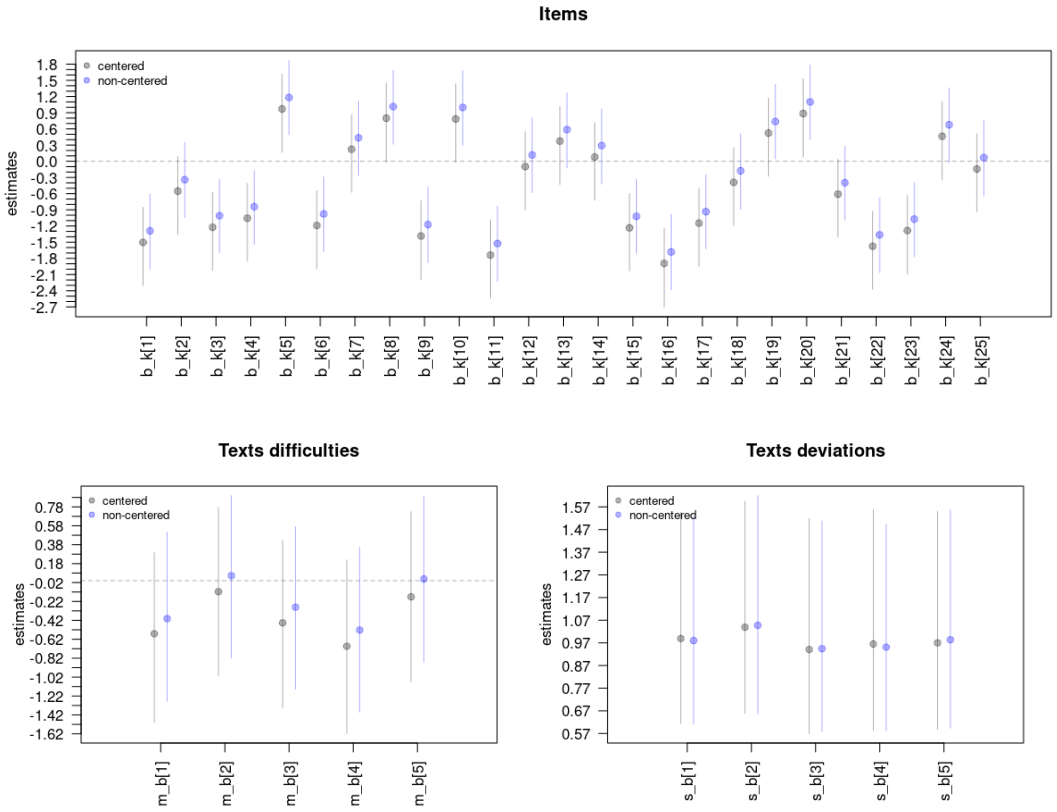

Figure 5.7: Application's first-order latent variable model (FOLV). Centered and non-centered parametrization. Items, texts difficulties, and texts deviations.

5.5.3 Psychometric properties

As in any standardized evaluation, instrument developers have a special interest in determine how difficult the items were, and in what part of the abilities measurement range they were located. From the top panel of figure 5.7, we notice that the items were scattered throughout a significant portion of the abilities range. However, we also notice the items were more prevalent on the lower tiers, with items located as low as 2.0 logits.

Furthermore, an specific benefit of the current implementation also allowed us to assess how difficult the texts were. The bottom left panel of figure 5.7 show the texts difficulties we located around -0.5 logits, with large confidence intervals, indicating the items within the texts were scatter in a wide range around the text mean.

It is important to mention, that in order to provide a sound analysis of the psychometric characteristics of the items, and what they imply for the instruments developers, one would need to have a access to the items and texts description. However, in this case, it was not possible due to the legal restrictions surrounding this information.

5.5.4 Test hypothesis

Finally, we proceed to assess the statistical evidence behind the hypothesis stated in section 5.4. It is important indicate that, more than having an interest in the actual parameter levels, e.g. the reading comprehension estimate for males and females, we had

an interest in the contrast between the parameters, that is, the difference between females and males. The reason for the latter rest on the fact that, testing the hypothesis for the levels would correspond to test how far those categories are from zero, much like in the usual frequentist null hypothesis testing ($H_0 : \beta = 0$). However, in our case, zero does not carry any critical significance for the levels, because we have arbitrarily assumed the individual's latent variables are normally distributed with mean zero (the intercept) and variance one. On the contrary, for our purposes., testing if the difference between two levels is zero ($H_0 : \beta_2 - \beta_1 = 0$) does carry inference value.

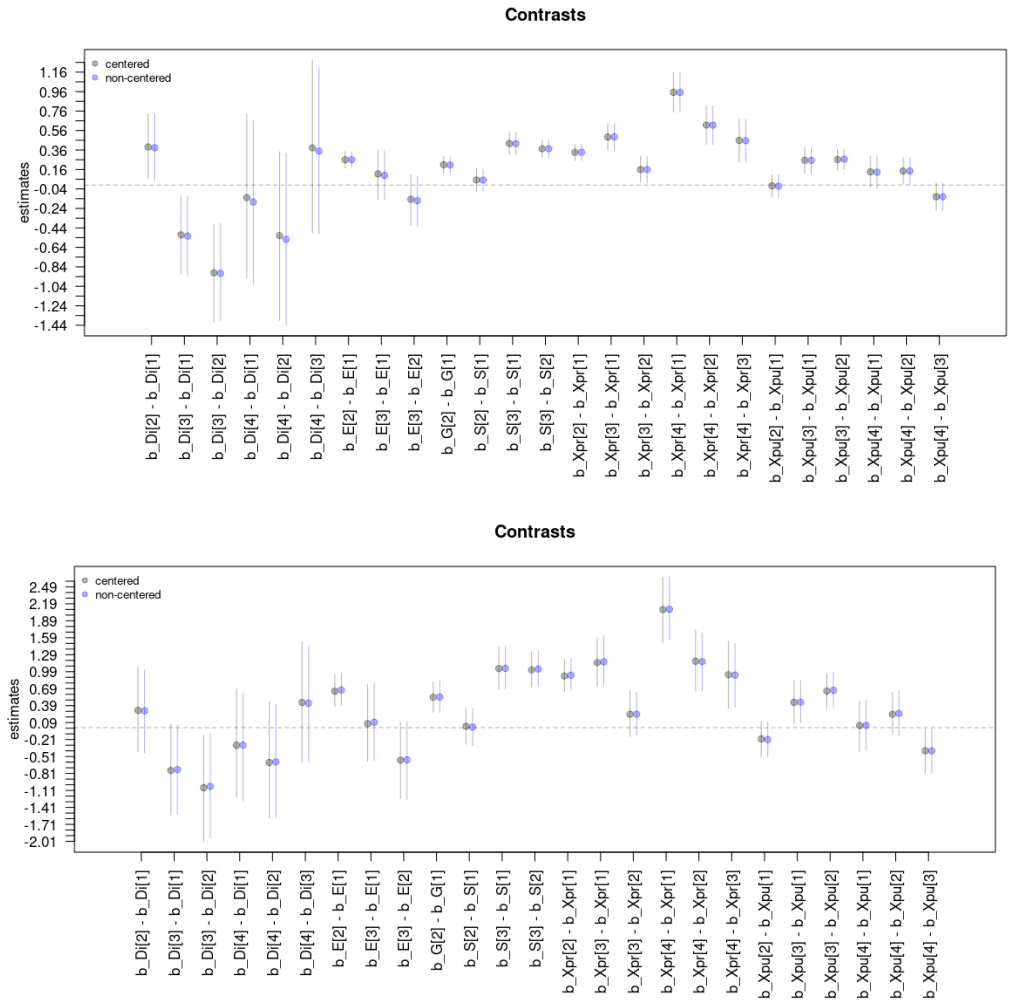

Figure 5.8: Application's first- and second-order latent variable model. CP and NCP comparison plot. (Top panel) Contrasts in the FOLV model. (Bottom panel) Contrasts in the SOLV model.

Figure 5.8 show the full set of contrast resulting from the implemented model. The first thing to notice, between the FOLV and SOLV models, is the difference in scale on which the effects are measured. We notice the evidence in favor or against a hypothesis is larger under the SOLV model. We argue this is due to the issues related with the estimation of the loading and correlations, which in this case, followed the results obtained in the previous chapter. As the reader can recall, in the previous chapter we observed that for larger sample sizes, the SOLV model estimated low loadings but kept higher levels

of correlations; and the same pattern have been replicated in our application, i.e. the estimated loadings are around 0.3, with correlations around 0.9. The reasons for these results have escaped explanation, and more careful analysis is required. However, is not hard to realize in this context that, if the effects of the covariates ultimately need to "explain" the responses, and these effects need to pass through the latent variable structure, given the low estimates of the loadings, larger effects on the second-order latent variable would be required. Given the lack of confidence the previous results generates on the magnitude of our covariate estimates, from this point onward, we will interpret our statistical evidence based on the FOLV model.

First, there is sufficient statistical evidence that age explains negatively the levels of reading comprehension; more specifically, the sub-dimensions of reading comprehension (not shown in the plot). The estimate for the age parameter was -0.036 with a 95% compatibility interval of $[-0.043, -0.031]$. This mean that, for each additional unit of age, the applicant departed from the minimum registered in the data (20 years), his/her abilities to find literal information, infer and reflect over a text, get diminished. Although the effect does not seem large on a unit scale, notice that a difference of 10 years among participants would represent nearly 0.7 logits of difference in abilities. As previously stated, this not conclusive evidence that the style of teaching affects the reading comprehension abilities. However, it does provide evidence that older individual inside the teaching career, could benefit from in-training services to improve their reading comprehension abilities.

Second, there is also evidence that disability explained the variability in the reading comprehension sub-dimensions, although the results were mildly unexpected. From the figure we noticed the contrast between having a low vision and no disability was positive (`b_Di[2] - b_Di[1]` $= 0.38$ $[0.051, 0.724]$). This is surprising as one would expect that people with low vision would be in disadvantage. However, a different story could be told. What could had happen is that because people had low vision, they read the exam more carefully, and therefore, were able to obtain better results. The current evidence seem to support the latter. Another unexpected result was that, individuals with low motor skills performed largely worse than individuals with no disability or low vision, as one can observe from the second and third labels of the plot. While the reason for this result is not puzzling, since individual have a disability after all, the actual mechanisms behind it are not entirely clear. Therefore, the educational authority could make a proper investigation of this evidence. Finally, the statistical evidence did not reject the notion that having an auditory disability, does not affect the reading comprehension abilities. Notice the large confidence intervals in the next three plot labels. However, we argue that the impossibility of getting a proper hypothesis test for these contrast, had more to do with the sample design, rather than the true effect being null. As one can intuit, a simple random sample would likely reflect most of the traits of a full data set, however, it is also clear that this design does not necessarily favors the hypothesis testing of contrast, especially, in cases where the groups of interest have low representativeness.

Third, the statistical evidence seem to support the incidental observation, about the quality of training on pedagogical institutes. From the figure we notice that, individual with university degrees had higher levels of reading comprehension abilities, versus individuals with institute degrees (`b_E[2] - b_E[1]` $= 0.260$ $[0.180, 0.340]$). Moreover, we observed that individuals with both degrees were not that different from individuals with institute degrees (`b_E[3] - b_E[1]` $= 0.099$ $[-0.151, 0.355]$). The latter makes sense, if we consider that it is more likely that individuals graduated from institutes, further

develop their education on universities, and not the other way around. However, the implications of this reverberate on two additional hypothesis, the educational authority might be interested on investigate: (i) either the negative offset of starting your pedagogical training on an institute is never recovered, even with the assumed "better" education from universities, or (ii) people from institutes further develop their education on "bad" quality universities. Nevertheless, considering all the previous, we must not forget the evidence presented here only speaks about the reading comprehension abilities, that is, evidence on other abilities need to be assessed. Moreover, we also need to consider the effects might be less or more impressive, once we consider the variability present within institutes.

Fourth, from the last 12 contrast on the top panel of figure 5.8, we see there is statistical evidence that experience improved the reading comprehension abilities. Additionally, we can notice the effects of private experience (`b_Xpr`) were larger than their public counterpart (`b_Xpu`). Finally, on both types of experiences, we can observe a pattern of diminishing returns on abilities, as the years of experience increases.

Fifth and final, the statistical evidence seem to favor the idea that instructing students, which require more complex levels of written and oral communication, might be benefiting the reading comprehension abilities of teachers. However, the hypothesis is favored only for teacher involved in the secondary educational level (notice the contrast `b_S[3] - b_S[1]` and `b_S[3] - b_S[2]`).

Chapter 6

Conclusions and discussion

As stated in the introductory section, the current research described and implemented the Bayesian GLLAMM model for dichotomous outcomes [56, 58, 65, 59], in the context of an educational data. The reason for the model's proposal revolved around the fact that educational data often presents multiple types of dependencies, that left unchecked, can cause IRT models to violate their assumptions of local independence. The latter is particularly important, as violation of these assumption prevent IRT models to reach appropriate inferences from the parameter estimates [73, 9, 32].

Moreover, in the context of the previously defined model, the current research also provided an assessment of the benefits resulting from changing the posterior sampling geometries. Multiple evidence pointed out the performance improvement on the MCMC methods from using non-centered parameterizations [18, 19, 51, 52, 6]. However, most of the evidence have been developed under Gaussian hierarchical models. So, it seemed sensible to provide a similar assessment for nonlinear latent stochastic models, like our implementation [52].

Finally, the research applied the newfound knowledge to a large standardized teacher assessments from Peru. The purpose of the latter was to evaluate the change of parametrization on a real data setting, determine the evidence in favor of our models of interest, produce psychometric analysis, and finally assess specific research hypothesis.

Therefore the main conclusion derived from our work were the followings:

1. In the context of the implemented model, the non-centered parametrization largely improved the performance of the MCMC chains, towards achieving ergodicity. This was true across models, simulated sample sizes, simulated replicas, and even under the real application, albeit with some caveats.

   The most important caveat, under the simulation and application setting, was that no matter the parametrization, no large difference in performance was observed in either the sub-dimensions' correlation or loading parameters. On this matter, however, no evidence supported the idea the parameters suffered from a further lack of identification.

2. Our proposed model was able to recover most of the simulated parameters with good precision.

   Similar to the previous result, the model still had issues estimating the sub-dimensions correlations and loadings. This result is important, as under the Confirmatory Fac-

tor Analysis theory (CFA), a SOLV model is only justified, if the lower-level correlations are high enough (usually above 0.8). Moreover, according to the same theory, once the SOLV model is fitted, assuming the model is correct, it is expected the correlation of the lower-level latent variables to be largely reduced, something that was not observed in our simulation studies, not even in accordance to our simulation parameters.

3. The proposed models managed to produce a rather well depiction of the true simulated ICC and IIF curves.

   The previous implied the models allow us to correctly recover the item's psychometric characteristics, a trait of high relevance for the development of evaluation instruments.

4. In terms of retrodiction accuracy, the models managed to capture the traits of the data, while avoiding its exact replication. These result were consistent across models, simulated sample sizes and replicas, and even under the final educational application.

   Consequently, it is safe to say the models produced similar encodings of the data, leaving the decision of choosing one model over the other, on a more theoretical ground.

5. The non-centered parametrization was slightly faster than the centered counterpart, although the magnitudes of the differences in running time were not large.

   This result is still important, as the non-centered parametrization was more complex, and required the sampling of more parameters than the centered counterpart. This mean that improving the performance of the MCMC, through a more complex model as the NCP, did not come with a cost on running time.

6. On the application side, the model provided an extra benefit, that is, we were able to asses the psychometric properties of texts, rather than just items.

7. Finally, in relation to our hypothesis of interest, the model was able to produce sound statistical results, supported not only by the statistical application, but also from a DAG guiding our interpretation and causal assumptions.

6.1 Future developments

From the simulation studies and application, we noticed the benefits of the non-centered parametrization did not extend to the estimation of correlation parameters or loadings. In this regard, it would be of interest to investigate why we observe this pattern, considering that the hypothesis of lack of identification have been discarded.

On the other hand, in similar sections, we have been surprised that although the chains produced by the centered parametrization, did not show signs of achieving ergodicity, its recovery capacity was at par of its non-centered counterpart. The considered hypothesis were related to the use of the HMC algorithm with a higher rejection criteria (`adapt_delta`= 0.99) and weakly regularizing priors. However, it would be interesting to test the validity of these hypothesis, using the full factorial design outlined in chapter 4,

and even adding a new factor to the mix, that is, a comparison between HMC and Gibbs sampling.

Related to the previous two statements, evidence on similar chapters have revealed that, some parameters within a model were sampled with better performance with the CP, in contrast to the NCP, while in the remaining parameters happened the opposite. This resonates with the statement of Papaspiliopoulos et al. [52], that the advantages of the NCP strategy largely depends on the specifics of the model and data, and that the CP and NCP are complements of each other, rather than replacements. Therefore, under this scenario, it seems sensible to investigate the benefits of Variational Inference methods (VI) to estimate the posterior distribution of the parameters. As the reader can recall, VI seek to produce a sample mechanism located in a continuous between a CP and NCP.

On the other hand, in the hypothesis test section of the application chapter, we realized that a simple random sample was not the appropriate sample design to evaluate all our contrasts of interest. This was specially true in variables with levels that had low representativeness, like disability. Furthermore, we realized that the full benefit of a DAG, comes from using it not only to structure our statistical model, but also to design the collection of data. In this sense, it would be enlightening to see if the hypothesis results are replicated with a more appropriate sample design.

Furthermore, in our application section, we assumed the relationship between the reading comprehension abilities and age was linear. However, nothing prevents that non-linearities are present in this relationship. In a similar spirit, it would be interesting to apply the GLLAMM model, in a setting were covariates are not only at the structural level, but also at the level of the responses, e.g. assessment in online environments.

Finally, remains as an interesting path for future research, the application of the GLLAMM to a multi-group setting, where an analysis of invariance is required.

Appendix A

Figures and tables

A.1 Chapter 3: Bayesian estimation

A.1.1 To center or not to center

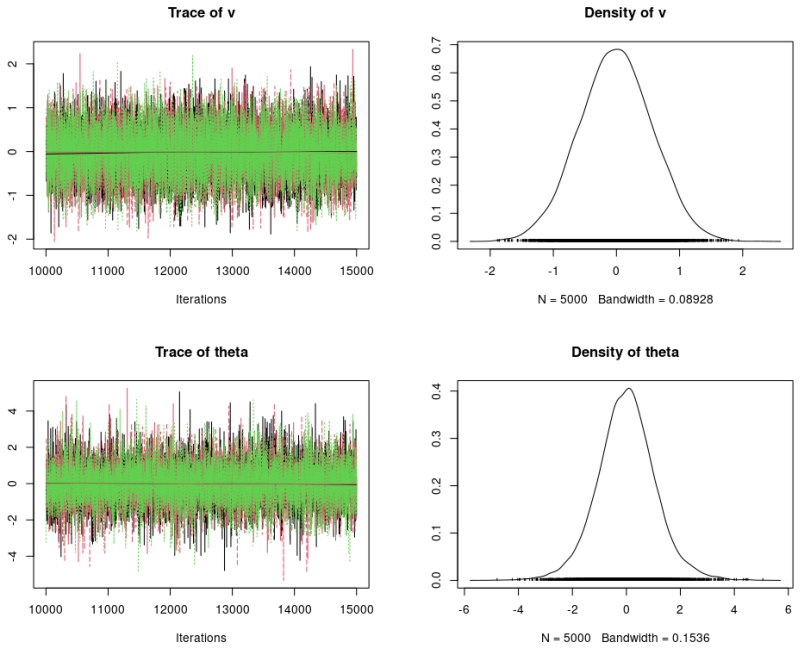

Figure A.1: The Devil's funnel. Centered Parametrization implemented in JAGS. It shows the traceplot and distribution of the parameters of interest.

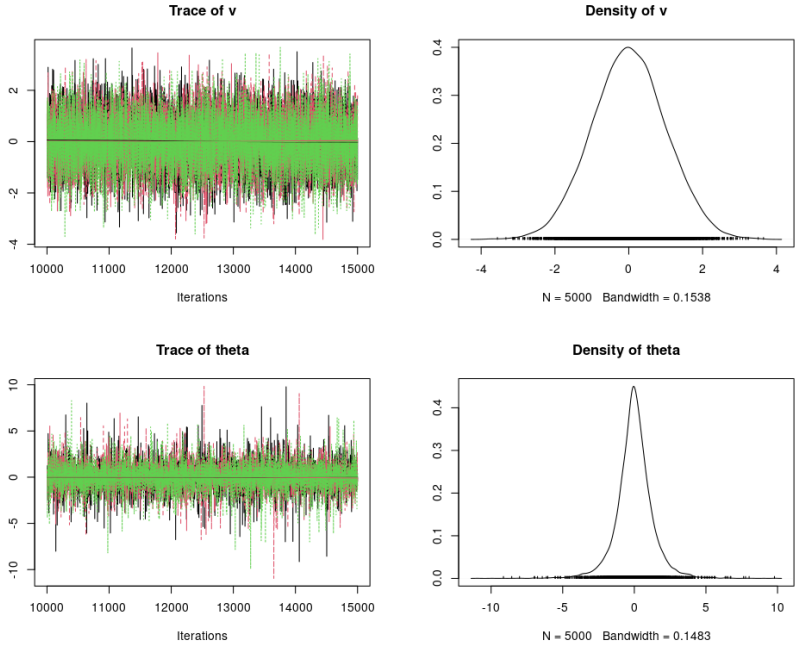

Figure A.2: The Devil's funnel. Centered Parametrization with mildly informative priors implemented in JAGS. It shows the traceplot and distribution of the parameters of interest.

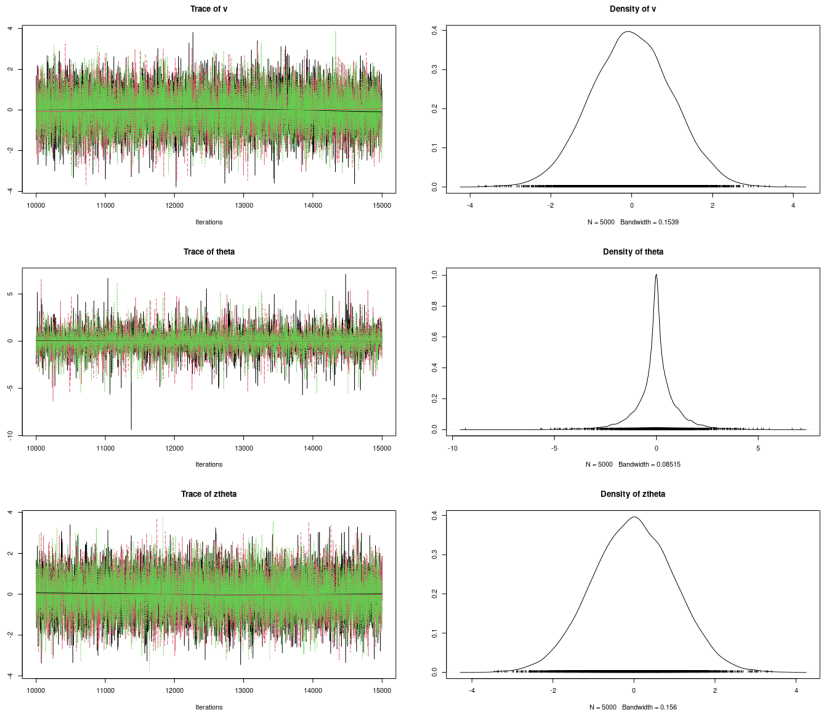

Figure A.3: The Devil's funnel. Non-Centered Parametrization implemented in JAGS. It shows the traceplot and distribution of the parameters of interest.

A.2 Chapter 4: Simulation study

A.2.1 Prior elicitation

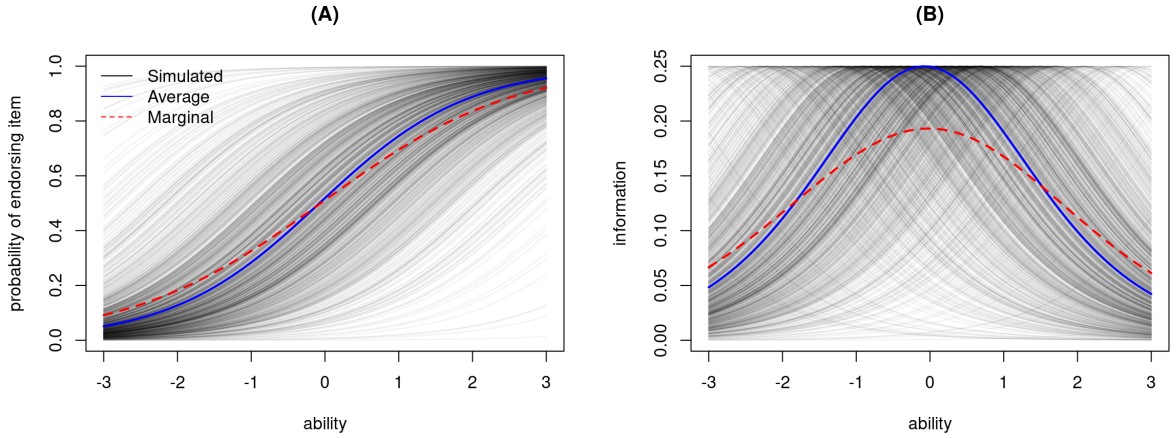

Figure A.4: Second-order latent variable model (SOLV). (A) Item Characteristics Curve, ICC. (B) Item Information Function, IIF.

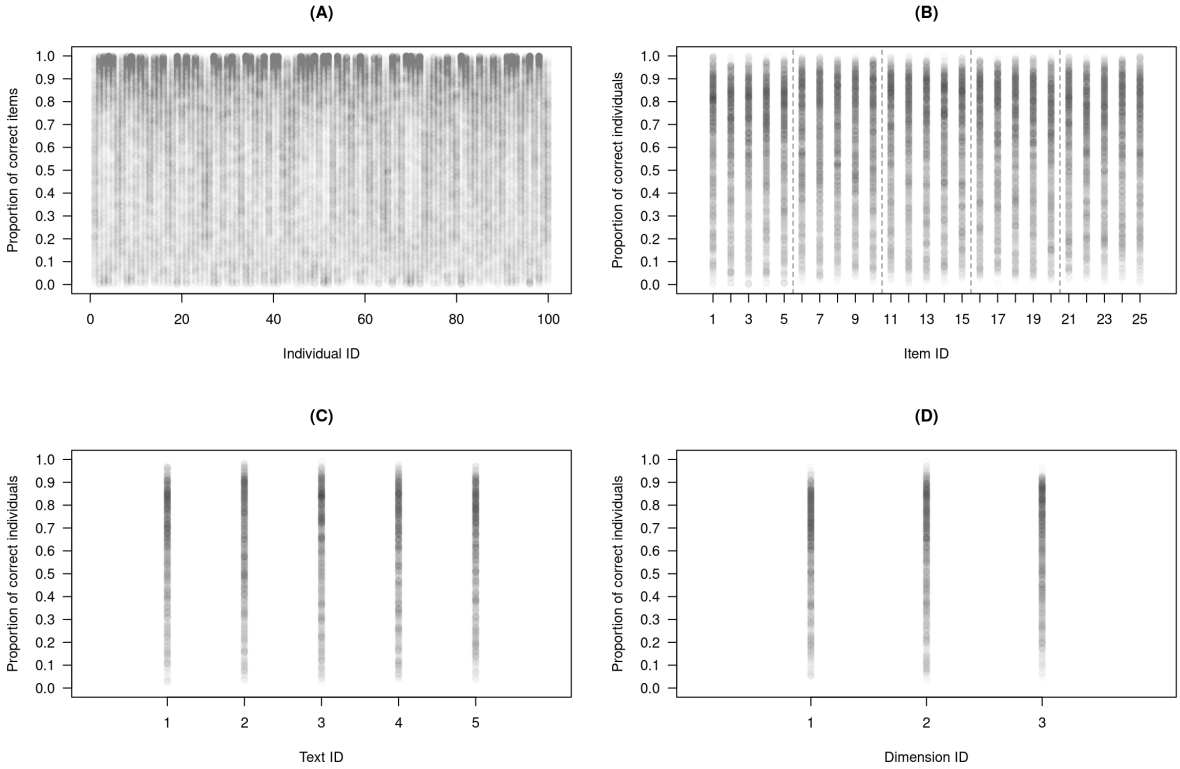

Figure A.5: Second-order latent variable model (SOLV). Aggregated endorsement rate per: (A) individuals, (B) items, (C) text or passage, and (D) measured dimension.

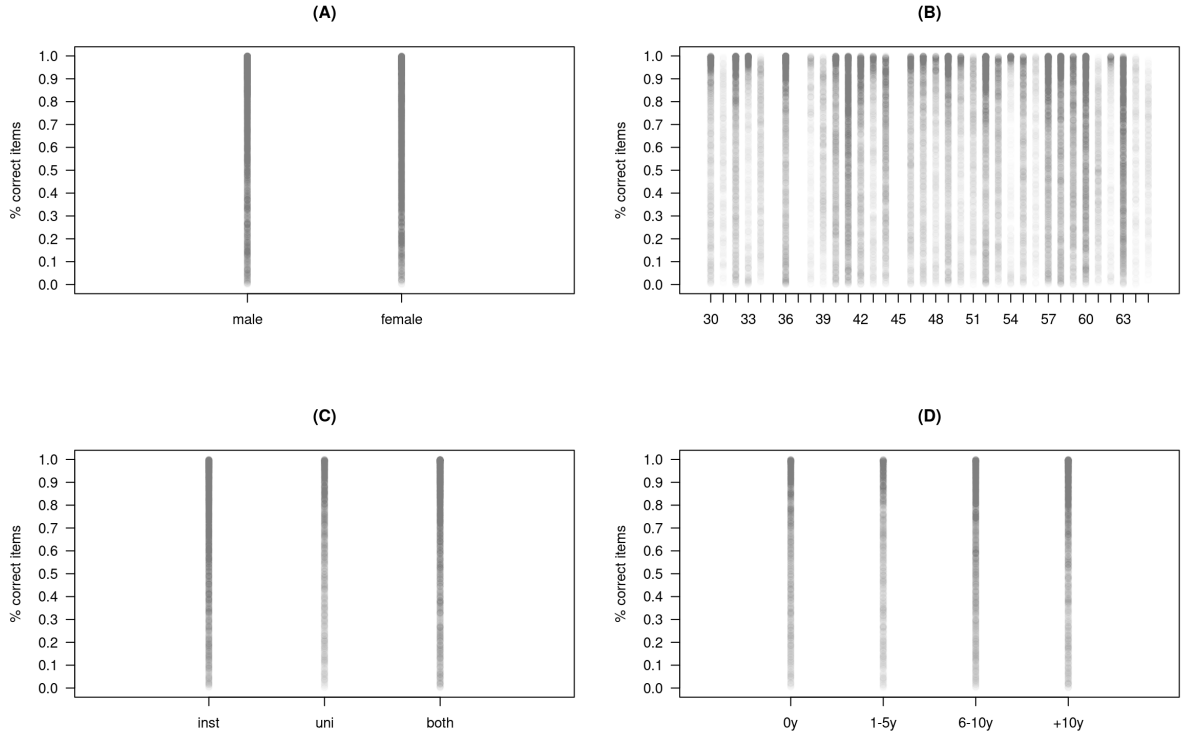

Figure A.6: Second-order latent variable model (SOLV). Aggregated endorsement rate per simulated covariate: (A) gender, (B) age, (C) education, and (D) experience.

A.2.2 Chain performance

This section shows only a small set of trace, trank and ACF plots for the parameters of interest. For all the plots across parameters, models, parametrizations, and replicas refer to the "chains" image section of the accompanying github page:
`https://github.com/jriveraespejo/thesis/tree/master/images/chains`

Similarly, the CP and NCP `n_eff` and `Rhat` comparison plots shown here is a small set of the full available plots. For the set of figures across parameters, models, and replicas refer to the "chains/stat" image section of the accompanying github page:
`https://github.com/jriveraespejo/thesis/tree/master/images/chains_stat`

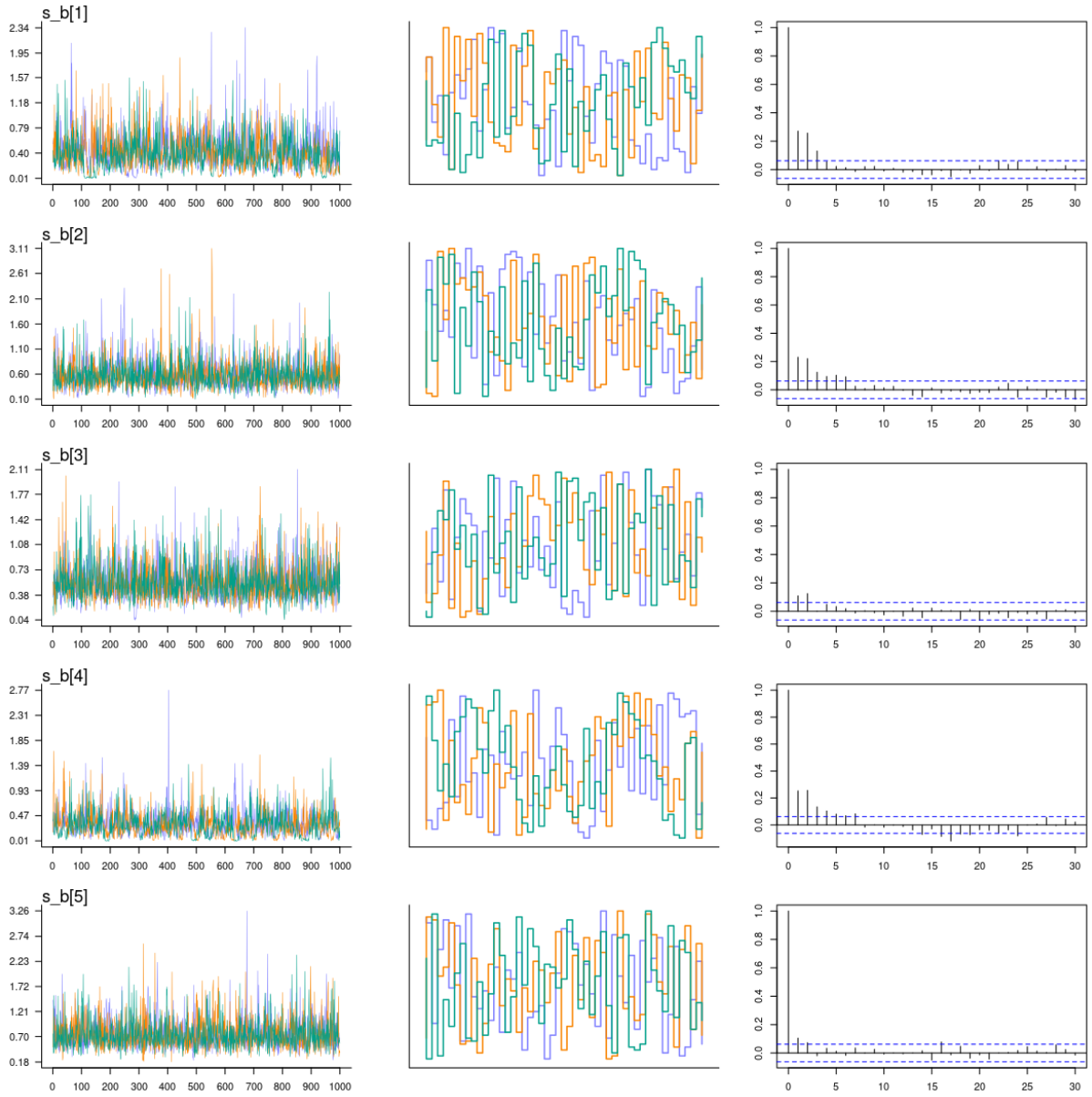

Figure A.7: First-order latent variable model (FOLV). Sample size 100, replica number 3. Centered parametrization. Difficulty deviation per text: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

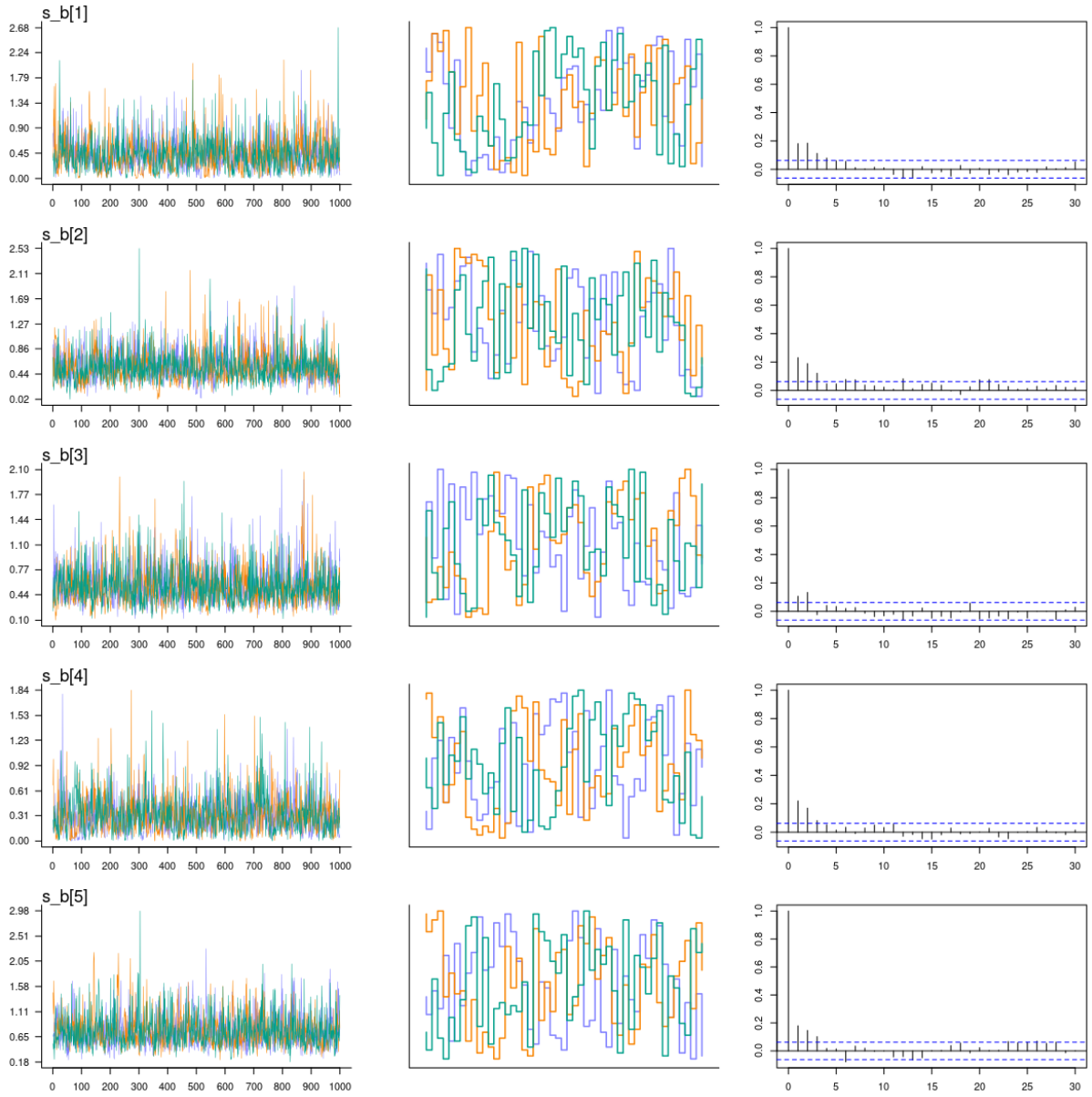

Figure A.8: First-order latent variable model (FOLV). Sample size 100, replica number 3. Non-centered parametrization. Difficulty deviation per text: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

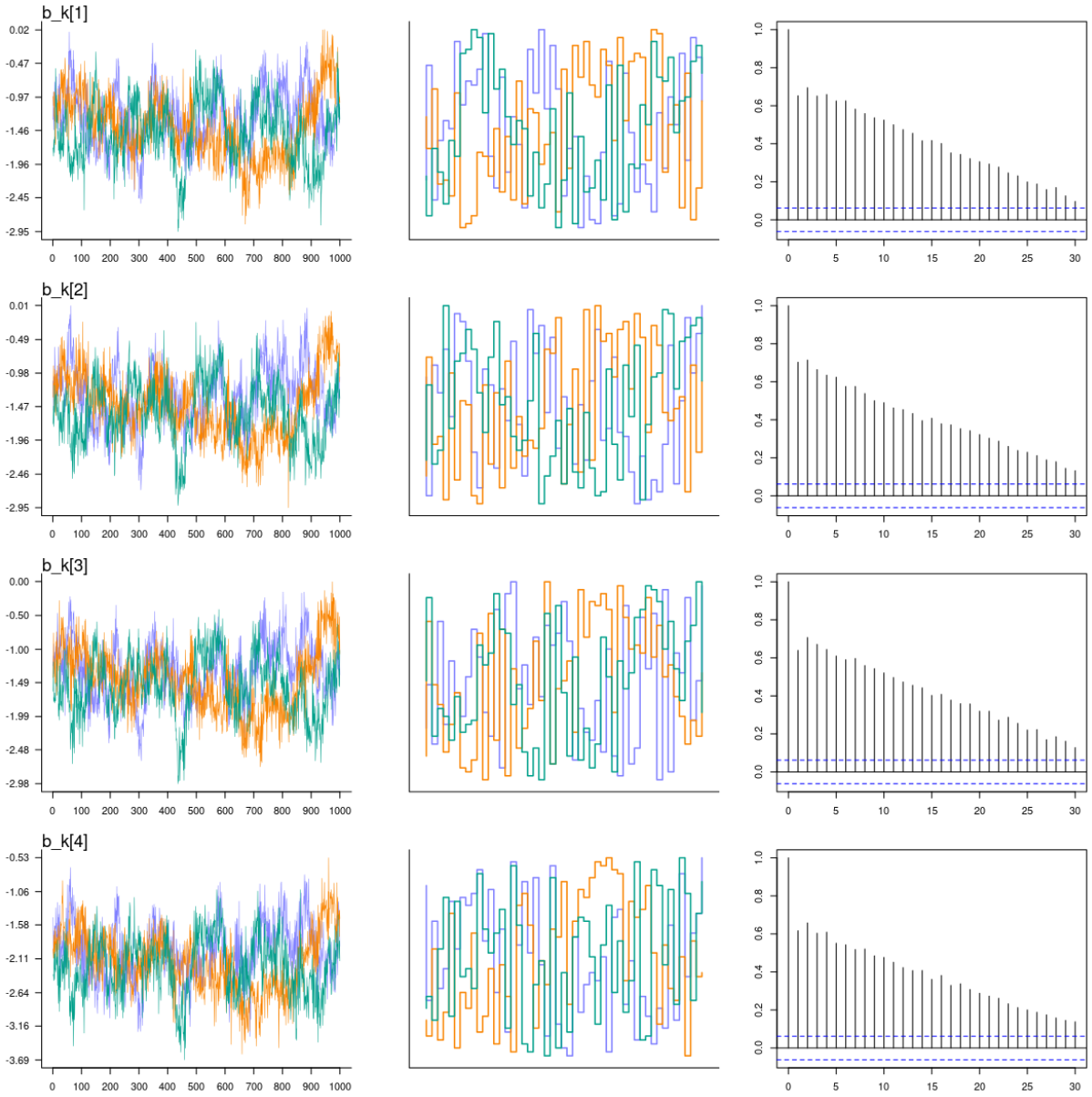

Figure A.9: First-order latent variable model (FOLV). Sample size 100, replica number 1. Centered parametrization. Difficulty per item: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

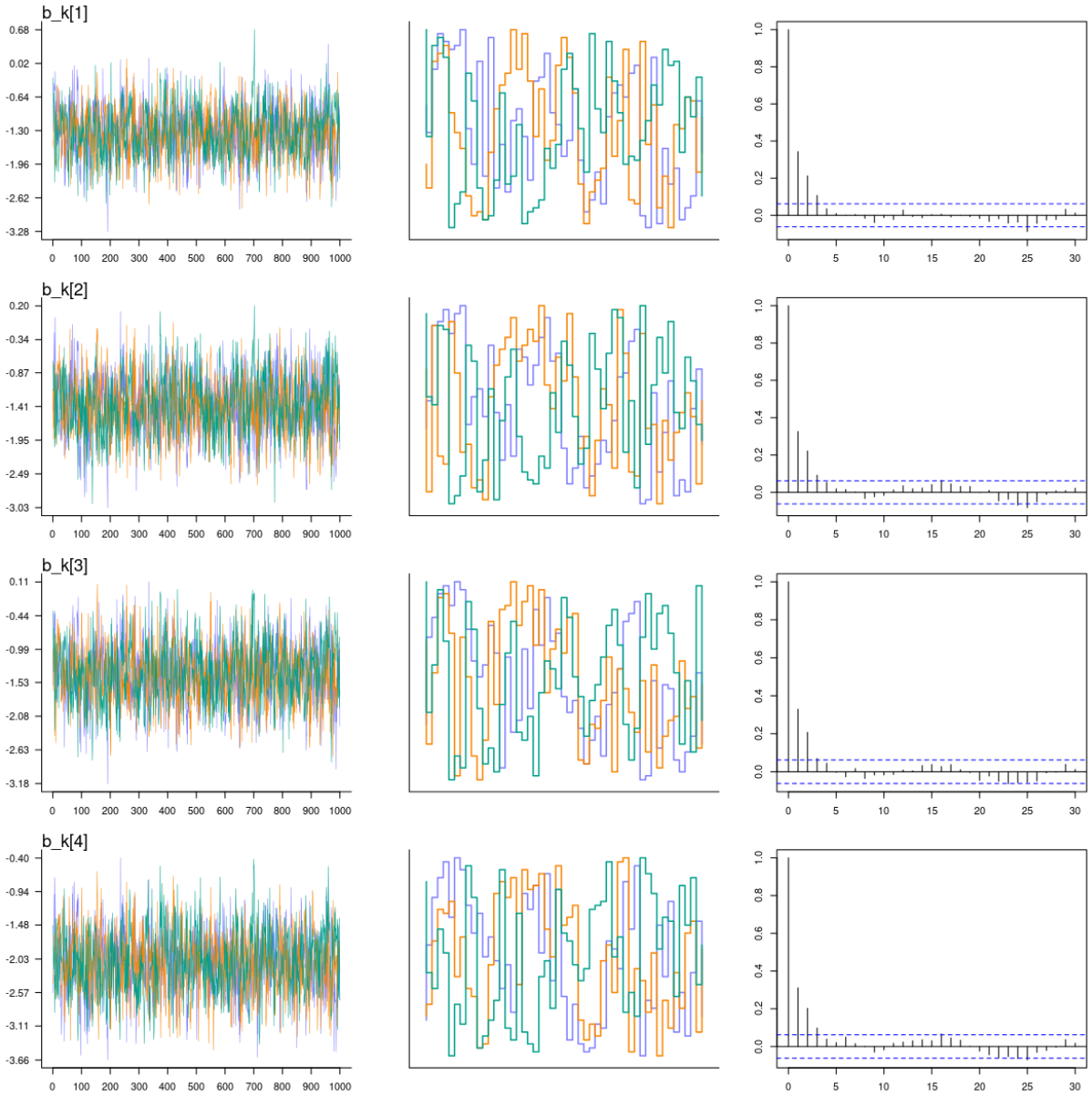

Figure A.10: First-order latent variable model (FOLV). Sample size 100, replica number 1. Non-centered parametrization. Difficulty per item: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

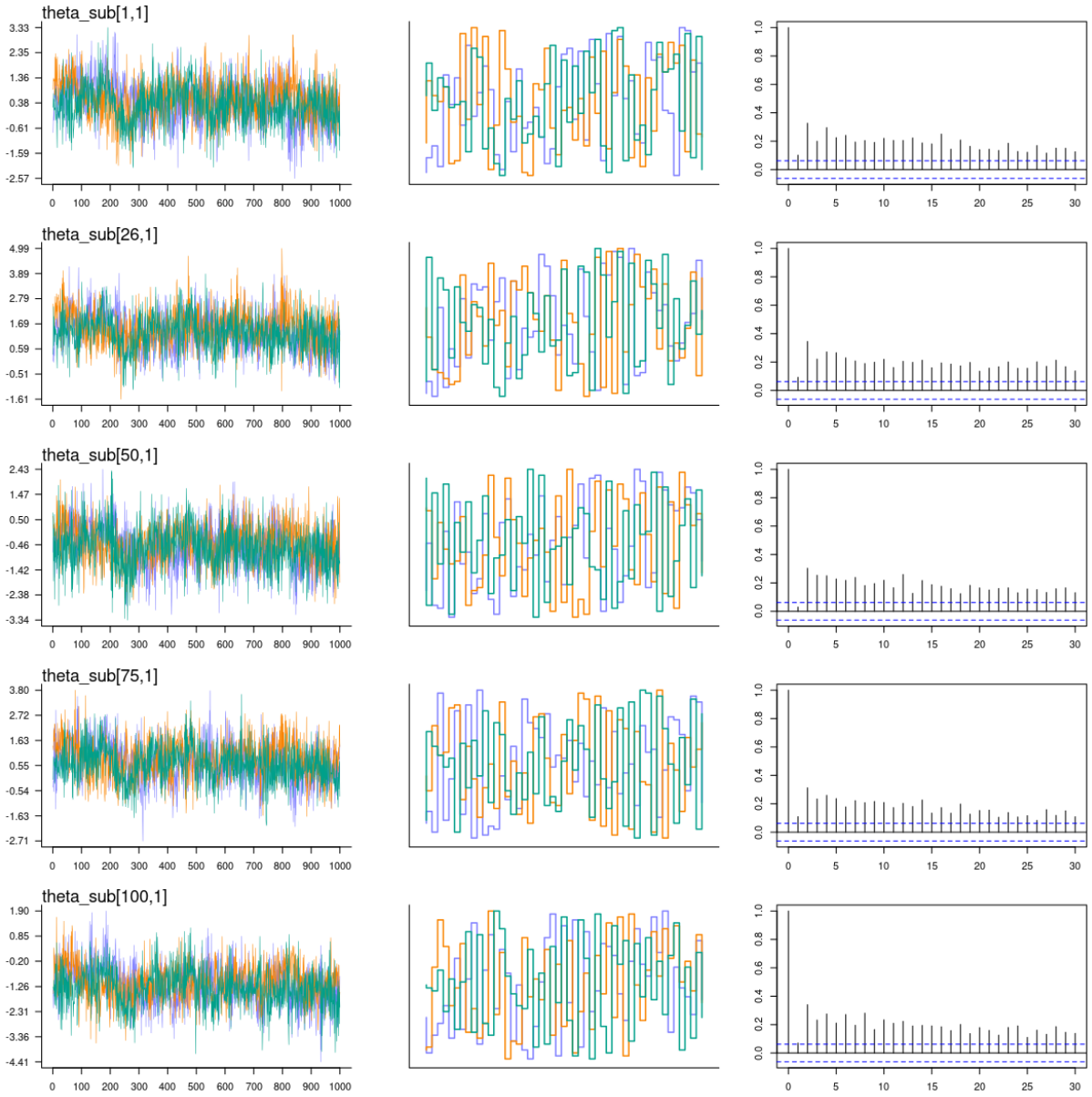

Figure A.11: First-order latent variable model (FOLV). Sample size 100, replica number 6. Centered parametrization. Individual's first sub-dimension: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

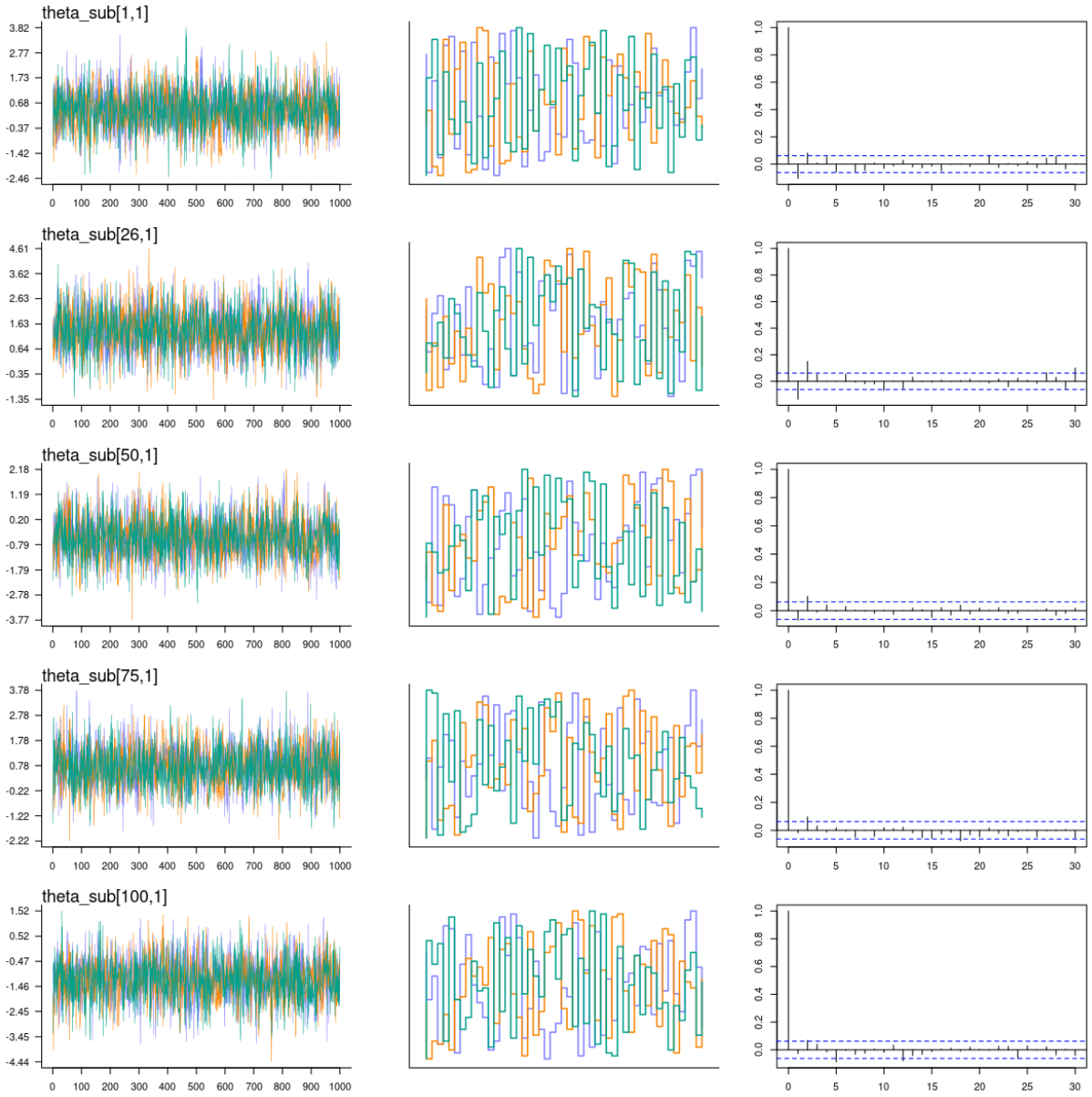

Figure A.12: First-order latent variable model (FOLV). Sample size 100, replica number 6. Non-centered parametrization. Individual's first sub-dimension: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

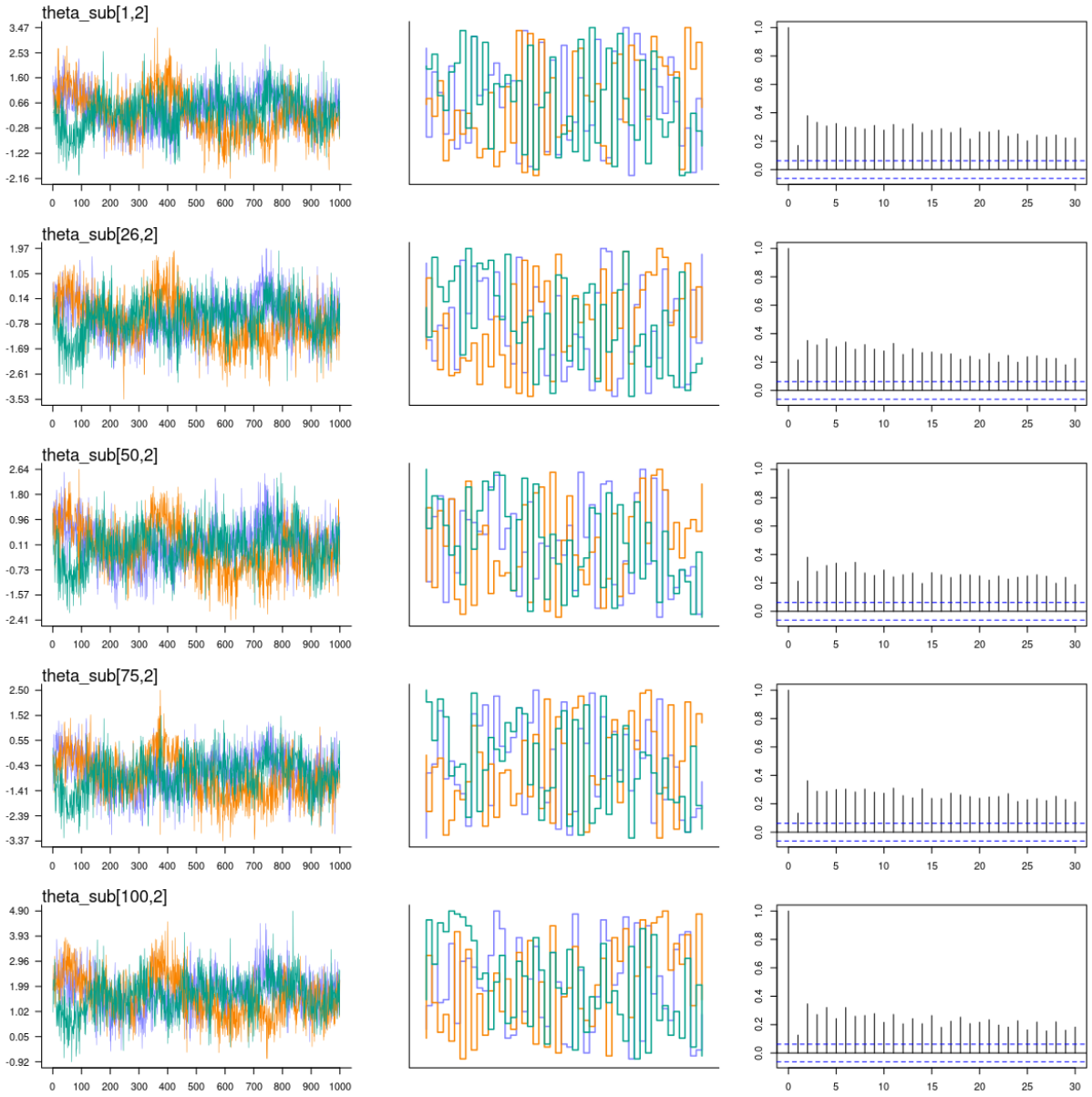

Figure A.13: First-order latent variable model (FOLV). Sample size 100, replica number 7. Centered parametrization. Individual's second sub-dimension: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

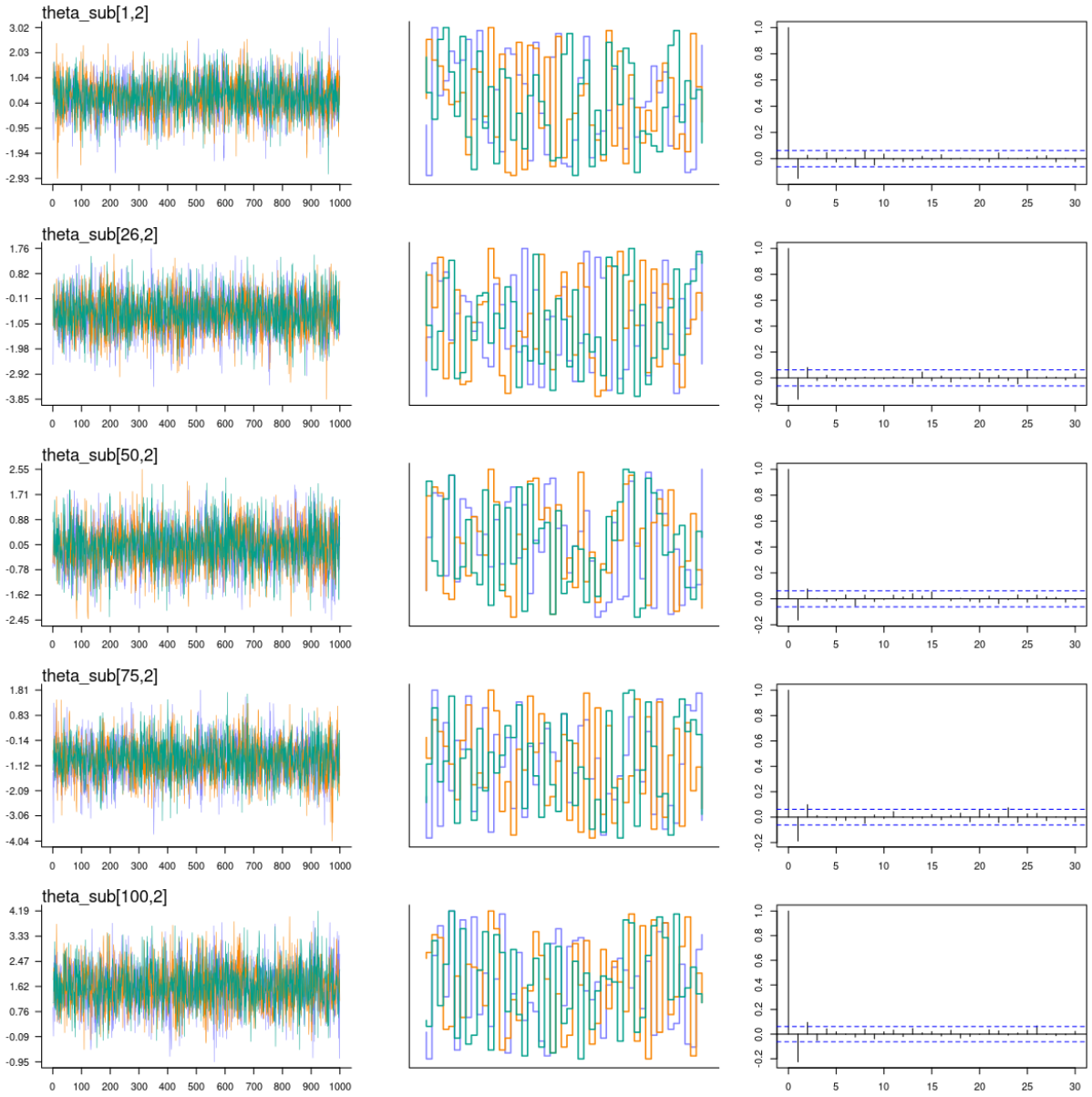

Figure A.14: First-order latent variable model (FOLV). Sample size 100, replica number 7. Non-centered parametrization. Individual's second sub-dimension: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

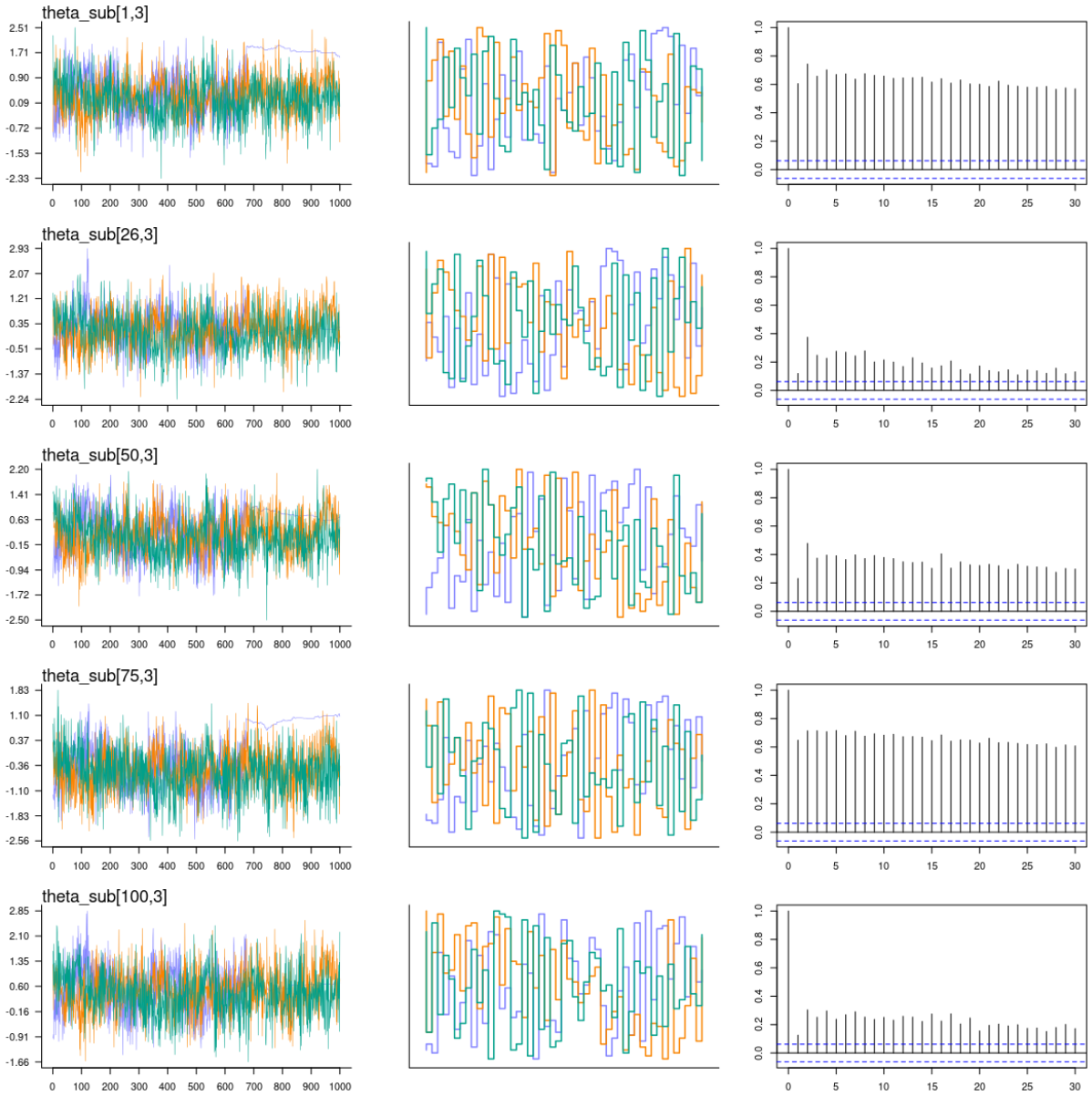

Figure A.15: First-order latent variable model (FOLV). Sample size 100, replica number 8. Centered parametrization. Individual's third sub-dimension: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

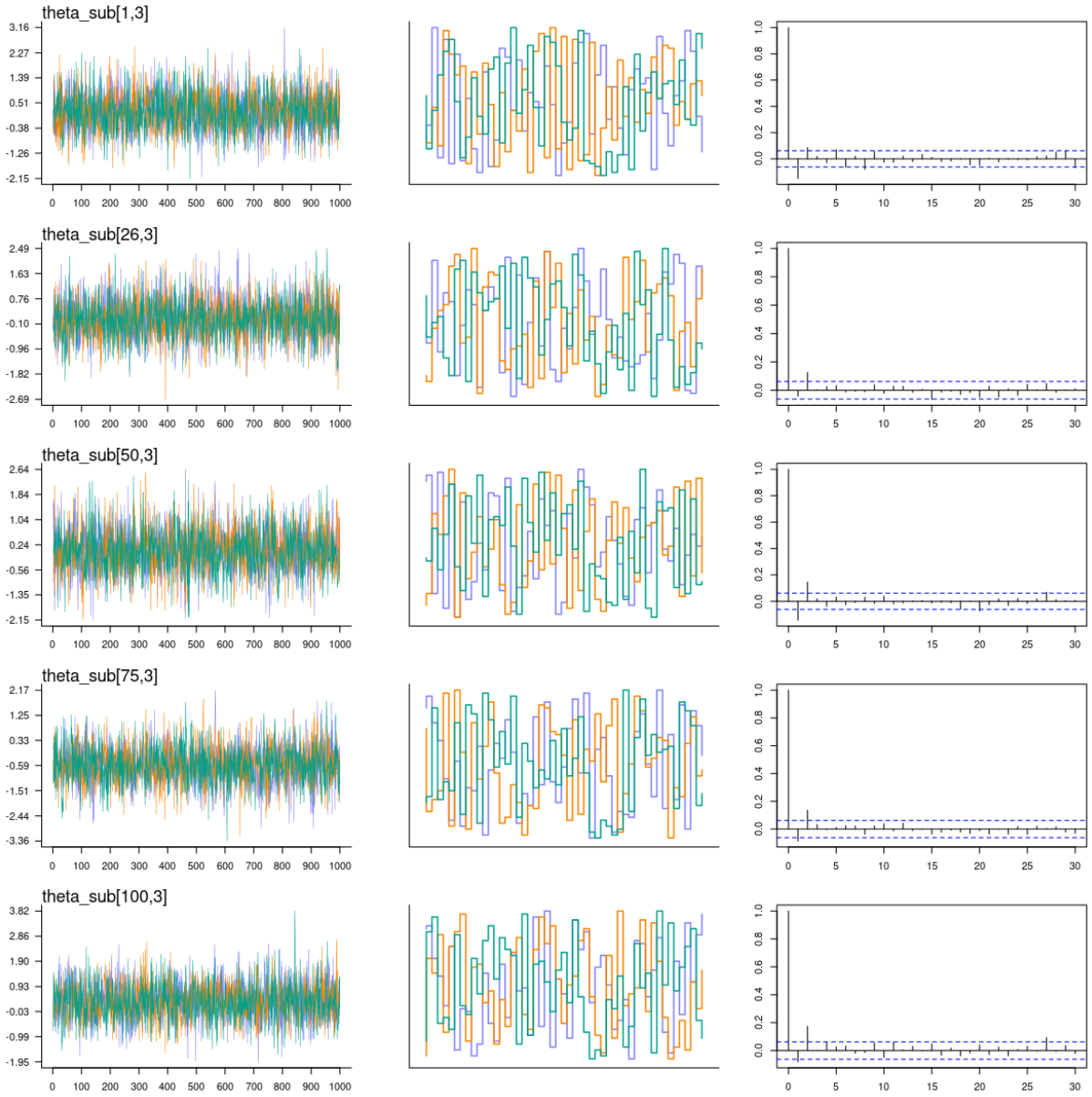

Figure A.16: First-order latent variable model (FOLV). Sample size 100, replica number 8. Non-centered parametrization. Individual's third sub-dimension: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

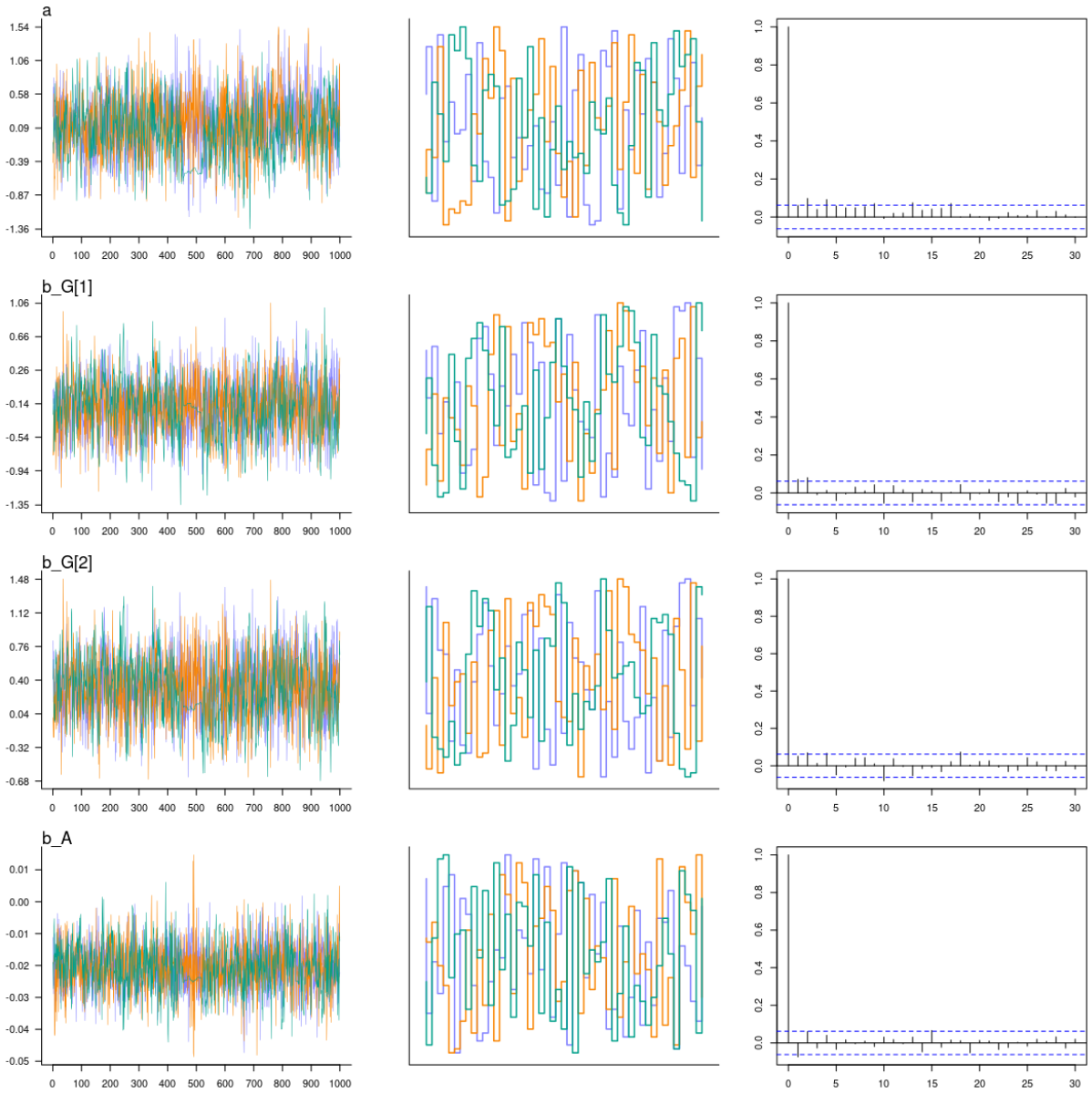

Figure A.17: First-order latent variable model (FOLV). Sample size 100, replica number 4. Centered parametrization. Regression parameters: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

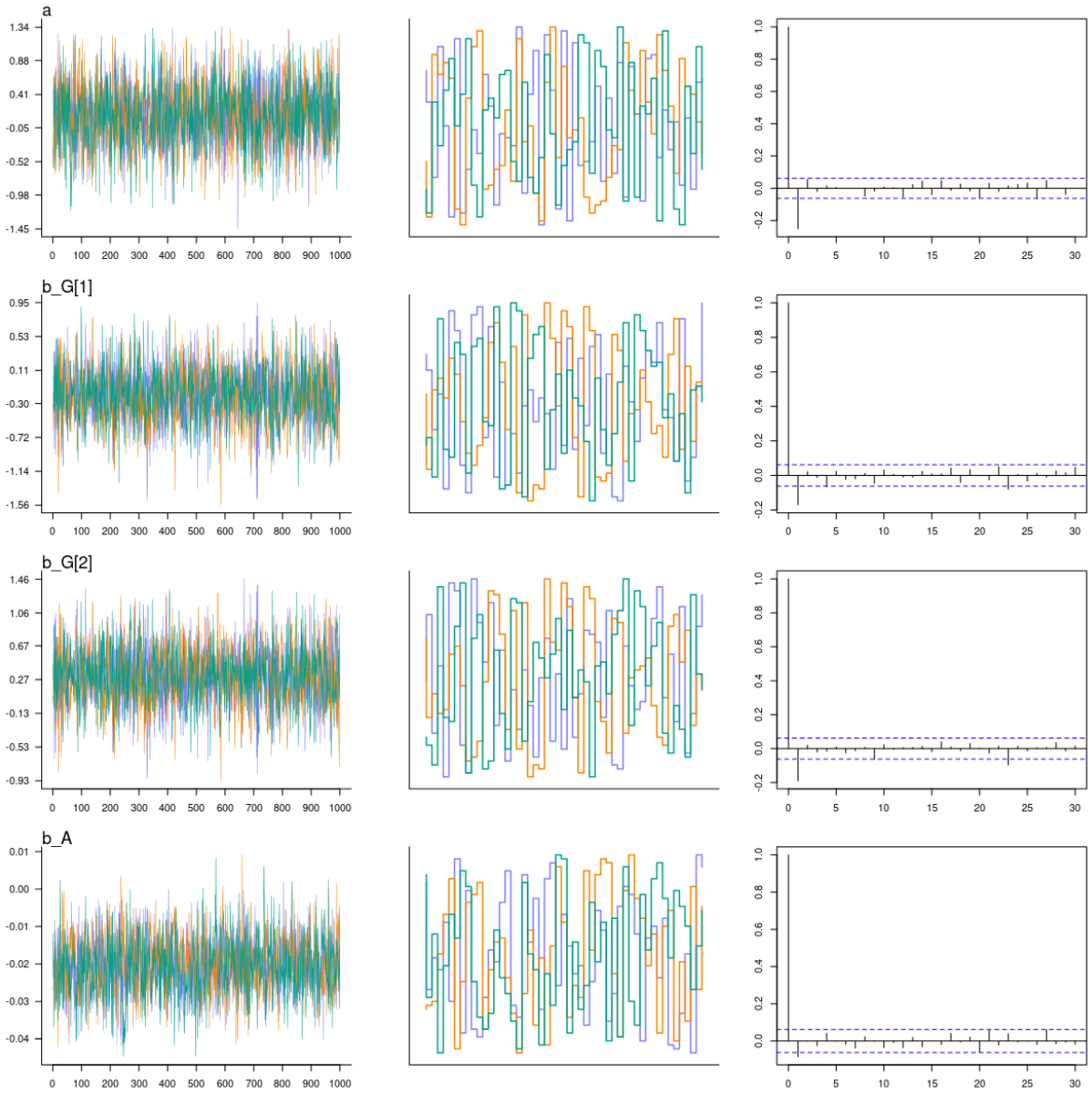

Figure A.18: First-order latent variable model (FOLV). Sample size 100, replica number 4. Non-centered parametrization. Regression parameters: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

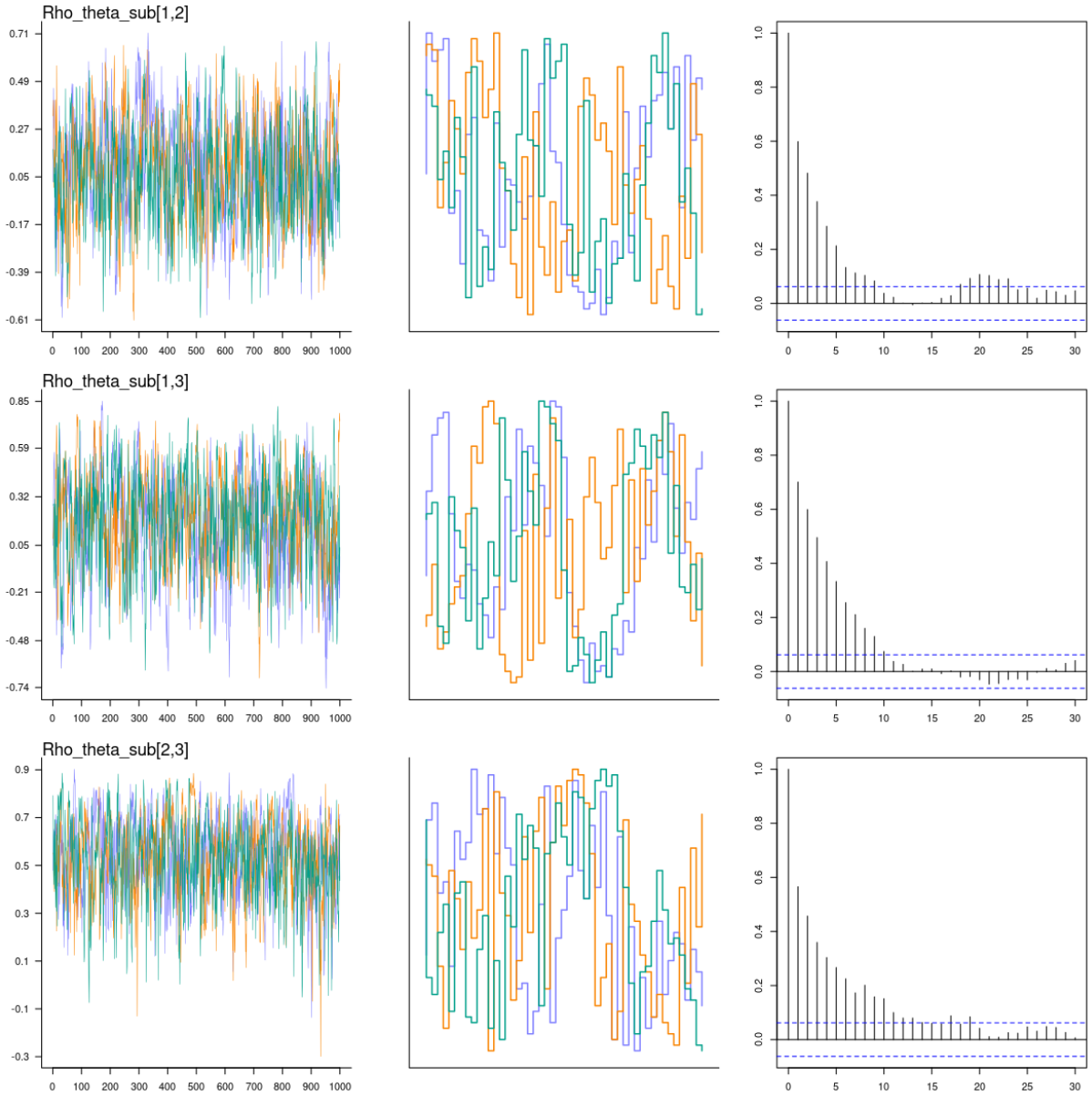

Figure A.19: First-order latent variable model (FOLV). Sample size 100, replica number 5. Centered parametrization. Correlation of sub-dimensions: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

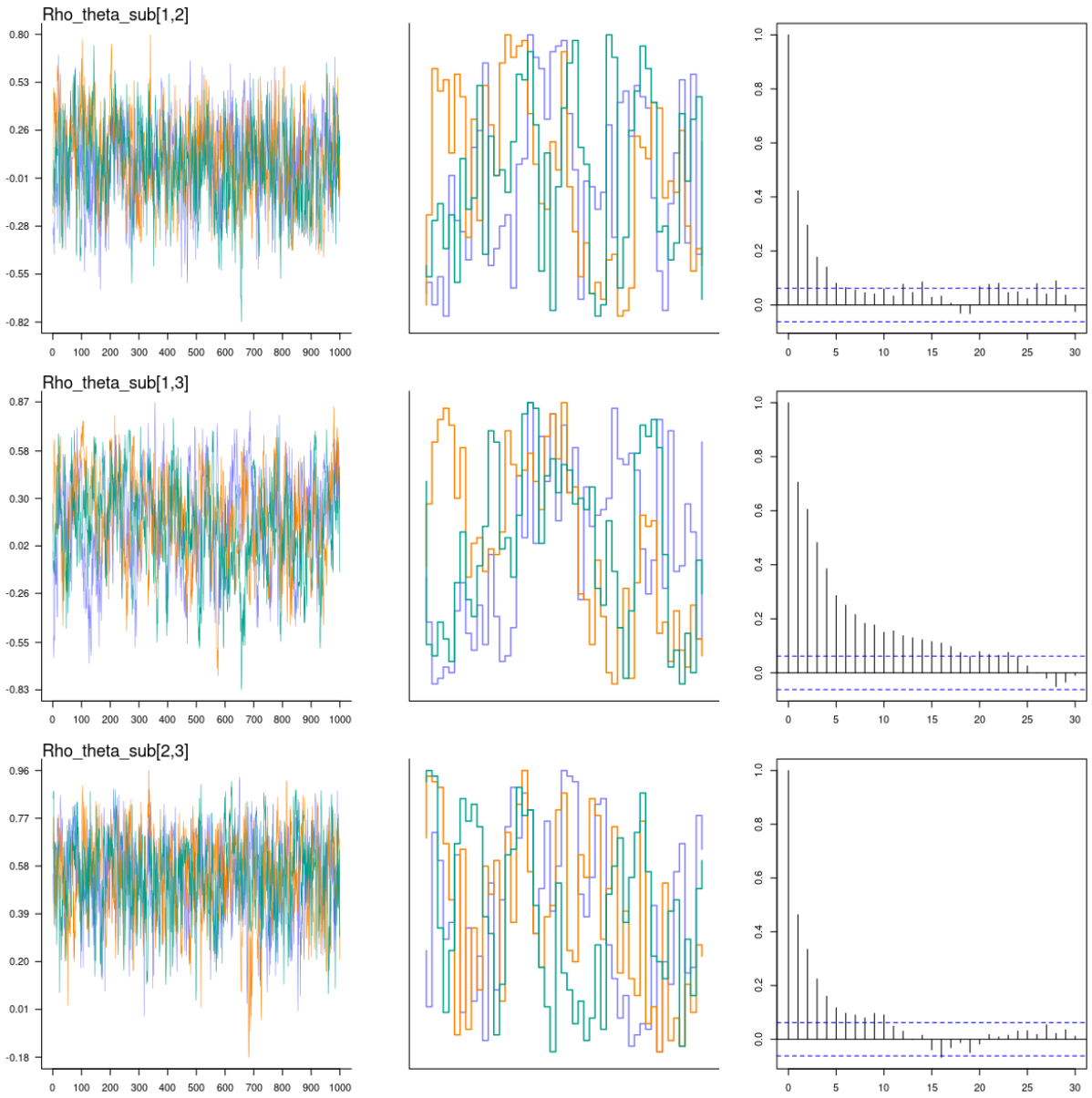

Figure A.20: First-order latent variable model (FOLV). Sample size 100, replica number 5. Non-centered parametrization. Correlation of sub-dimensions: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

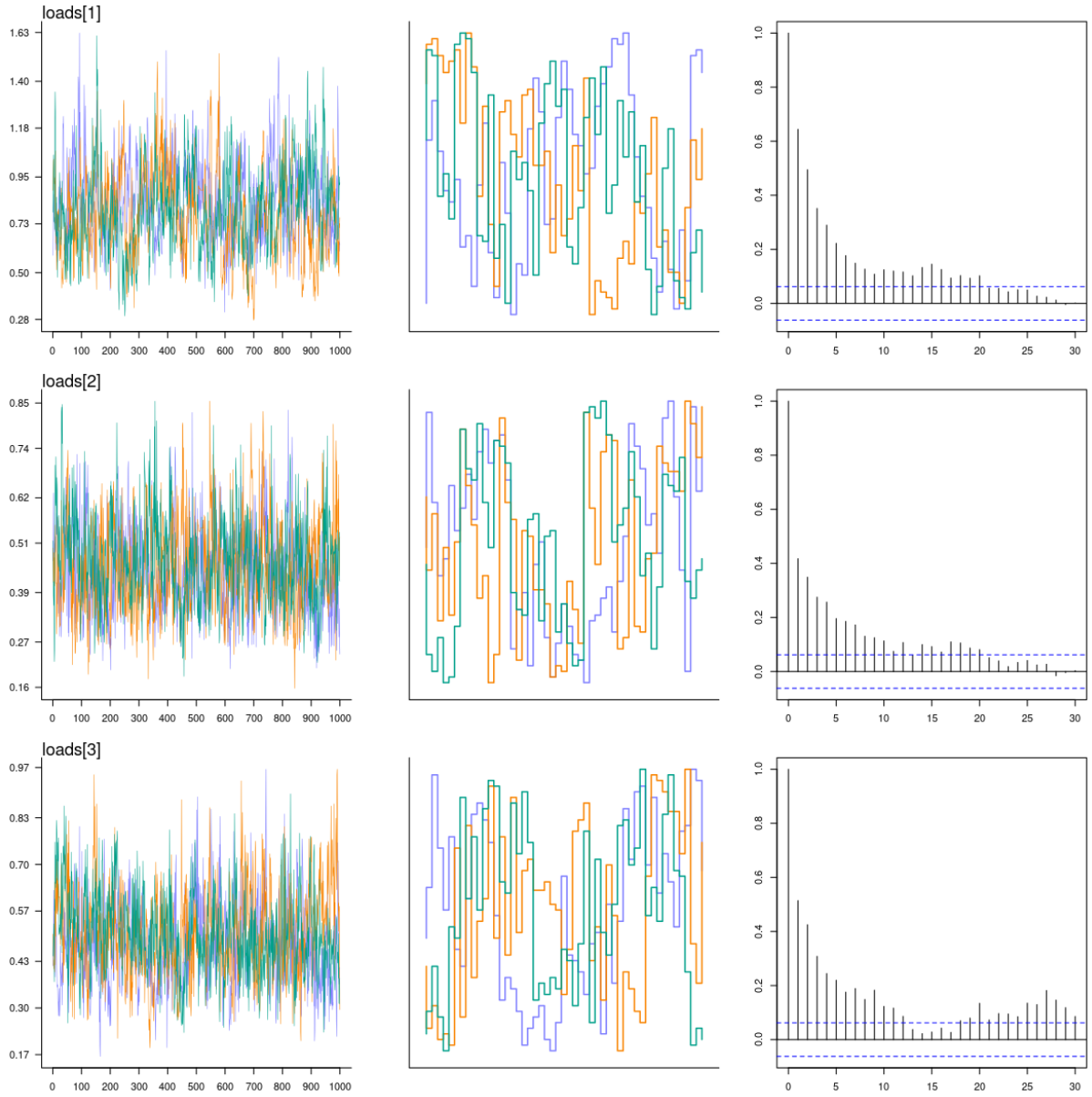

Figure A.21: Second-order latent variable model (SOLV). Sample size 100, replica number 9. Centered parametrization. Loadings: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

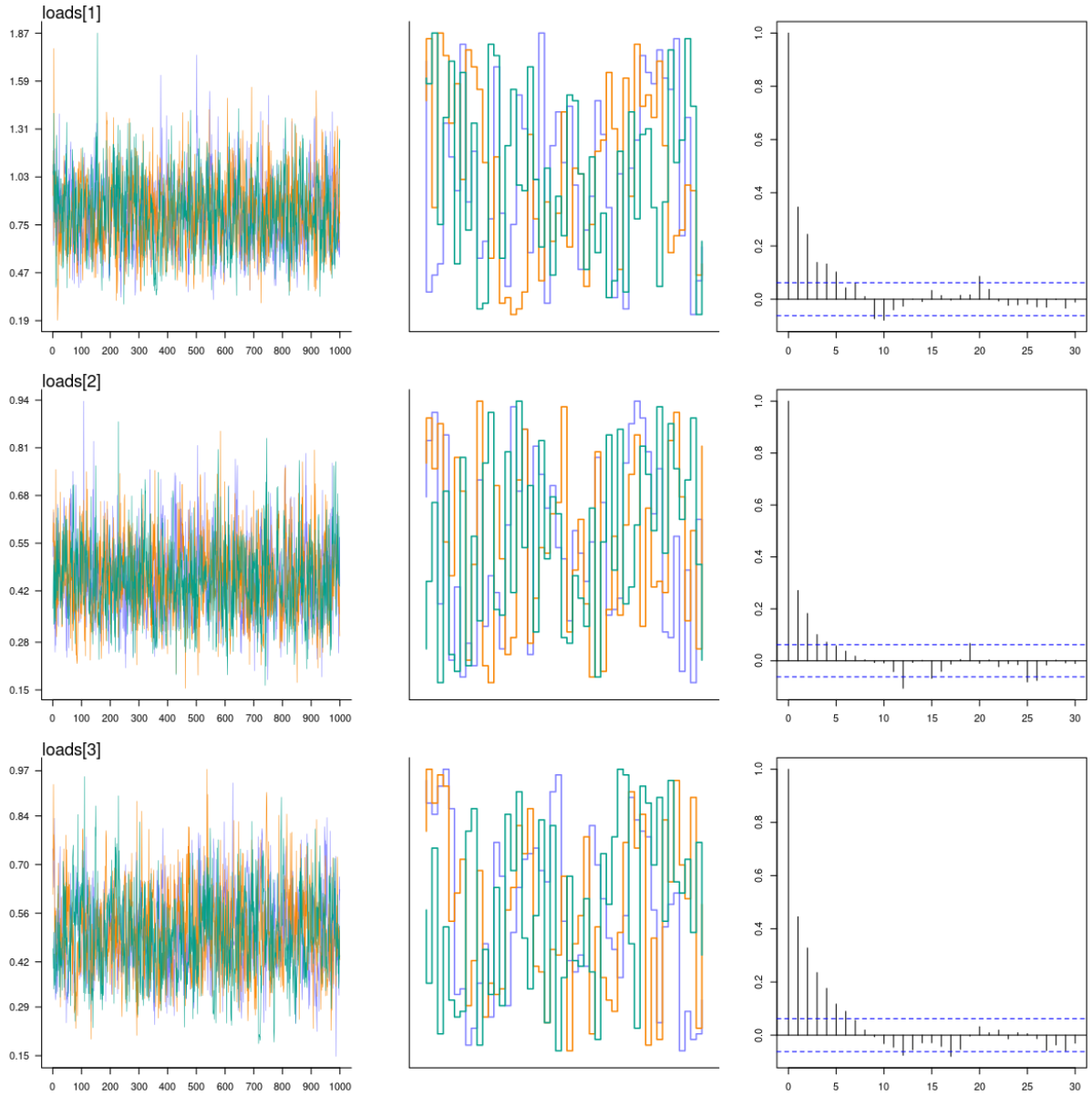

Figure A.22: Second-order latent variable model (SOLV). Sample size 100, replica number 9. Non-centered parametrization. Loadings: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

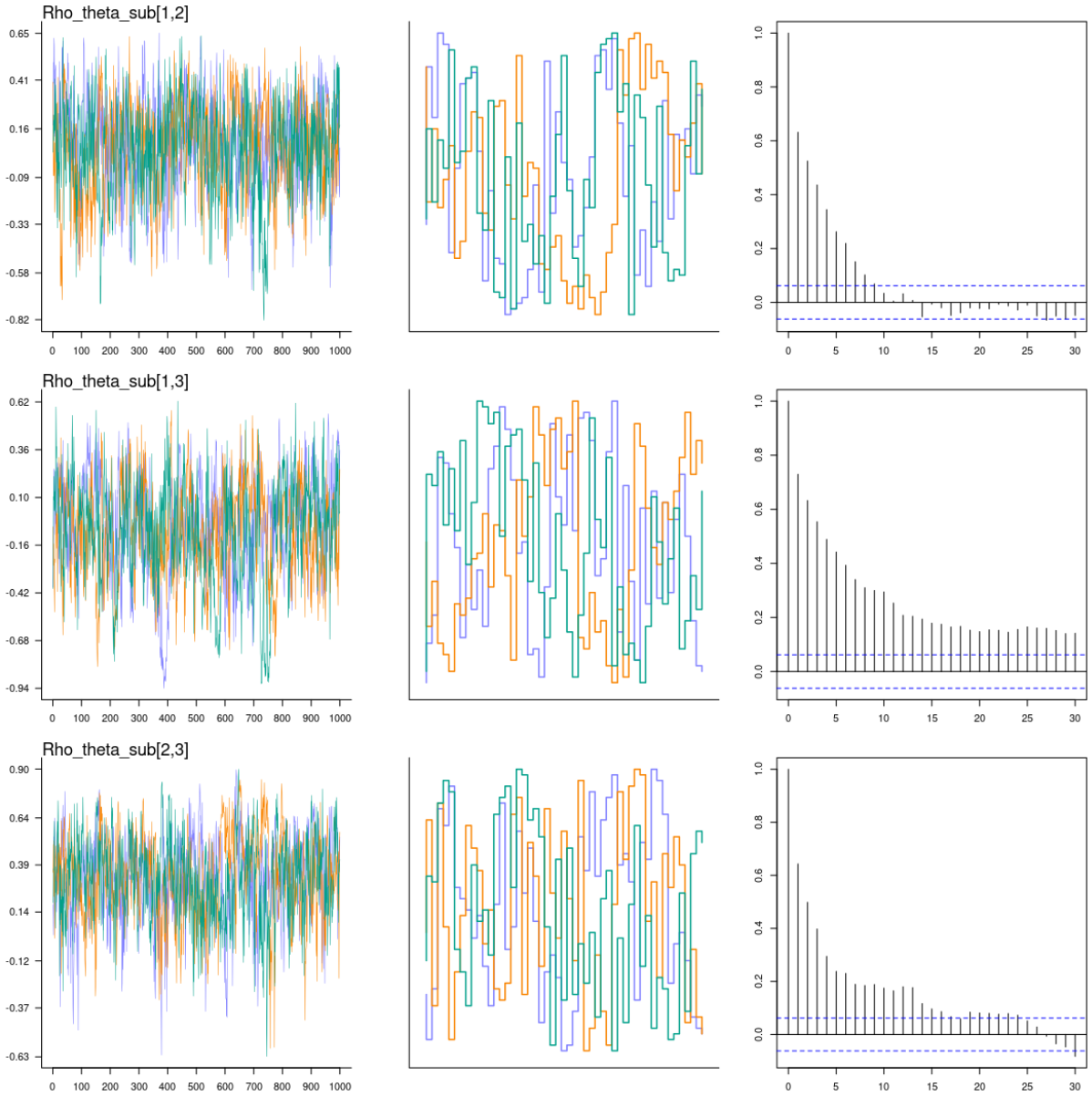

Figure A.23: Second-order latent variable model (SOLV). Sample size 100, replica number 1. Centered parametrization. Correlation of sub-dimensions: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

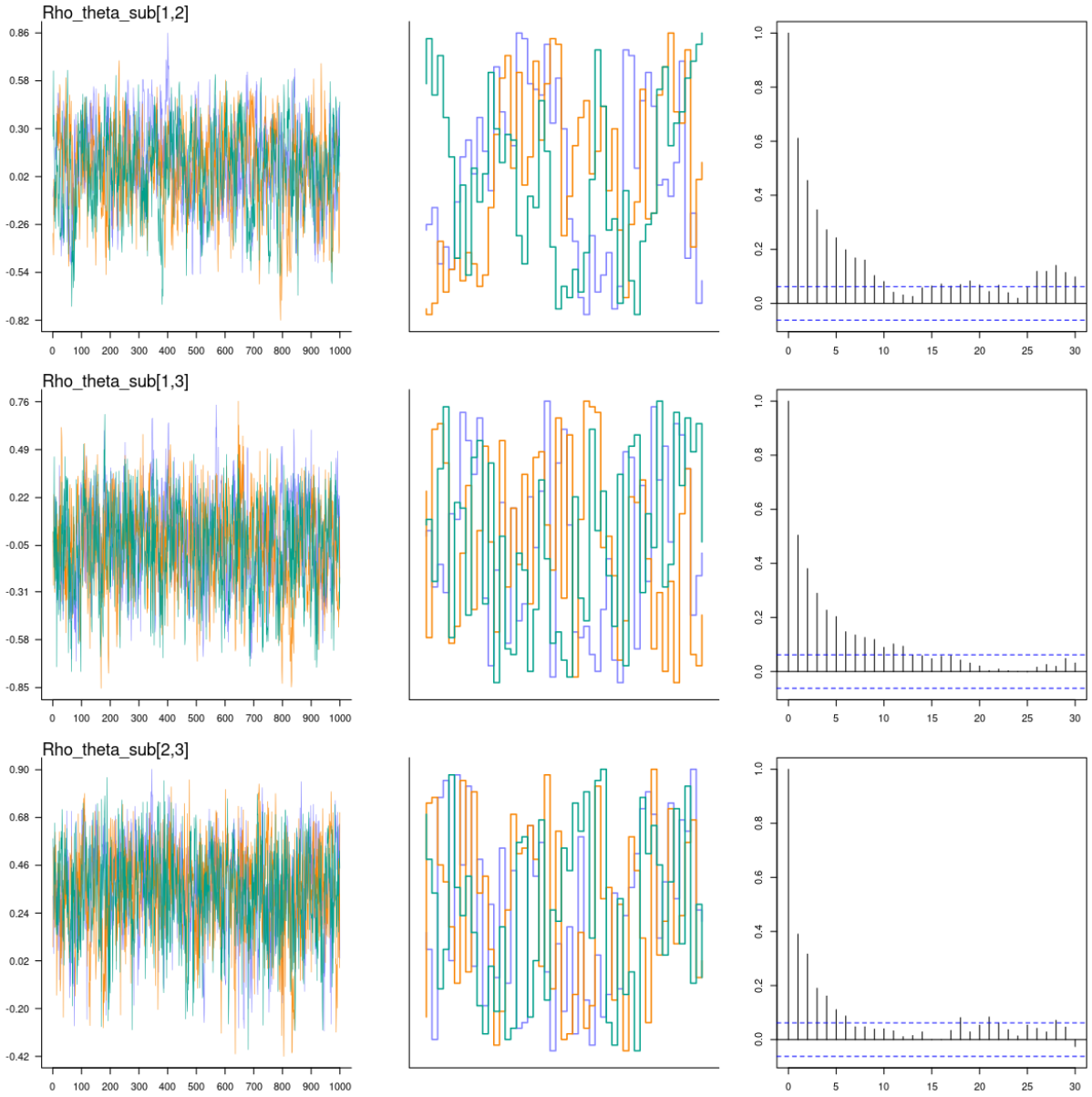

Figure A.24: Second-order latent variable model (SOLV). Sample size 100, replica number 1. Non-centered parametrization. Correlation of sub-dimensions: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

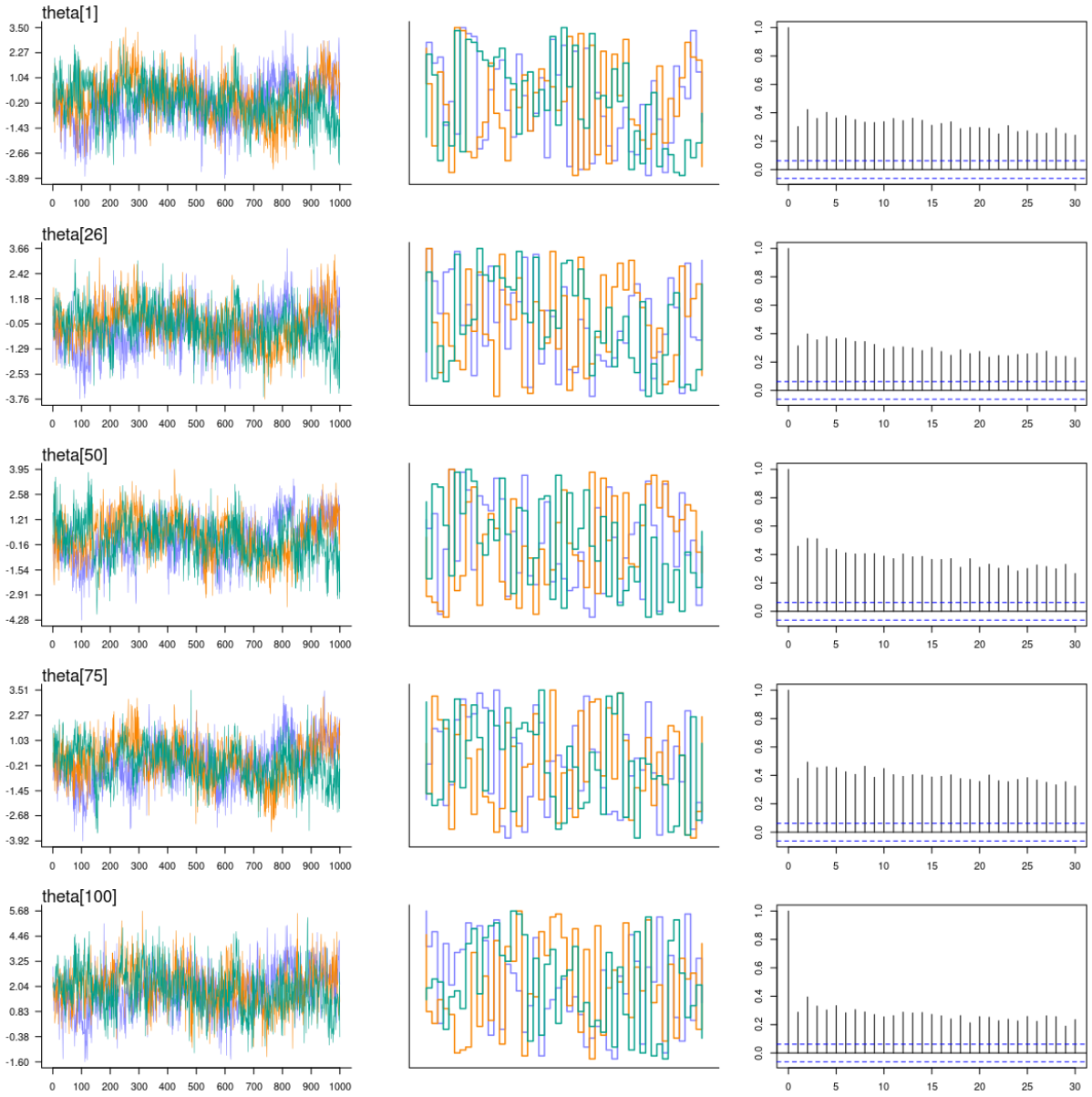

Figure A.25: Second-order latent variable model (SOLV). Sample size 100, replica number 10. Centered parametrization. Highest-order dimension: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

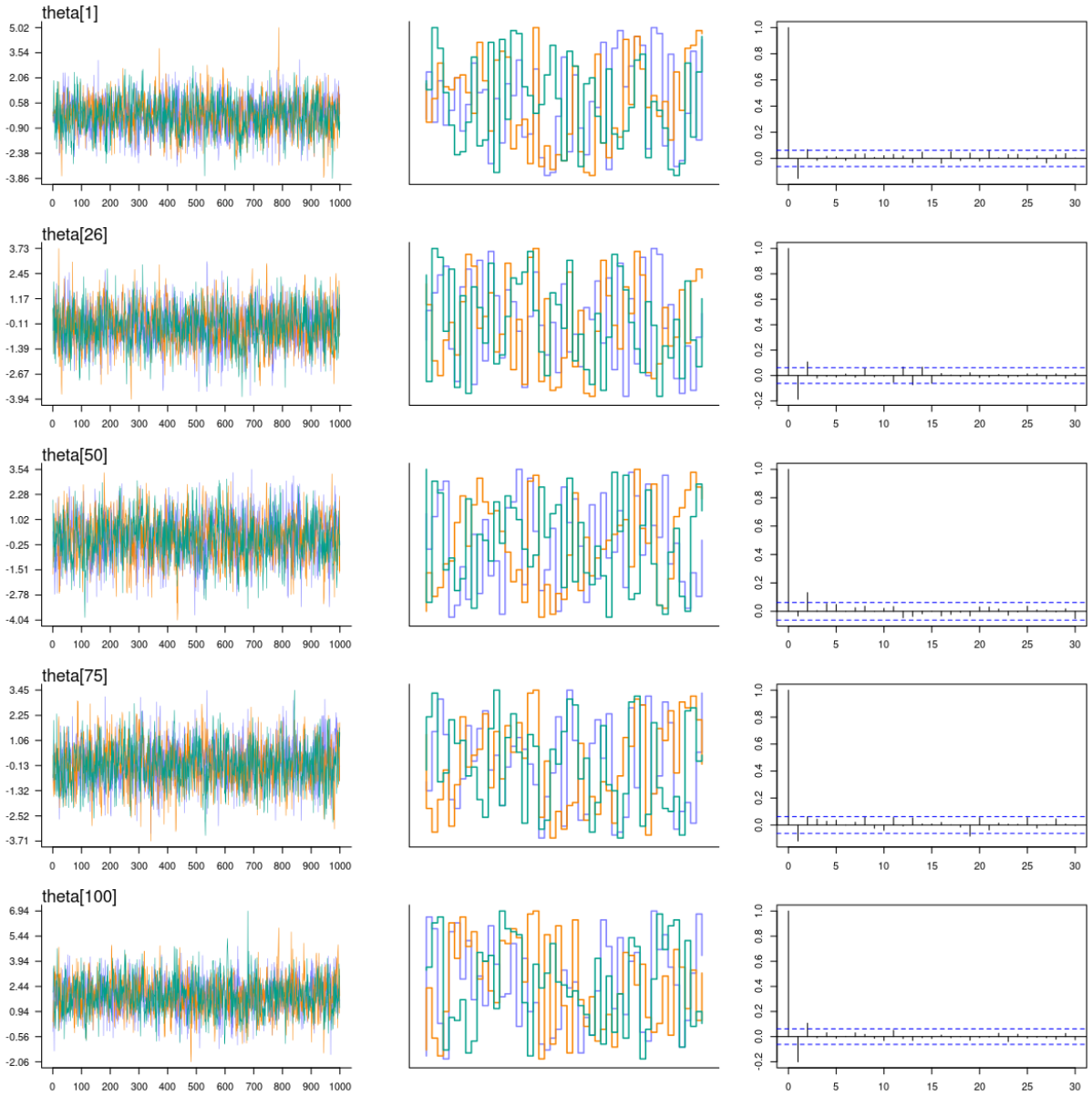

Figure A.26: Second-order latent variable model (SOLV). Sample size 100, replica number 10. Non-centered parametrization. Highest-order dimension: (Left) trace plot, (Middle) trank plot, (Right) auto-correlation plot.

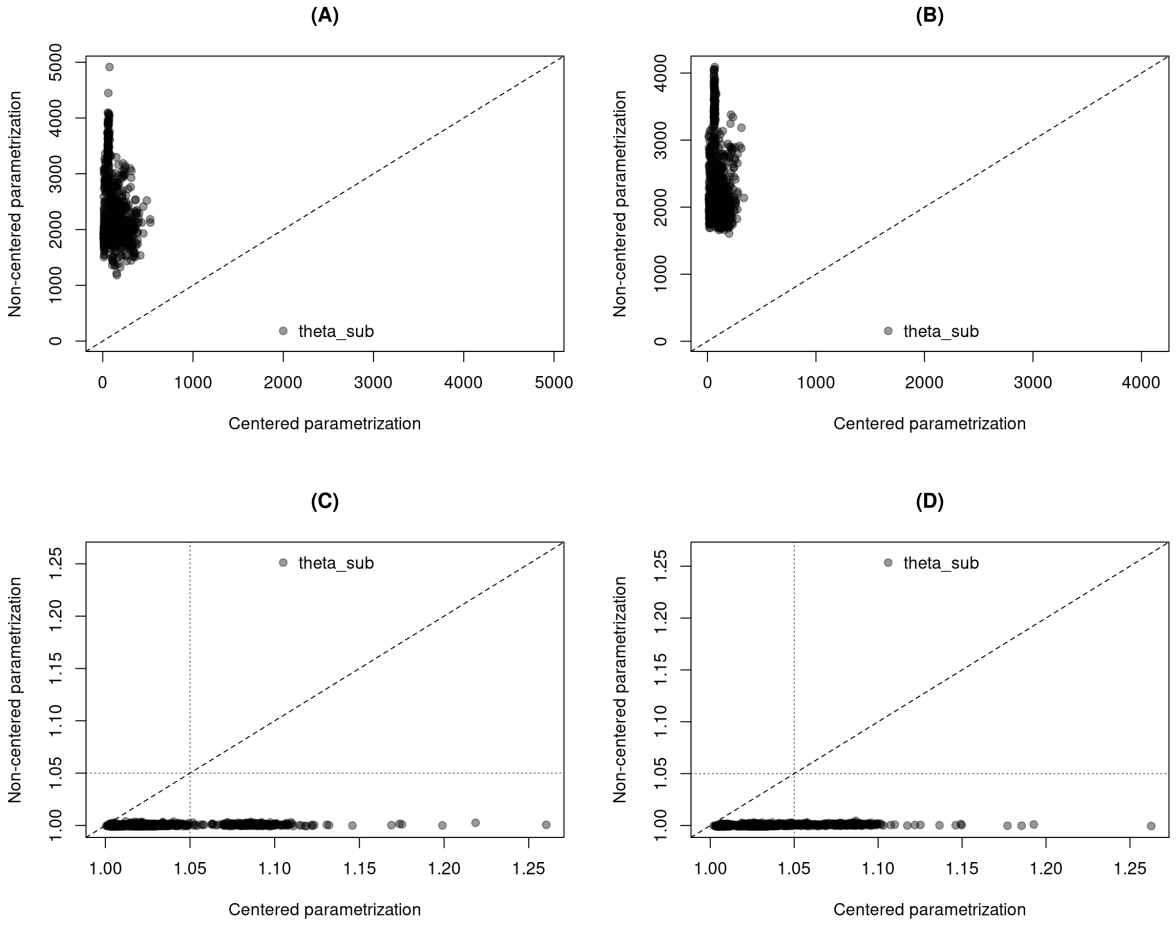

Figure A.27: First-order latent variable model (FOLV). Sample size 100, all replicas. CP and NCP comparison plot. (A) `n_eff` for the first sub-dimension. (B) `n_eff` for the second sub-dimensions. (C) `Rhat` for the first sub-dimension. (D) `Rhat` for the second sub-dimensions. Diagonal discontinuous line describes equality between CP and NCP. Vertical and horizontal discontinuous lines is set at `Rhat`= 1.05.

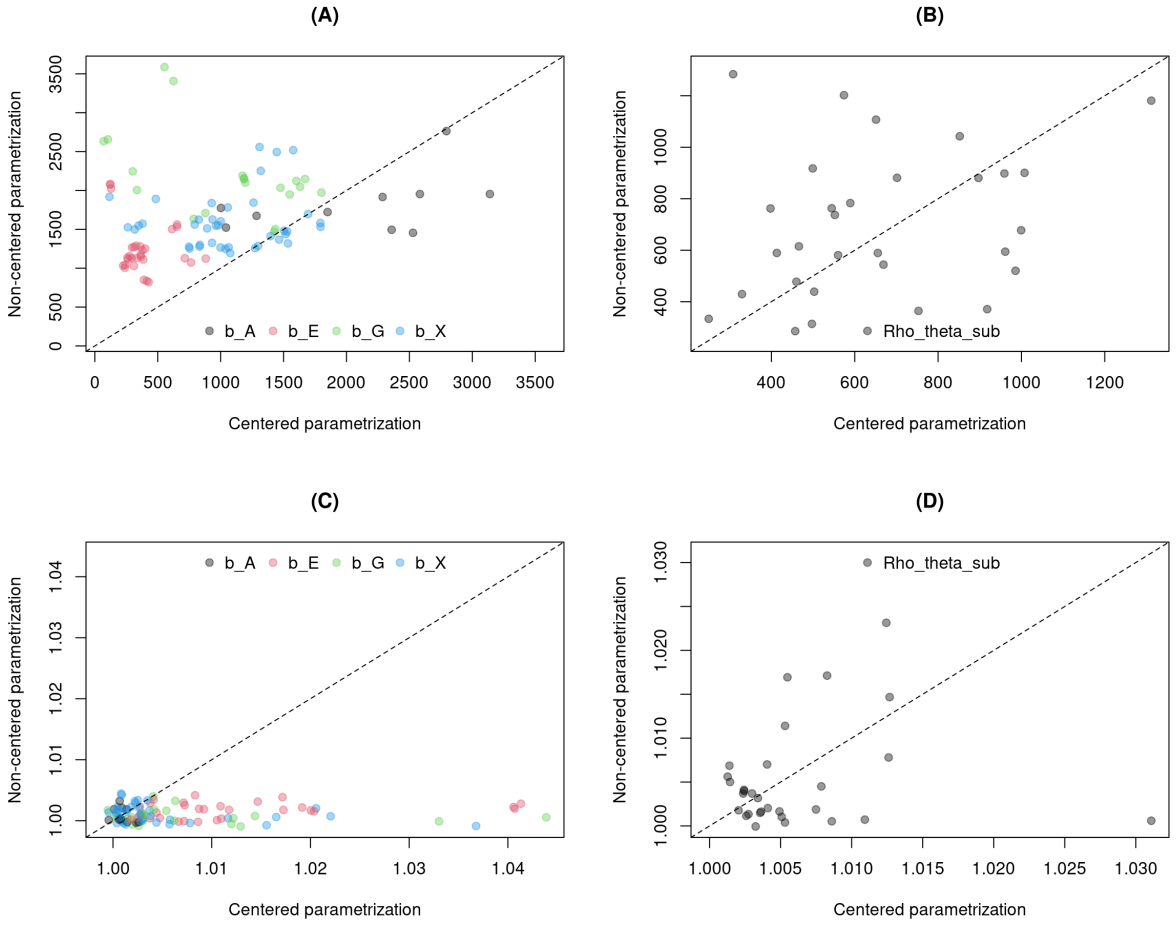

Figure A.28: First-order latent variable model (FOLV). Sample size 100, all replicas. CP and NCP comparison plot. (A) `n_eff` for regression parameters. (B) `n_eff` for correlations among sub-dimensions. (C) `Rhat` for regression parameters. (D) `Rhat` for correlations among sub-dimensions. Diagonal discontinuous line describes equality between CP and NCP. Vertical and horizontal discontinuous lines is set at `Rhat`= 1.05.

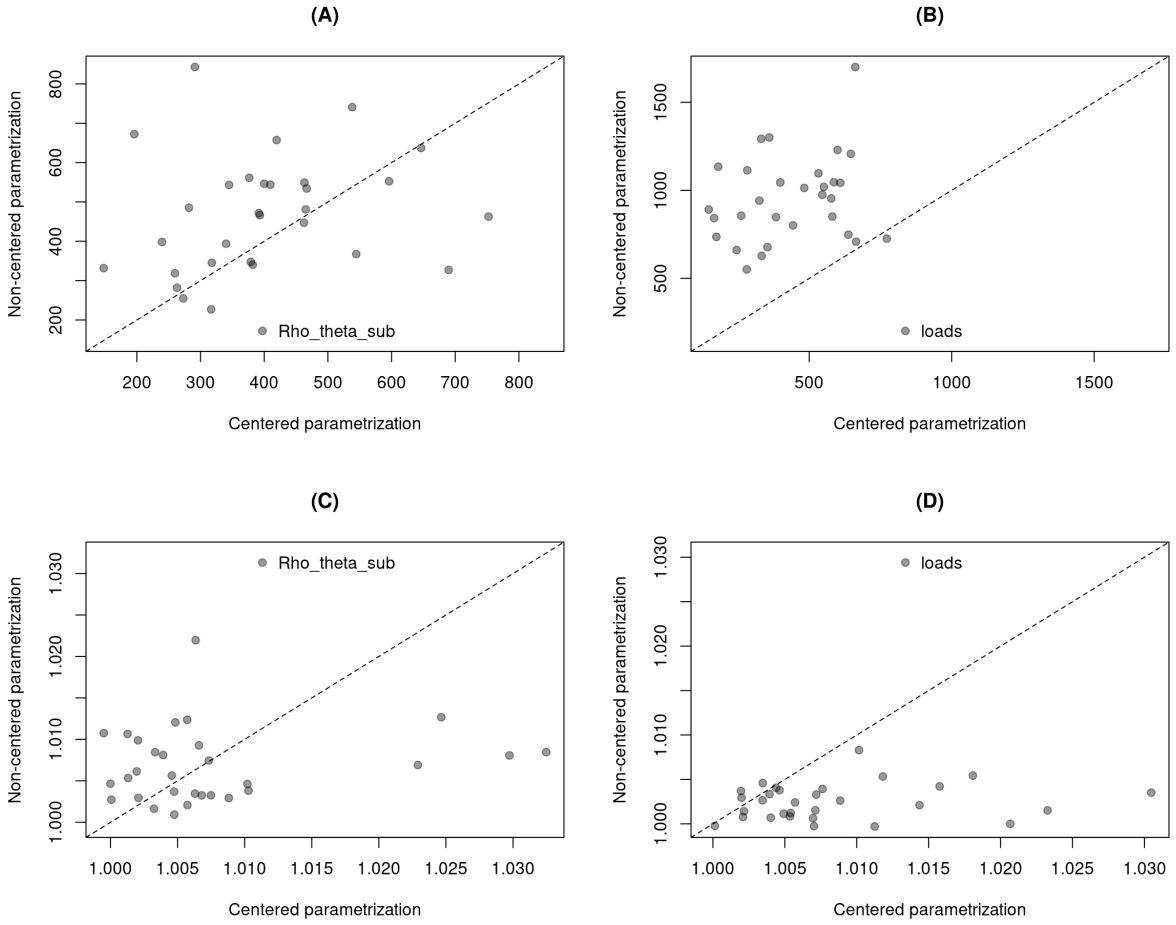

Figure A.29: Second-order latent variable model (SOLV). Sample size 100, all replicas. CP and NCP comparison plot. (A) `n_eff` for the correlations among sub-dimension. (B) `n_eff` for the loadings. (C) `Rhat` for the correlation among sub-dimensions. (D) `Rhat` for the loadings. Diagonal discontinuous line describes equality between CP and NCP. Vertical and horizontal discontinuous lines is set at `Rhat`= 1.05.

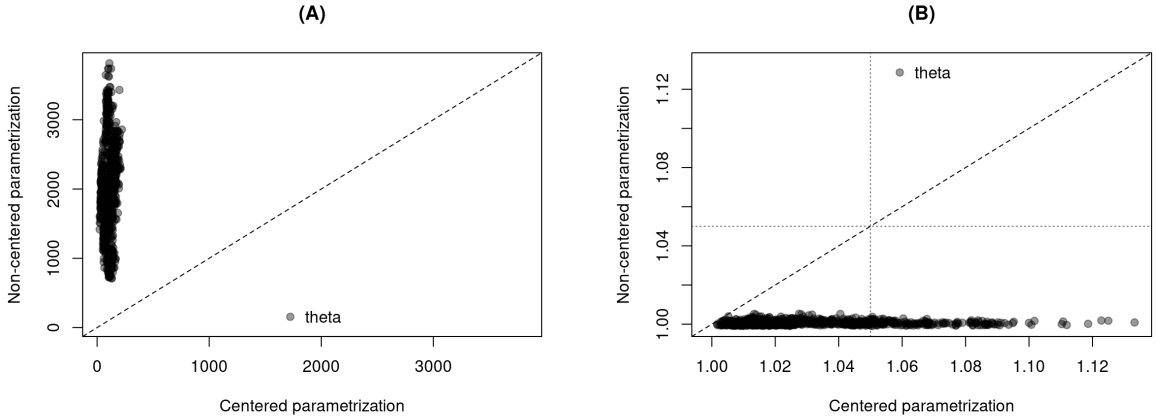

Figure A.30: Second-order latent variable model (SOLV). Sample size 100, all replicas. CP and NCP comparison plot. (A) `n_eff` for the higher-order latent variable. (B) `Rhat` for the higher-order latent variable. Diagonal discontinuous line describes equality between CP and NCP. Vertical and horizontal discontinuous lines is set at `Rhat`= 1.05.

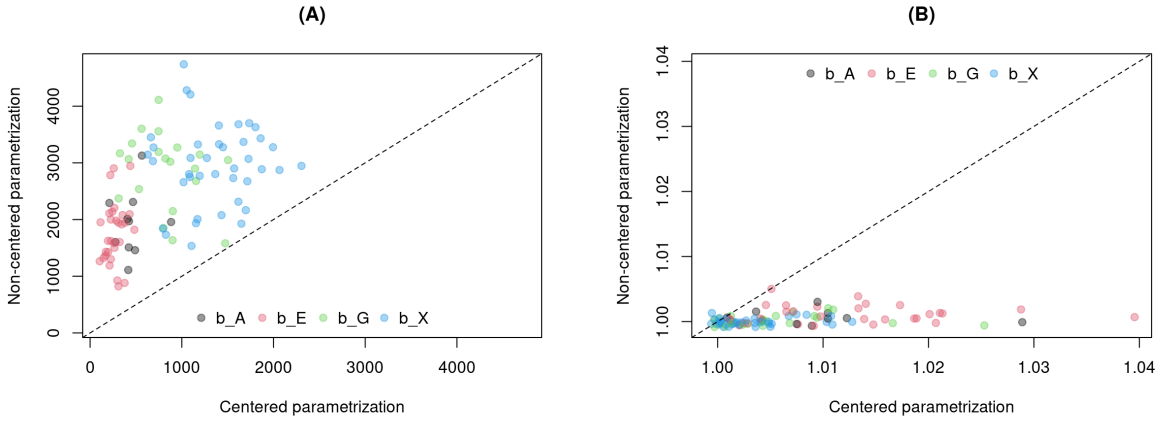

Figure A.31: Second-order latent variable model (SOLV). Sample size 100, all replicas. CP and NCP comparison plot. (A) `n_eff` for regression parameters. (B) `Rhat` for the first sub-dimension. (C) `Rhat` for regression parameters. (D) `Rhat` for the second sub-dimensions. Diagonal discontinuous line describes equality between CP and NCP. Vertical and horizontal discontinuous lines is set at `Rhat`= 1.05.

A.2.3 Recovery capacity

This section shows only a small set of tables about the recovery capacity of the model. In case the reader wants to replicate the calculations or access the full set of statistics refer to the "data/tables" section of the accompanying github page:
`https://github.com/jriveraespejo/thesis/tree/master/data_tables`

Moreover, in case the reader wants to inspect the recovery plots per model, parametrization, sample size, and replica refer to the "recovery plots" image section of the accompanying github page:
`https://github.com/jriveraespejo/thesis/tree/master/images/recovery_plots`

		Number of	RMSE_B $(\theta_{j1}^{(2)})$			
	Parametrization	individuals(replicas)	mean	sd	min	max
1	CP	100(10)	0.639	0.151	0.223	1.069
2	CP	250(10)	0.594	0.138	0.254	1.103
3	CP	500(10)	0.623	0.145	0.215	1.079
4	NCP	100(10)	0.632	0.148	0.256	1.032
5	NCP	250(10)	0.589	0.136	0.271	1.066
6	NCP	500(10)	0.610	0.143	0.214	1.087

Table A.1: First-order latent variable model (FOLV). Aggregated RMSE_B for the first individual sub-dimension.

		Number of	RMSE_B $(\theta_{j2}^{(2)})$			
	Parametrization	individuals(replicas)	mean	sd	min	max
1	CP	100(10)	0.608	0.148	0.318	0.930
2	CP	250(10)	0.599	0.134	0.256	0.939
3	CP	500(10)	0.618	0.143	0.228	1.056
4	NCP	100(10)	0.601	0.145	0.295	0.911
5	NCP	250(10)	0.593	0.135	0.253	0.963
6	NCP	500(10)	0.607	0.138	0.202	1.055

Table A.2: First-order latent variable model (FOLV). Aggregated RMSE_B for the second individual sub-dimension.

	Parametrization	Number of individuals(replicas)	RMSE$_B$ ($\theta^{(2)}_{j3}$) mean	sd	min	max
1	CP	100	0.604	0.142	0.257	1.000
2	CP	250	0.612	0.128	0.321	0.973
3	CP	500	0.612	0.150	0.172	1.079
4	NCP	100	0.595	0.139	0.245	0.968
5	NCP	250	0.609	0.131	0.288	0.990
6	NCP	500	0.601	0.147	0.220	1.082

Table A.3: First-order latent variable model (FOLV). Aggregated RMSE$_B$ for the third individual sub-dimension.

	Parametrization	Variable	Parameter	Number of replicas	Sample size 100	250	500
1	CP	Intercept	Γ_0	10	0.097	0.070	0.082
2	CP	gender(male)	$\Gamma_1[1]$	10	0.203	0.223	0.205
3	CP	gender(female)	$\Gamma_1[2]$	10	0.240	0.232	0.241
4	CP	age	Γ_2	10	0.008	0.006	0.003
5	CP	education(institute)	$\Gamma_3[1]$	10	0.249	0.113	0.119
6	CP	education(university)	$\Gamma_3[2]$	10	0.127	0.135	0.115
7	CP	education(both)	$\Gamma_3[3]$	10	0.157	0.113	0.110
8	CP	experience(0y)	$\Gamma_4[1]$	10	0.119	0.064	0.102
9	CP	experience(5y)	$\Gamma_4[2]$	10	0.157	0.082	0.082
10	CP	experience(10y)	$\Gamma_4[3]$	10	0.163	0.156	0.071
11	CP	experience(11+y)	$\Gamma_4[4]$	10	0.148	0.132	0.100
12	NCP	Intercept	Γ_0	10	0.089	0.065	0.083
13	NCP	gender(males)	$\Gamma_1[1]$	10	0.199	0.221	0.200
14	NCP	gender(females)	$\Gamma_1[2]$	10	0.247	0.229	0.235
15	NCP	age	Γ_2	10	0.008	0.006	0.003
16	NCP	education(institute)	$\Gamma_3[1]$	10	0.232	0.125	0.118
17	NCP	education(university)	$\Gamma_3[2]$	10	0.123	0.130	0.107
18	NCP	education(both)	$\Gamma_3[3]$	10	0.164	0.126	0.128
19	NCP	experience(0y)	$\Gamma_4[1]$	10	0.114	0.065	0.096
20	NCP	experience(5y)	$\Gamma_4[2]$	10	0.155	0.085	0.076
21	NCP	experience(10y)	$\Gamma_4[3]$	10	0.162	0.154	0.066
22	NCP	experience(11+y)	$\Gamma_4[4]$	10	0.150	0.133	0.095

Table A.4: First-order latent variable model (FOLV). RMSE$_B$ of regression parameters.

	Parametrization	Variable	Parameter	Number of replicas	Sample size 100	250	500
1	CP	females - males	$\Gamma_1[2] - \Gamma_1[1]$	10	0.131	0.112	0.088
2	CP	university - institute	$\Gamma_3[2] - \Gamma_3[1]$	10	0.239	0.167	0.081
3	CP	both - institute	$\Gamma_3[3] - \Gamma_3[1]$	10	0.267	0.079	0.084
4	CP	both - university	$\Gamma_3[3] - \Gamma_3[2]$	10	0.183	0.138	0.111
5	CP	$5y - 0y$	$\Gamma_4[2] - \Gamma_4[1]$	10	0.228	0.089	0.102
6	CP	$10y - 0y$	$\Gamma_4[3] - \Gamma_4[1]$	10	0.241	0.155	0.099
7	CP	$10y - 5y$	$\Gamma_4[3] - \Gamma_4[2]$	10	0.208	0.163	0.075
8	CP	$11{+}y - 0y$	$\Gamma_4[4] - \Gamma_4[1]$	10	0.196	0.147	0.076
9	CP	$11{+}y - 5y$	$\Gamma_4[4] - \Gamma_4[2]$	10	0.199	0.164	0.102
10	CP	$11{+}y - 10y$	$\Gamma_4[4] - \Gamma_4[3]$	10	0.197	0.137	0.087
11	NCP	females - males	$\Gamma_1[2] - \Gamma_1[1]$	10	0.134	0.113	0.088
12	NCP	university - institute	$\Gamma_3[2] - \Gamma_3[1]$	10	0.239	0.168	0.083
13	NCP	both - institute	$\Gamma_3[3] - \Gamma_3[1]$	10	0.267	0.079	0.083
14	NCP	both - university	$\Gamma_3[3] - \Gamma_3[2]$	10	0.182	0.139	0.111
15	NCP	$5y - 0y$	$\Gamma_4[2] - \Gamma_4[1]$	10	0.231	0.090	0.101
16	NCP	$10y - 0y$	$\Gamma_4[3] - \Gamma_4[1]$	10	0.239	0.155	0.097
17	NCP	$10y - 5y$	$\Gamma_4[3] - \Gamma_4[2]$	10	0.201	0.162	0.074
18	NCP	$11{+}y - 0y$	$\Gamma_4[4] - \Gamma_4[1]$	10	0.194	0.147	0.075
19	NCP	$11{+}y - 5y$	$\Gamma_4[4] - \Gamma_4[2]$	10	0.192	0.163	0.101
20	NCP	$11{+}y - 10y$	$\Gamma_4[4] - \Gamma_4[3]$	10	0.195	0.134	0.085

Table A.5: First-order latent variable model (FOLV). RMSE_B of contrast parameters.

	Parametrization	Parameter	Number of replicas	Sample size 100	250	500
1	CP	$\rho_{1,2}$	10	0.704	0.585	0.606
2	CP	$\rho_{1,3}$	10	0.719	0.591	0.617
3	CP	$\rho_{2,3}$	10	0.638	0.623	0.615
4	NCP	$\rho_{1,2}$	10	0.700	0.585	0.605
5	NCP	$\rho_{1,3}$	10	0.721	0.591	0.617
6	NCP	$\rho_{2,3}$	10	0.636	0.623	0.615

Table A.6: First-order latent variable model (FOLV). RMSE_B of correlations among sub-dimensions.

	Parametrization	Parameter	Number of replicas	Sample size 100	250	500
1	CP	$\eta_1^{(3)}$	10	0.133	0.185	0.305
2	CP	$\eta_2^{(3)}$	10	0.242	0.183	0.258
3	CP	$\eta_3^{(3)}$	10	0.197	0.144	0.180
4	CP	$\eta_4^{(3)}$	10	0.261	0.214	0.310
5	CP	$\eta_5^{(3)}$	10	0.191	0.343	0.245
6	NCP	$\eta_1^{(3)}$	10	0.115	0.147	0.256
7	NCP	$\eta_2^{(3)}$	10	0.228	0.193	0.190
8	NCP	$\eta_3^{(3)}$	10	0.219	0.141	0.170
9	NCP	$\eta_4^{(3)}$	10	0.266	0.204	0.252
10	NCP	$\eta_5^{(3)}$	10	0.189	0.318	0.230

Table A.7: First-order latent variable model (FOLV). RMSE_B of texts difficulties.

	Parametrization	Parameter	Number of replicas	Sample size 100	250	500
1	CP	$\sigma_1^{(3)}$	10	0.200	0.126	0.231
2	CP	$\sigma_2^{(3)}$	10	0.229	0.155	0.163
3	CP	$\sigma_3^{(3)}$	10	0.196	0.127	0.197
4	CP	$\sigma_4^{(3)}$	10	0.198	0.147	0.150
5	CP	$\sigma_5^{(3)}$	10	0.243	0.223	0.184
6	NCP	$\sigma_1^{(3)}$	10	0.202	0.134	0.236
7	NCP	$\sigma_2^{(3)}$	10	0.235	0.159	0.163
8	NCP	$\sigma_3^{(3)}$	10	0.202	0.129	0.196
9	NCP	$\sigma_4^{(3)}$	10	0.200	0.152	0.144
10	NCP	$\sigma_5^{(3)}$	10	0.242	0.220	0.181

Table A.8: First-order latent variable model (FOLV). RMSE_B of texts difficulty deviations.

	Parametrization	Sample size	Number of items(replicas)	RMSE_B $(\eta_k^{(2)})$ mean	sd	min	max
1	CP	100	25(10)	0.301	0.061	0.159	0.389
2	CP	250	25(10)	0.203	0.039	0.138	0.278
3	CP	500	25(10)	0.201	0.031	0.137	0.275
4	NCP	100	25(10)	0.289	0.055	0.159	0.391
5	NCP	250	25(10)	0.188	0.036	0.130	0.268
6	NCP	500	25(10)	0.155	0.037	0.090	0.264

Table A.9: First-order latent variable model (FOLV). Aggregated RMSE_B for items difficulties.

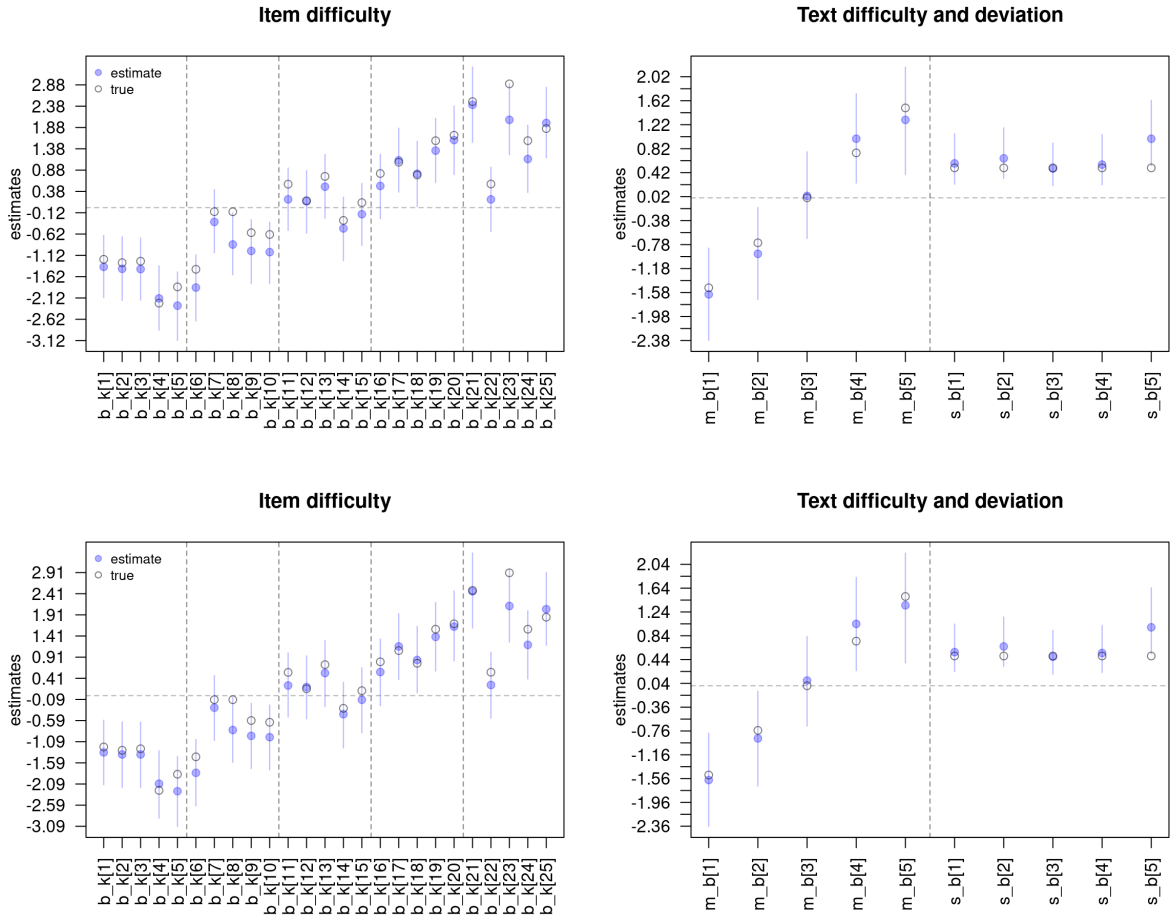

Figure A.32: First-order latent variable model (FOLV). Sample size 100, replica 1. Items and text parameters. Top two panels correspond to the centered parametrization (CP). Top bottom panels correspond to the non-centered parametrization (NCP).

	Parametrization	Number of individuals(replicas)	RMSE$_B$ ($\theta_{j1}^{(2)}$) mean	sd	min	max
1	CP	100(10)	0.693	0.152	0.293	1.014
2	CP	250(10)	0.628	0.143	0.289	1.115
3	CP	500(10)	0.669	0.157	0.289	1.212
4	NCP	100(10)	0.683	0.148	0.308	1.016
5	NCP	250(10)	0.616	0.142	0.298	1.108
6	NCP	500(10)	0.667	0.156	0.267	1.185

Table A.10: Second-order latent variable model (SOLV). Aggregated RMSE$_B$ for the first individual sub-dimension.

		Number of	RMSE$_B$ ($\theta^{(2)}_{j2}$)			
	Parametrization	individuals(replicas)	mean	sd	min	max
1	CP	100(10)	0.628	0.158	0.276	0.955
2	CP	250(10)	0.653	0.135	0.318	1.087
3	CP	500(10)	0.649	0.145	0.257	1.133
4	NCP	100(10)	0.623	0.157	0.250	0.955
5	NCP	250(10)	0.631	0.135	0.276	1.059
6	NCP	500(10)	0.645	0.144	0.270	1.139

Table A.11: Second-order latent variable model (SOLV). Aggregated RMSE$_B$ for the second individual sub-dimension.

		Number of	RMSE$_B$ ($\theta^{(2)}_{j3}$)			
	Parametrization	individulas(replicas)	mean	sd	min	max
1	CP	100(10)	0.617	0.137	0.279	0.992
2	CP	250(10)	0.664	0.142	0.352	1.119
3	CP	500(10)	0.651	0.153	0.263	1.163
4	NCP	100(10)	0.606	0.137	0.247	0.962
5	NCP	250(10)	0.647	0.138	0.324	1.089
6	NCP	500(10)	0.648	0.153	0.252	1.175

Table A.12: Second-order latent variable model (SOLV). Aggregated RMSE$_B$ for the third individual sub-dimension.

		Number of	RMSE$_B$ ($\theta^{(3)}_{j}$)			
	Parametrization	individuals(replicas)	mean	sd	min	max
1	CP	100(10)	0.736	0.166	0.390	1.077
2	CP	250(10)	0.908	0.185	0.489	1.388
3	CP	500(10)	1.131	0.232	0.438	1.857
4	NCP	100(10)	0.717	0.156	0.362	1.072
5	NCP	250(10)	0.831	0.179	0.468	1.281
6	NCP	500(10)	1.119	0.224	0.501	1.746

Table A.13: Second-order latent variable model (SOLV). Aggregated RMSE$_B$ for the individual higher-order dimension.

	Parametrization	Variable	Parameter	Number of replicas	Sample size 100	250	500
1	CP	Intercept	Γ_0	10	0.11	0.12	0.16
2	CP	gender(male)	$\Gamma_1[1]$	10	0.31	0.48	0.48
3	CP	gender(female)	$\Gamma_1[2]$	10	0.17	0.12	0.13
4	CP	age	Γ_2	10	0.02	0.01	0.03
5	CP	education(institute)	$\Gamma_3[1]$	10	0.37	0.51	0.56
6	CP	education(university)	$\Gamma_3[2]$	10	0.49	0.59	0.81
7	CP	education(both)	$\Gamma_3[3]$	10	0.27	0.18	0.24
8	CP	experience($0y$)	$\Gamma_4[1]$	10	0.22	0.46	0.77
9	CP	experience($5y$)	$\Gamma_4[2]$	10	0.21	0.16	0.18
10	CP	experience($10y$)	$\Gamma_4[3]$	10	0.18	0.13	0.30
11	CP	experience($11{+}y$)	$\Gamma_4[4]$	10	0.14	0.27	0.39
12	NCP	Intercept	Γ_0	10	0.11	0.12	0.16
13	NCP	gender(males)	$\Gamma_1[1]$	10	0.31	0.47	0.48
14	NCP	gender(females)	$\Gamma_1[2]$	10	0.16	0.10	0.12
15	NCP	age	Γ_2	10	0.02	0.01	0.03
16	NCP	education(institute)	$\Gamma_3[1]$	10	0.35	0.46	0.56
17	NCP	education(university)	$\Gamma_3[2]$	10	0.50	0.63	0.79
18	NCP	education(both)	$\Gamma_3[3]$	10	0.27	0.19	0.22
19	NCP	experience($0y$)	$\Gamma_4[1]$	10	0.22	0.45	0.77
20	NCP	experience($5y$)	$\Gamma_4[2]$	10	0.21	0.15	0.19
21	NCP	experience($10y$)	$\Gamma_4[3]$	10	0.19	0.13	0.30
22	NCP	experience($11{+}y$)	$\Gamma_4[4]$	10	0.14	0.28	0.39

Table A.14: Second-order latent variable model (SOLV). RMSE_B of regression parameters.

	Parametrization	Contrast	Parameter	Number of replicas	Sample size 100	250	500
1	CP	females - males	$\Gamma_1[2] - \Gamma_1[1]$	10	0.29	0.51	0.56
2	CP	university - institute	$\Gamma_3[2] - \Gamma_3[1]$	10	0.78	1.06	1.32
3	CP	both - institute	$\Gamma_3[3] - \Gamma_3[1]$	10	0.57	0.58	0.72
4	CP	both - university	$\Gamma_3[3] - \Gamma_3[2]$	10	0.43	0.52	0.67
5	CP	$5y - 0y$	$\Gamma_4[2] - \Gamma_4[1]$	10	0.30	0.41	0.65
6	CP	$10y - 0y$	$\Gamma_4[3] - \Gamma_4[1]$	10	0.34	0.50	1.07
7	CP	$10y - 5y$	$\Gamma_4[3] - \Gamma_4[2]$	10	0.29	0.23	0.46
8	CP	$11{+}y - 0y$	$\Gamma_4[4] - \Gamma_4[1]$	10	0.24	0.70	1.15
9	CP	$11{+}y - 5y$	$\Gamma_4[4] - \Gamma_4[2]$	10	0.30	0.41	0.56
10	CP	$11{+}y - 10y$	$\Gamma_4[4] - \Gamma_4[3]$	10	0.28	0.32	0.21
11	NCP	females - males	$\Gamma_1[2] - \Gamma_1[1]$	10	0.29	0.51	0.57
12	NCP	university - institute	$\Gamma_3[2] - \Gamma_3[1]$	10	0.79	1.05	1.33
13	NCP	both - institute	$\Gamma_3[3] - \Gamma_3[1]$	10	0.57	0.58	0.72
14	NCP	both - university	$\Gamma_3[3] - \Gamma_3[2]$	10	0.43	0.51	0.68
15	NCP	$5y - 0y$	$\Gamma_4[2] - \Gamma_4[1]$	10	0.30	0.41	0.65
16	NCP	$10y - 0y$	$\Gamma_4[3] - \Gamma_4[1]$	10	0.34	0.49	1.06
17	NCP	$10y - 5y$	$\Gamma_4[3] - \Gamma_4[2]$	10	0.30	0.23	0.46
18	NCP	$11{+}y - 0y$	$\Gamma_4[4] - \Gamma_4[1]$	10	0.24	0.69	1.15
19	NCP	$11{+}y - 5y$	$\Gamma_4[4] - \Gamma_4[2]$	10	0.30	0.40	0.56
20	NCP	$11{+}y - 10y$	$\Gamma_4[4] - \Gamma_4[3]$	10	0.28	0.32	0.21

Table A.15: Second-order latent variable model (SOLV). RMSE_B of contrast parameters.

	Parametrization	Parameter	Number of replicas	Sample size 100	250	500
1	CP	$\rho_{1,2}$	10	0.20	0.26	0.23
2	CP	$\rho_{1,3}$	10	0.15	0.27	0.21
3	CP	$\rho_{2,3}$	10	0.27	0.22	0.22
4	NCP	$\rho_{1,2}$	10	0.20	0.26	0.23
5	NCP	$\rho_{1,3}$	10	0.15	0.26	0.20
6	NCP	$\rho_{2,3}$	10	0.27	0.22	0.22

Table A.16: Second-order latent variable model (SOLV). RMSE_B of correlations among sub-dimensions.

	Parametrization	Parameter	Number of replicas	Sample size 100	250	500
1	CP	$\lambda_1^{(2)}$	10	0.35	0.51	0.52
2	CP	$\lambda_2^{(2)}$	10	0.43	0.47	0.54
3	CP	$\lambda_3^{(2)}$	10	0.42	0.47	0.53
4	NCP	$\lambda_1^{(2)}$	10	0.35	0.51	0.53
5	NCP	$\lambda_2^{(2)}$	10	0.43	0.47	0.54
6	NCP	$\lambda_3^{(2)}$	10	0.42	0.47	0.53

Table A.17: Second-order latent variable model (SOLV). RMSE_B of the loadings for each among sub-dimension.

	Parametrization	Parameter	Number of replicas	Sample size 100	250	500
1	CP	$\eta_1^{(3)}$	10	0.133	0.185	0.305
2	CP	$\eta_2^{(3)}$	10	0.242	0.183	0.258
3	CP	$\eta_3^{(3)}$	10	0.197	0.144	0.180
4	CP	$\eta_4^{(3)}$	10	0.261	0.214	0.310
5	CP	$\eta_5^{(3)}$	10	0.191	0.343	0.245
11	NCP	$\eta_1^{(3)}$	10	0.115	0.147	0.256
12	NCP	$\eta_2^{(3)}$	10	0.228	0.193	0.190
13	NCP	$\eta_3^{(3)}$	10	0.219	0.141	0.170
14	NCP	$\eta_4^{(3)}$	10	0.266	0.204	0.252
15	NCP	$\eta_5^{(3)}$	10	0.189	0.318	0.230

Table A.18: Second-order latent variable model (SOLV). RMSE_B of texts difficulties.

	Parametrization	Parameter	Number of replicas	Sample size 100	250	500
6	CP	$\sigma_1^{(3)}$	10	0.200	0.126	0.231
7	CP	$\sigma_2^{(3)}$	10	0.229	0.155	0.163
8	CP	$\sigma_3^{(3)}$	10	0.196	0.127	0.197
9	CP	$\sigma_4^{(3)}$	10	0.198	0.147	0.150
10	CP	$\sigma_5^{(3)}$	10	0.243	0.223	0.184
16	NCP	$\sigma_1^{(3)}$	10	0.202	0.134	0.236
17	NCP	$\sigma_2^{(3)}$	10	0.235	0.159	0.163
18	NCP	$\sigma_3^{(3)}$	10	0.202	0.129	0.196
19	NCP	$\sigma_4^{(3)}$	10	0.200	0.152	0.144
20	NCP	$\sigma_5^{(3)}$	10	0.242	0.220	0.181

Table A.19: Second-order latent variable model (SOLV). RMSE_B of texts difficulty deviations.

	Parametrization	Sample size	Number of items(replicas)	RMSE_B $(\eta_k^{(2)})$ mean	sd	min	max
1	CP	100	25(10)	0.203	0.060	0.147	0.267
2	CP	250	25(10)	0.248	0.027	0.217	0.266
3	CP	500	25(10)	0.218	0.011	0.206	0.227
4	NCP	100	25(10)	0.206	0.059	0.152	0.268
5	NCP	250	25(10)	0.246	0.026	0.216	0.264
6	NCP	500	25(10)	0.218	0.012	0.204	0.227

Table A.20: Second-order latent variable model (SOLV). Aggregated RMSE_B for items difficulties.

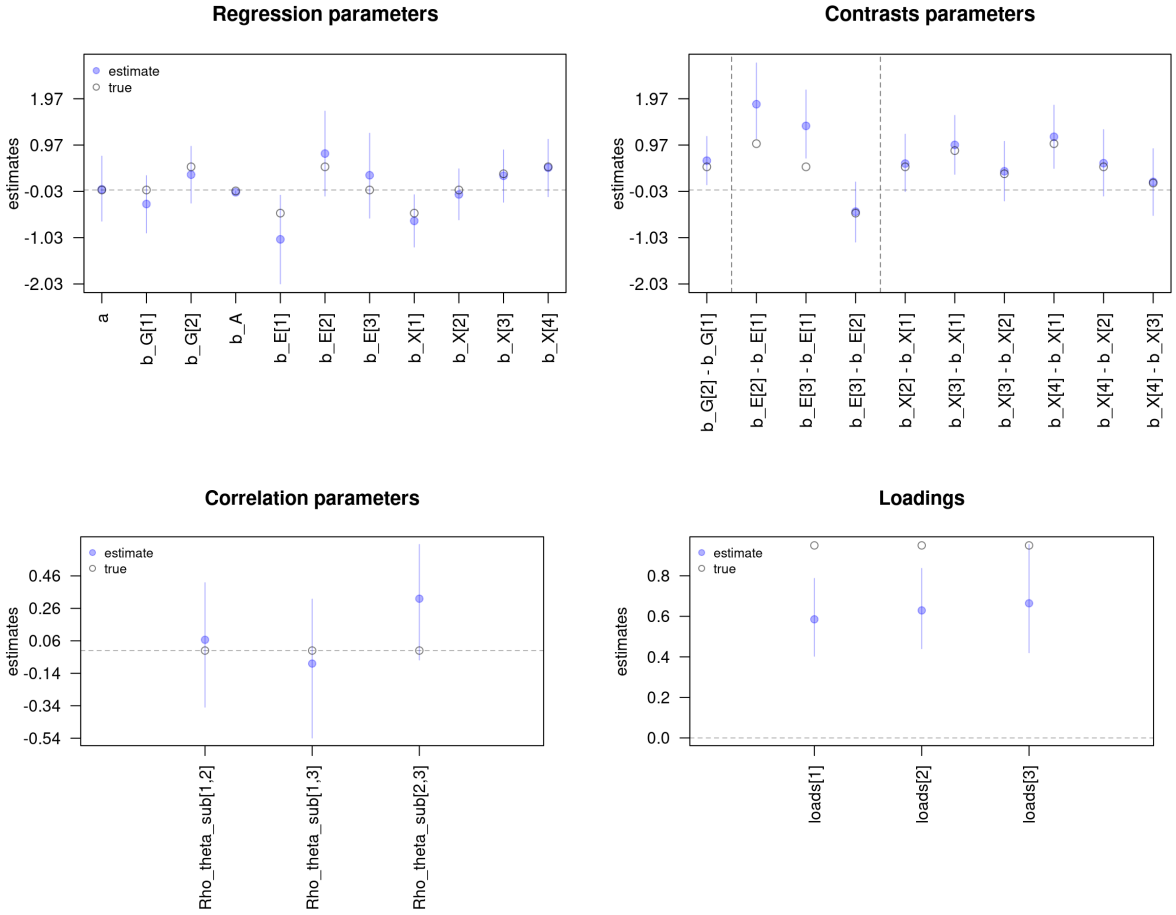

Figure A.33: Second-order latent variable model (SOLV). Centered parametrization. Sample size 100, replica 1. Regression, contrast, correlation, and loading parameters. The true correlation parameters corresponds to simulation values.

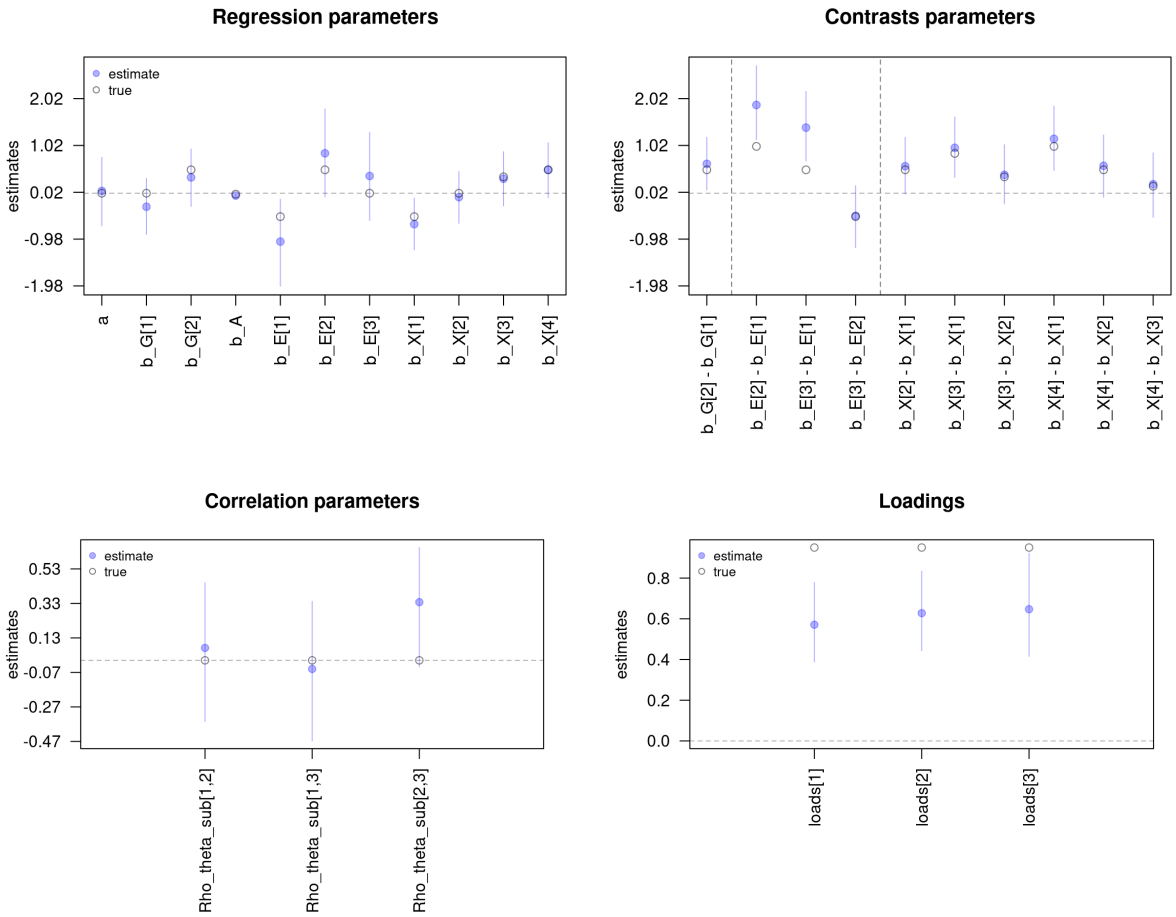

Figure A.34: Second-order latent variable model (SOLV). Centered parametrization. Sample size 100, replica 1. Regression, contrast, correlation, and loading parameters. The true correlation parameters corresponds to simulation values.

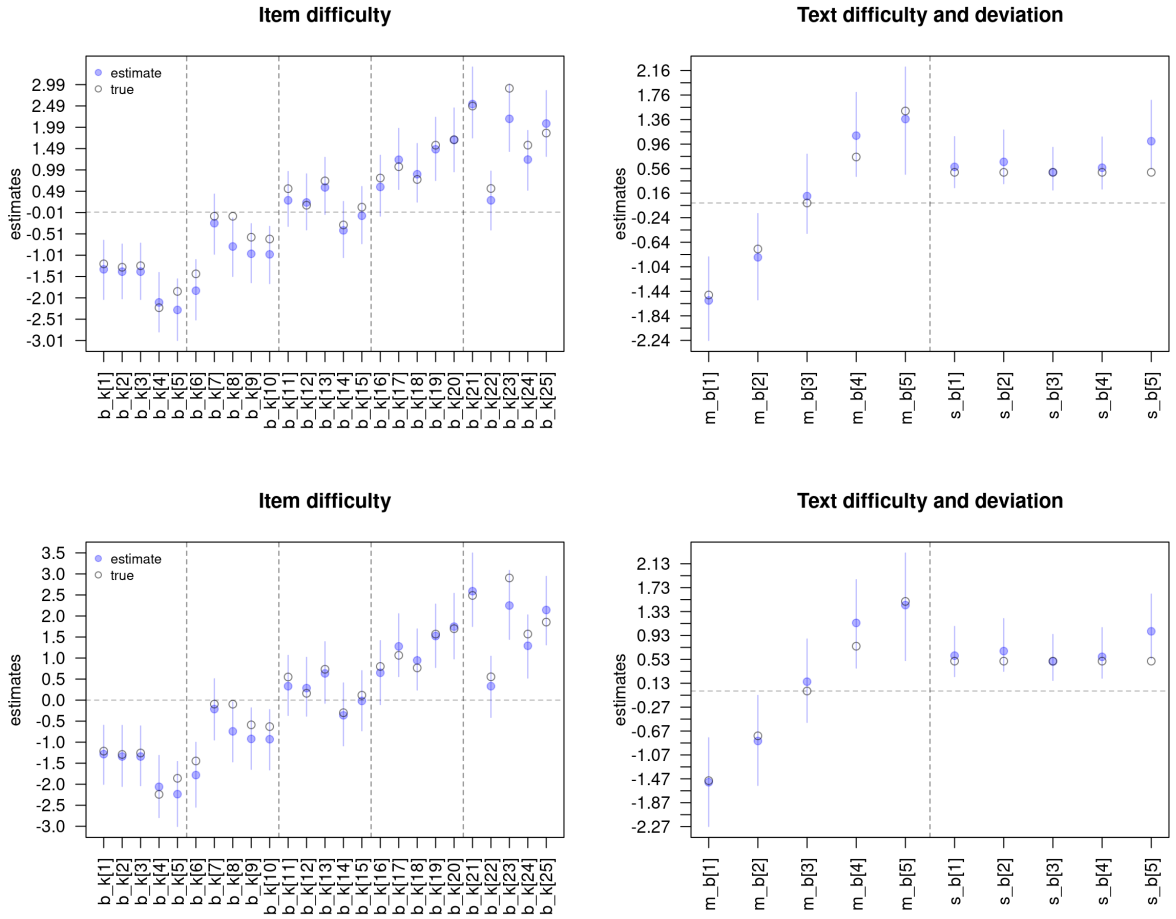

Figure A.35: Second-order latent variable model (SOLV). Sample size 100, replica 1. Items and text parameters. Top two panels correspond to the centered parametrization (CP). Top bottom panels correspond to the non-centered parametrization (NCP).

A.2.4 Retrodictive accuracy

The current section shows only a small set of figures related to the retrodictive accuracy of the model. In case the reader wants to inspect the full set of figures refer to the "retrodictive plots" image section of the accompanying github page:
`https://github.com/jriveraespejo/thesis/tree/master/images/retrodictive_plots`

On the other hand, if the reader wants to replicate or inspect the retrodictive tables refer to the "data/tables" section of the accompanying github page:
`https://github.com/jriveraespejo/thesis/tree/master/data_tables`

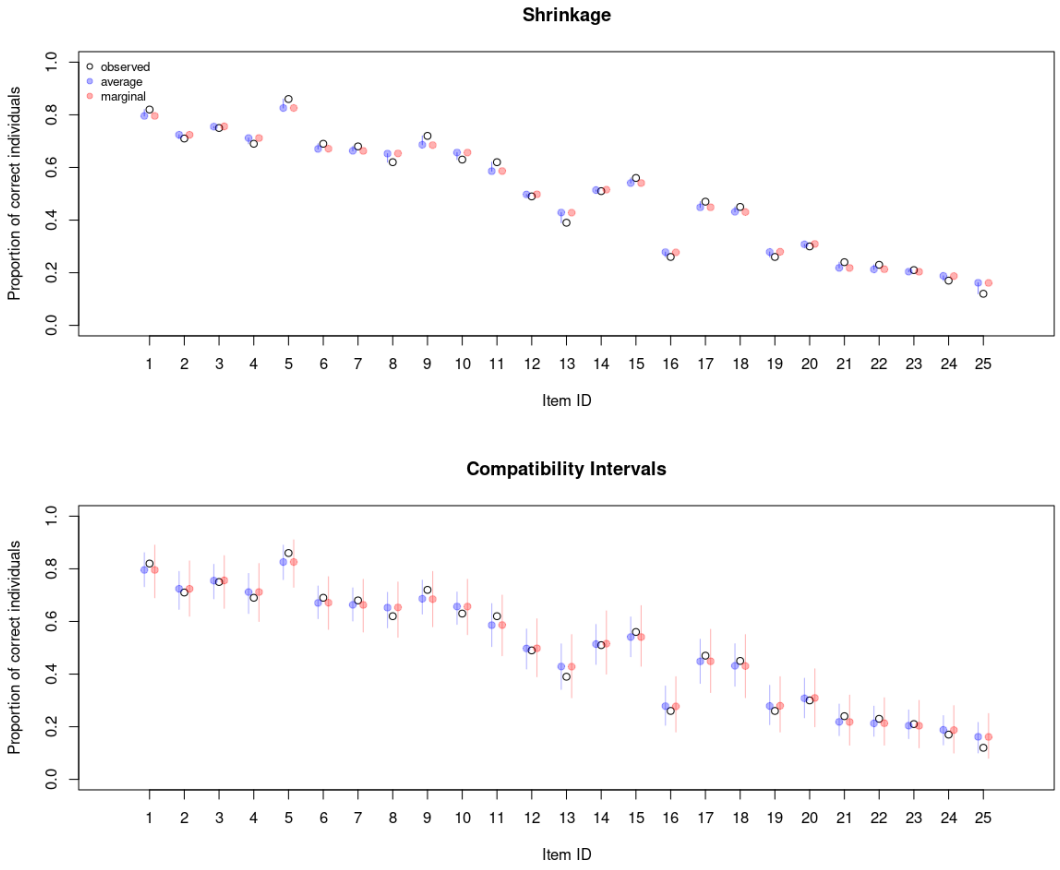

Figure A.36: First-order latent variable model (FOLV). Sample size 100, replica number 4. Non-centered parametrization. Items predictive plot.

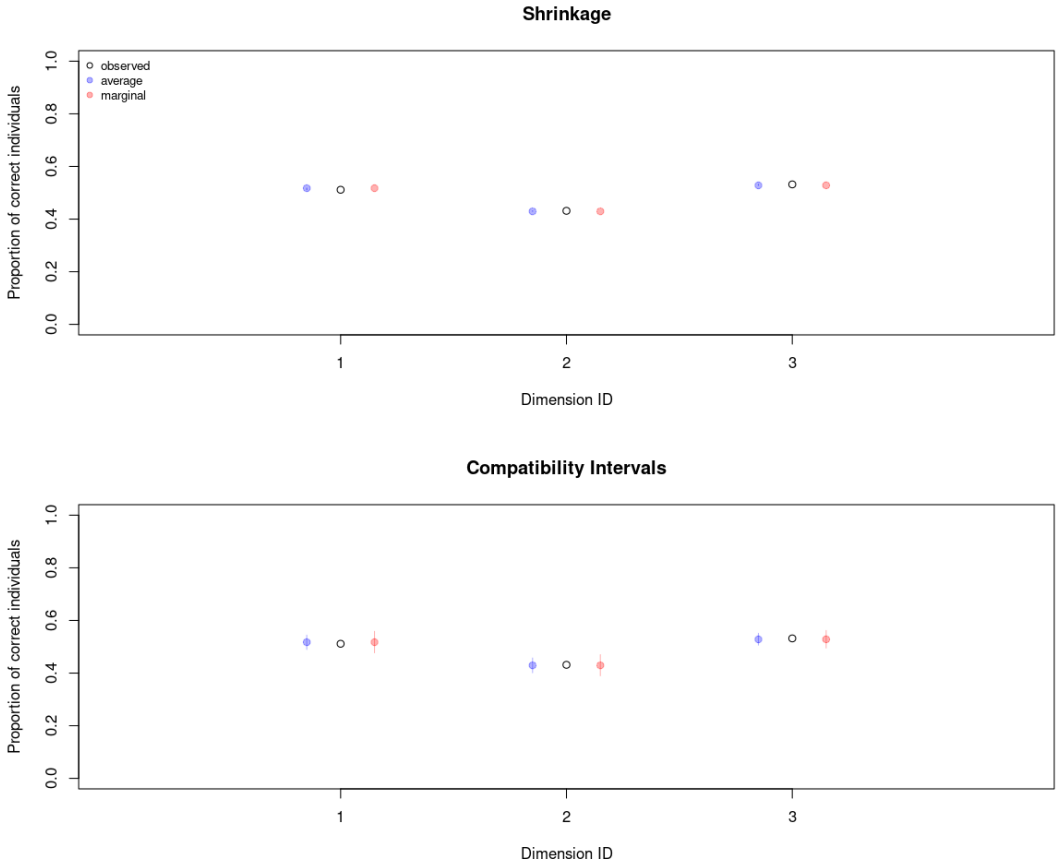

Figure A.37: First-order latent variable model (FOLV). Sample size 100, replica number 4. Non-centered parametrization. Dimension predictive plot.

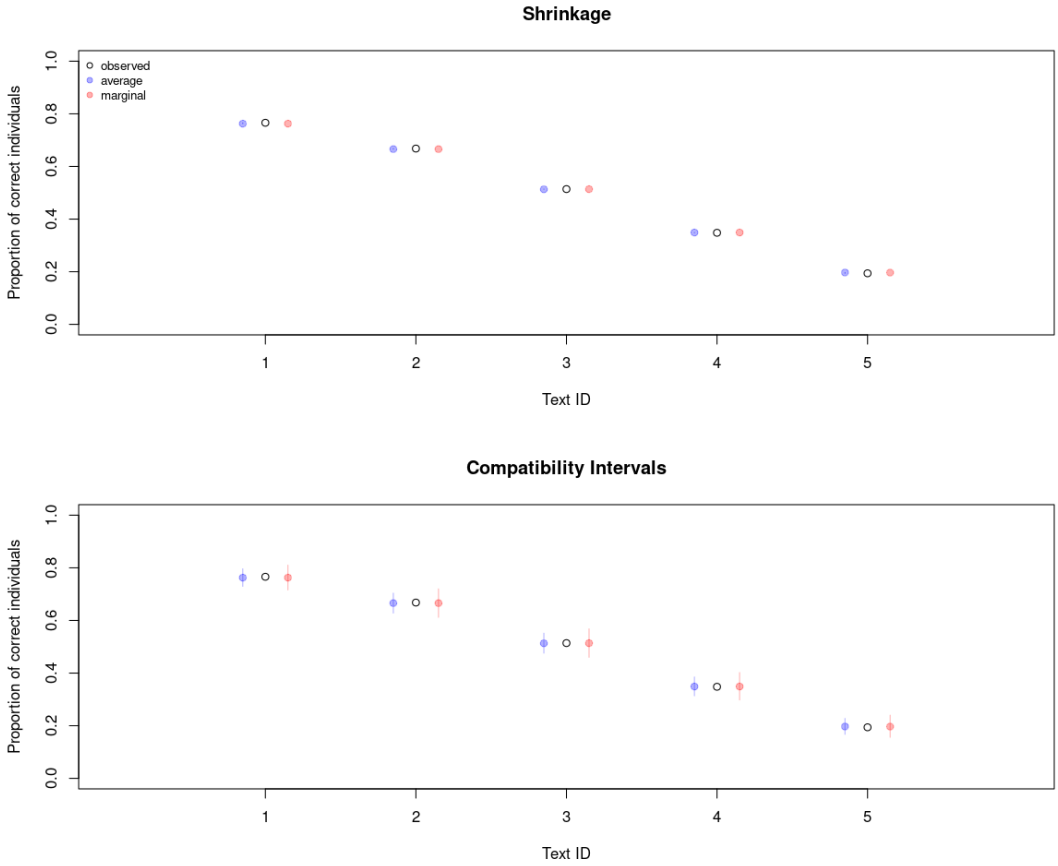

Figure A.38: First-order latent variable model (FOLV). Sample size 100, replica number 4. Non-centered parametrization. Text predictive plot.

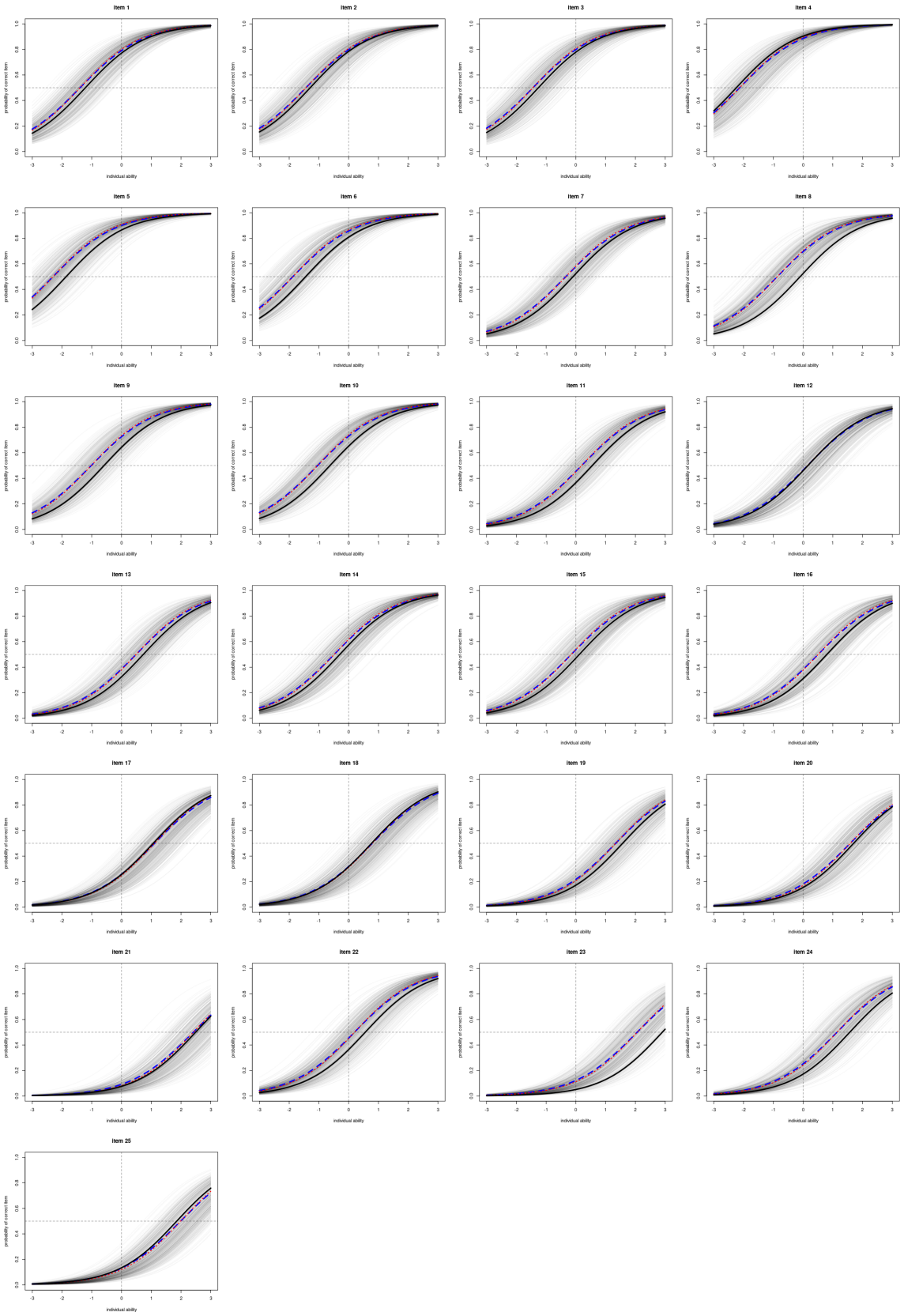

Figure A.39: First-order latent variable model (FOLV). Sample size 100, replica number 1. Centered parametrization. Item characteristic curves (ICC). (black solid line) true ICC, (gray solid line) simulated predicted ICC, (blue discontinuous line) average predicted ICC, (red discontinuous line) marginal predicted ICC.

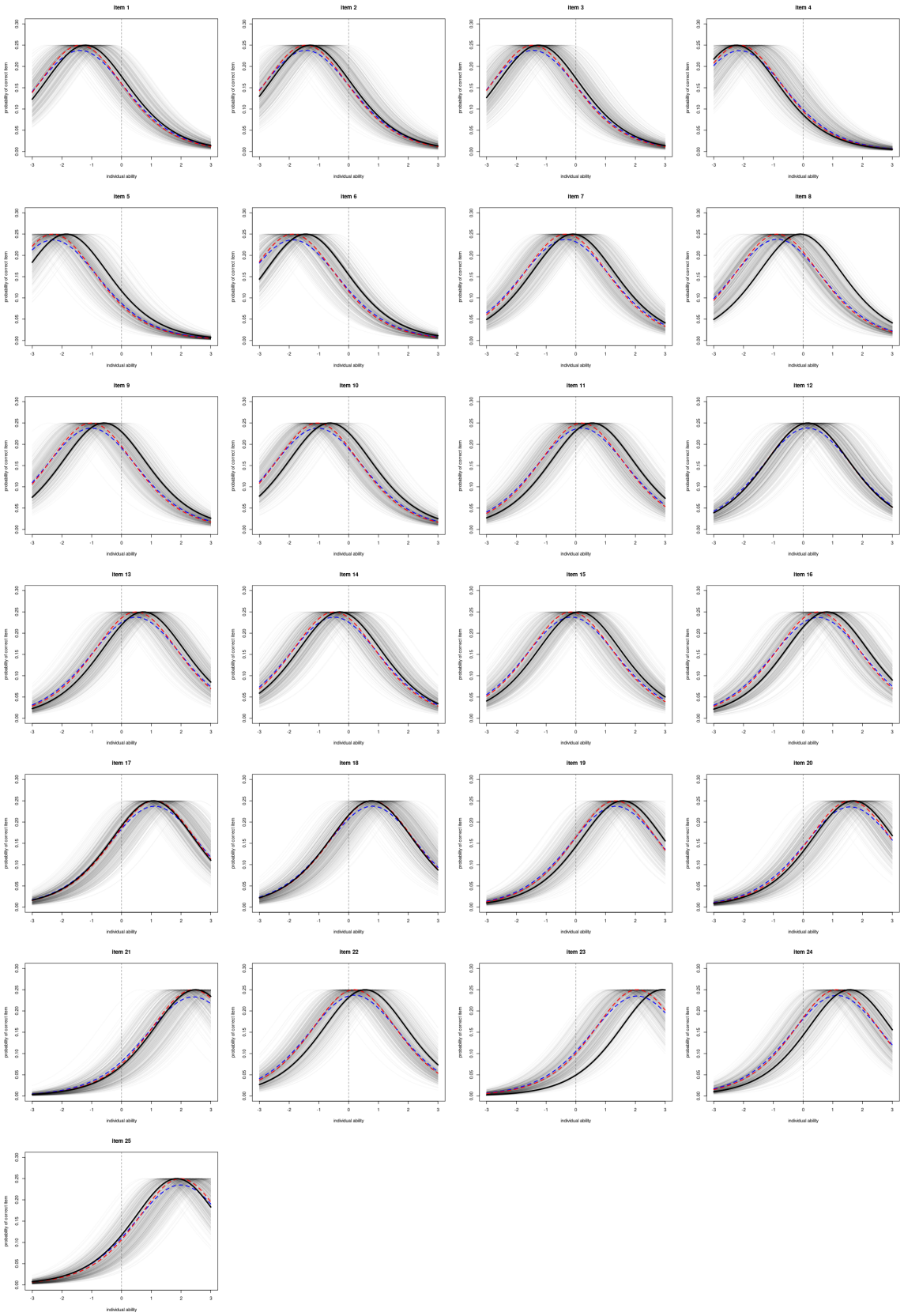

Figure A.40: First-order latent variable model (FOLV). Sample size 100, replica number 1. Non-centered parametrization. Item information function (IIF). (black solid line) true ICC, (gray solid line) simulated predicted ICC, (blue discontinuous line) average predicted ICC, (red discontinuous line) marginal predicted ICC.

		Sample	$\overline{\text{RMSE}}_W$			RMSE_B		
	Parametrization	size	mean	min	max	mean	min	max
1	CP	100	0.026	0.024	0.028	0.002	0.001	0.004
2	CP	250	0.016	0.015	0.017	0.001	0.000	0.001
3	CP	500	0.011	0.010	0.012	0.000	0.000	0.001
4	NCP	100	0.026	0.023	0.027	0.002	0.001	0.003
5	NCP	250	0.016	0.015	0.018	0.001	0.000	0.002
6	NCP	500	0.011	0.010	0.012	0.000	0.000	0.001

Table A.21: First-order latent variable model (FOLV). Centered and non-centered parametrization. Within and between replicas items predictive RMSE.

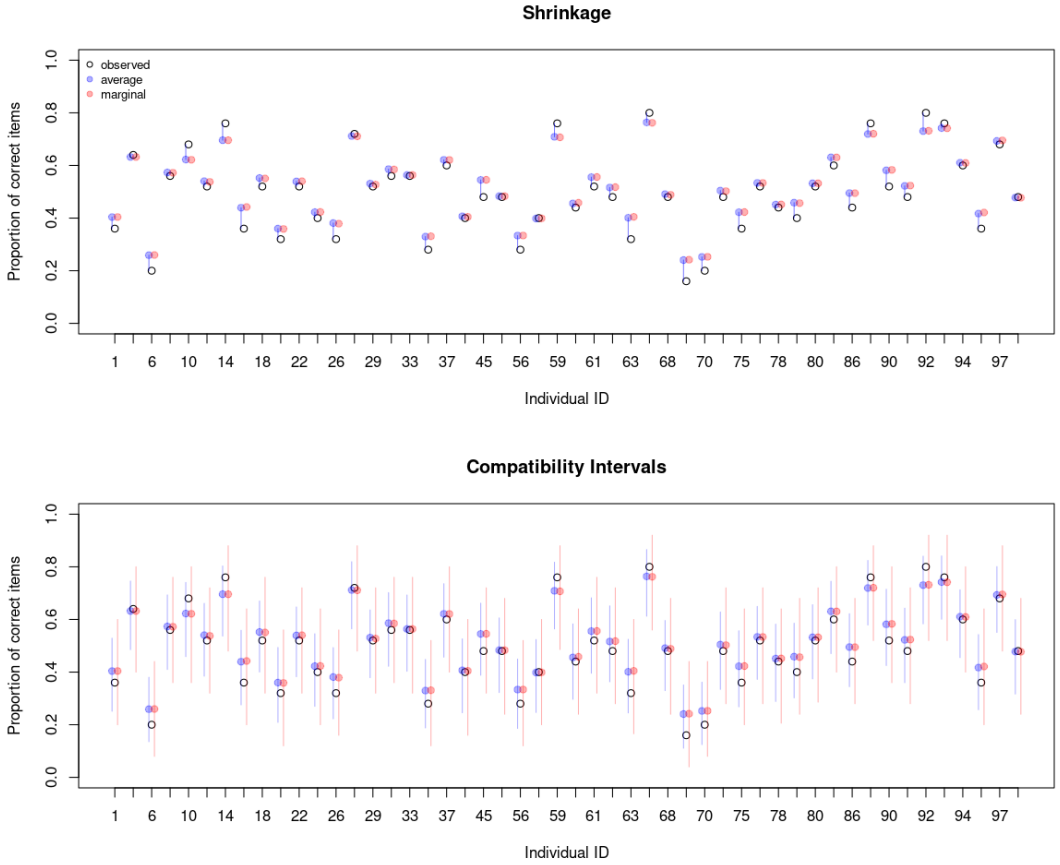

Figure A.41: Second-order latent variable model (SOLV). Sample size 100, replica number 4. Non-centered parametrization. Individual predictive plot.

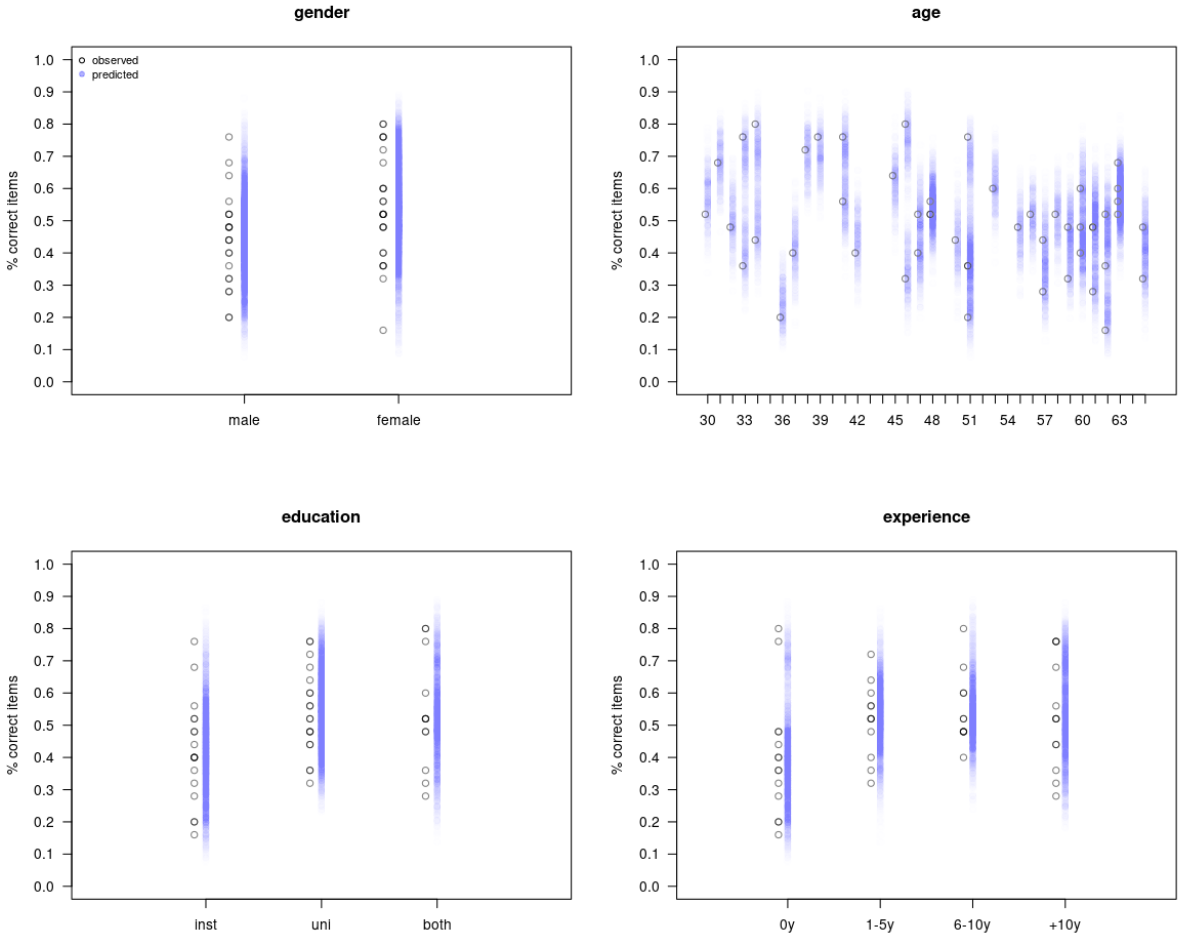

Figure A.42: Second-order latent variable model (SOLV). Sample size 100, replica number 4. Non-centered parametrization. Individual predictive plot per covariate.

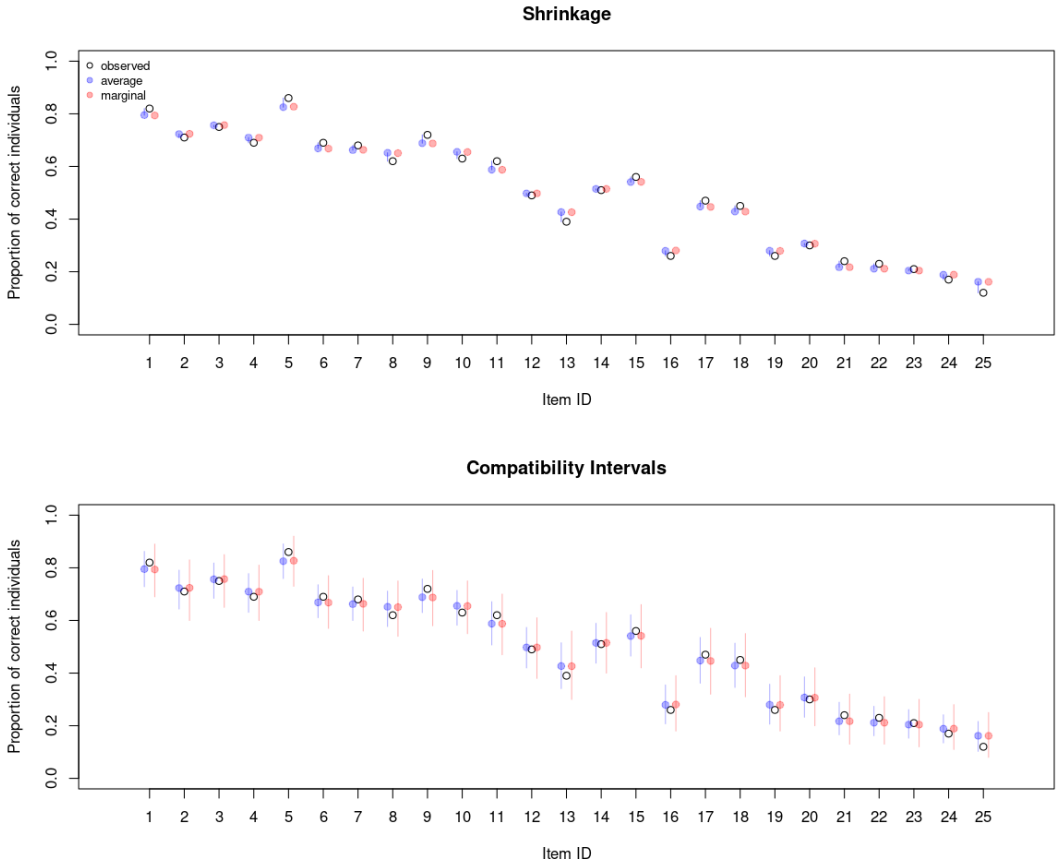

Figure A.43: Second-order latent variable model (SOLV). Sample size 100, replica number 4. Non-centered parametrization. Items predictive plot.

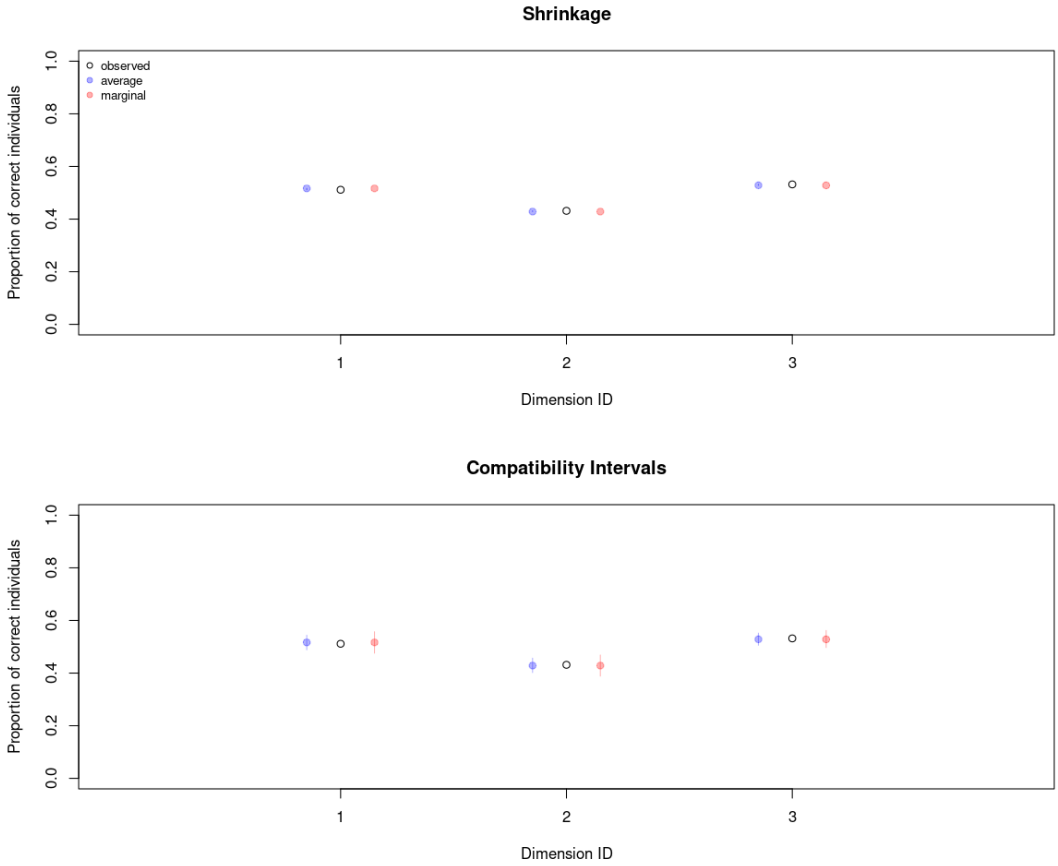

Figure A.44: Second-order latent variable model (SOLV). Sample size 100, replica number 4. Non-centered parametrization. Dimension predictive plot.

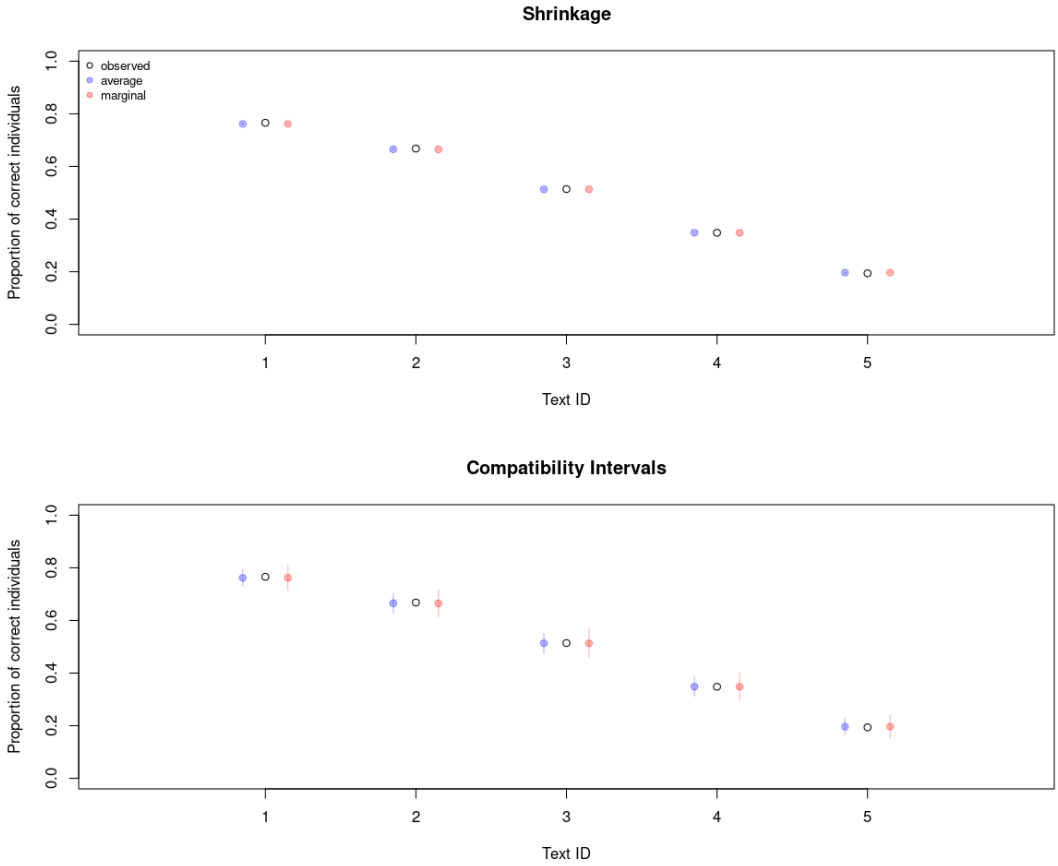

Figure A.45: Second-order latent variable model (SOLV). Sample size 100, replica number 4. Non-centered parametrization. Text predictive plot.

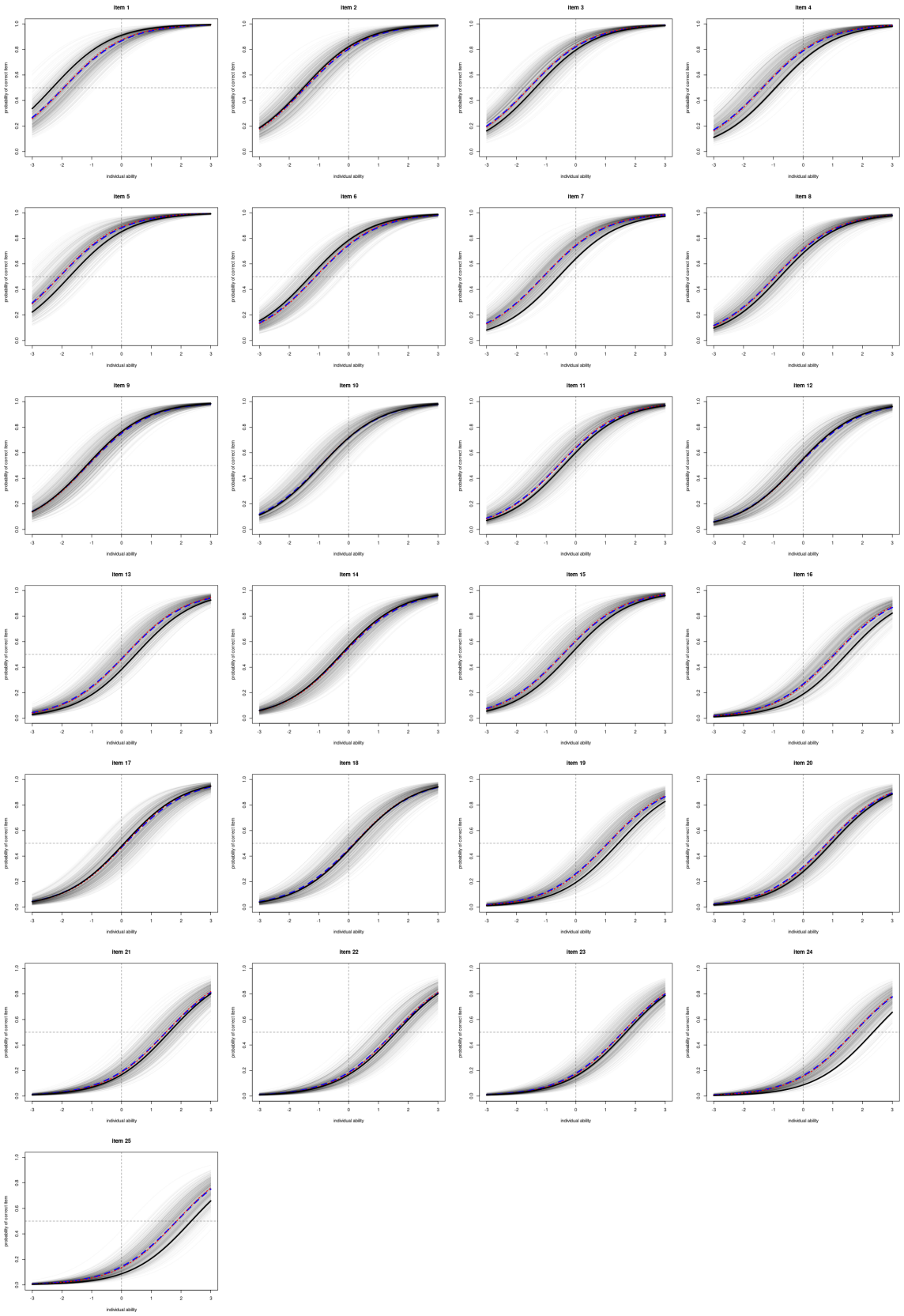

Figure A.46: Second-order latent variable model (SOLV). Sample size 100, replica number 4. Non-centered parametrization. Item characteristic curves (ICC). (black solid line) true ICC, (gray solid line) simulated predicted ICC, (blue discontinuous line) average predicted ICC, (red discontinuous line) marginal predicted ICC.

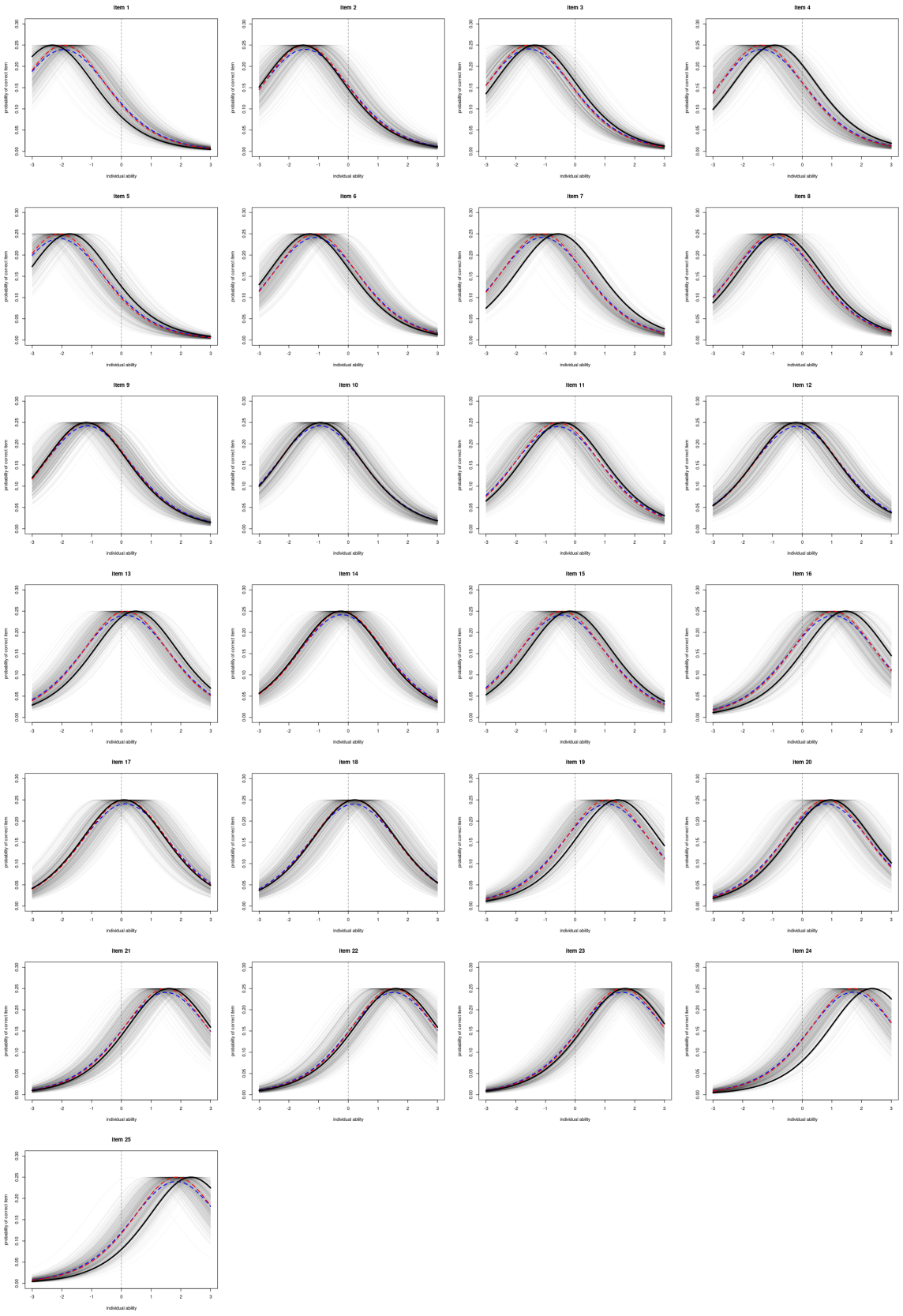

Figure A.47: Second-order latent variable model (SOLV). Sample size 100, replica number 4. Non-centered parametrization. Item information function (IIF). (black solid line) true ICC, (gray solid line) simulated predicted ICC, (blue discontinuous line) average predicted ICC, (red discontinuous line) marginal predicted ICC.

		Sample	$\overline{\text{RMSE}}_W$			RMSE_B		
	Parametrization	size	mean	min	max	mean	min	max
1	CP	100	0.025	0.023	0.027	0.002	0.001	0.003
2	CP	250	0.016	0.015	0.017	0.001	0.000	0.002
3	CP	500	0.011	0.010	0.012	0.001	0.000	0.001
4	NCP	100	0.025	0.023	0.027	0.002	0.001	0.003
5	NCP	250	0.016	0.015	0.017	0.001	0.000	0.002
6	NCP	500	0.011	0.010	0.012	0.001	0.000	0.001

Table A.22: Second-order latent variable model (SOLV). Centered and non-centered parametrization. Within and between replicas items predictive RMSE.

A.3 Chapter 5: Application

This section shows only a small set of figures related to the MCMC chains for the application. In case the reader wants to inspect any other plot produced for this section refer to the "application plots" image section of the accompanying github page:
`https://github.com/jriveraespejo/thesis/tree/master/images/application_plots`

A.3.1 Parametrization performance

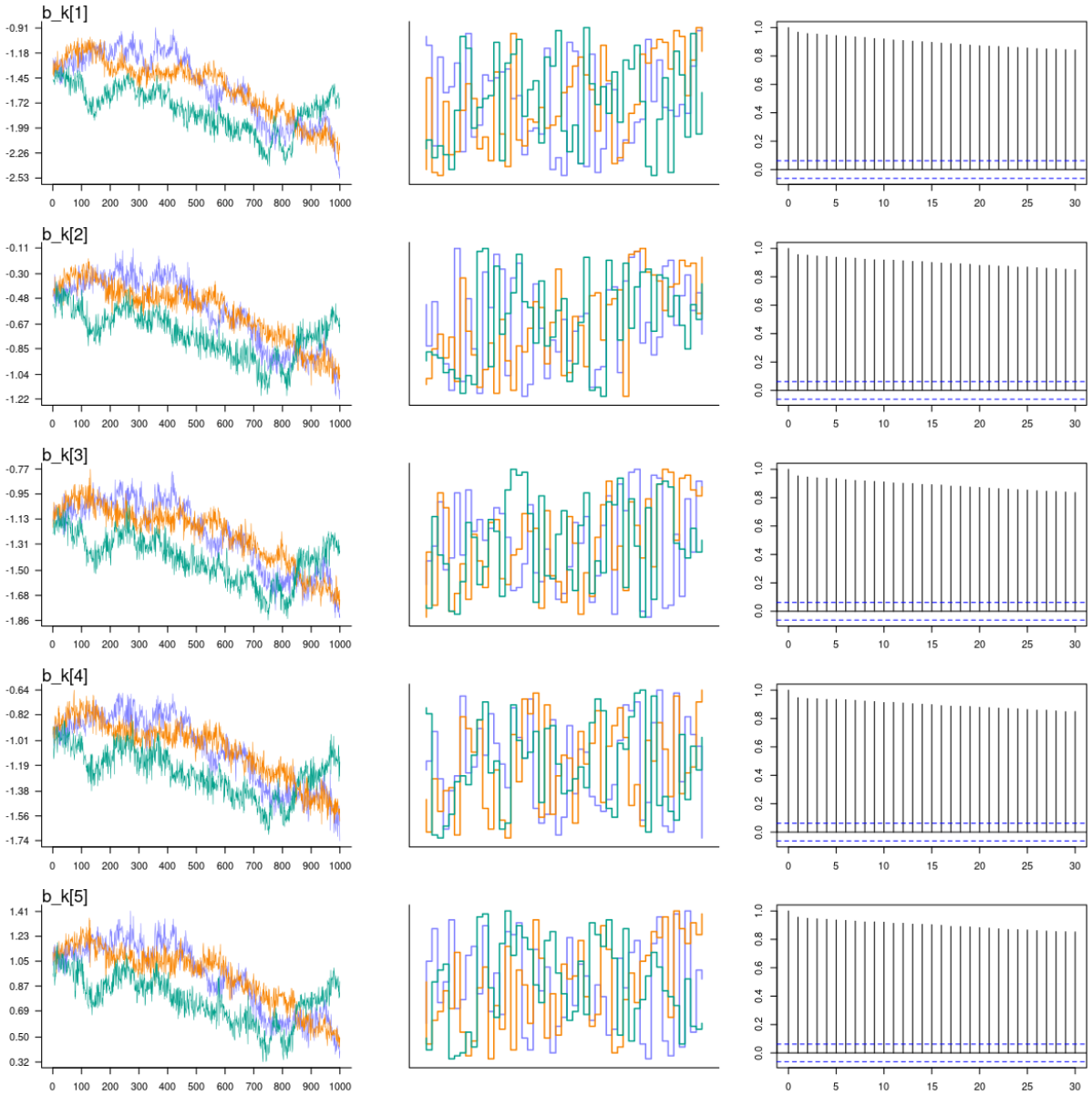

Figure A.48: Application's second-order latent variable model (SOLV). Centered parametrization. Items difficulty: (Left) trace plot, (Middle) trank plot, (Right) autocorrelation plot.

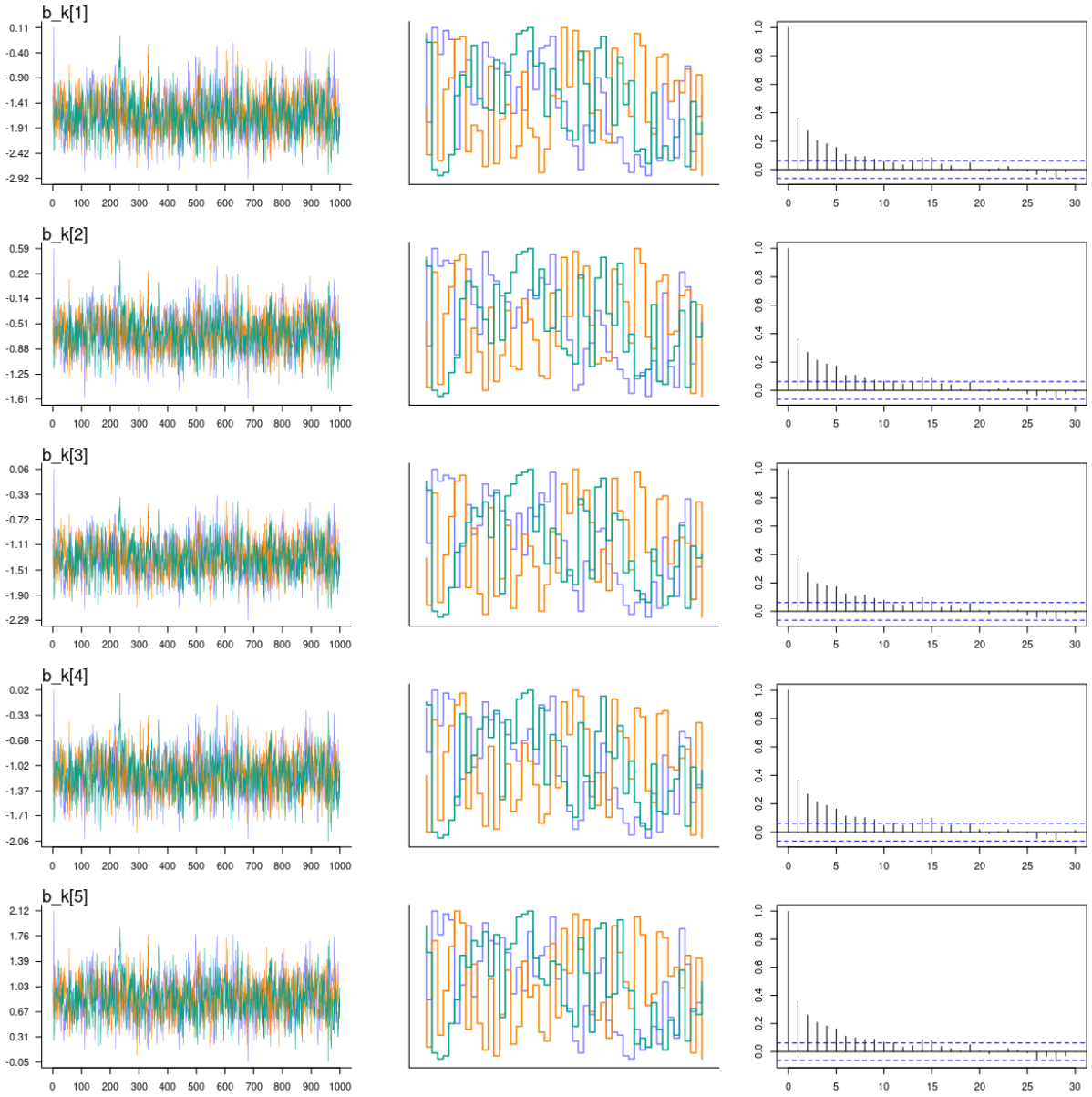

Figure A.49: Application's second-order latent variable model (SOLV). Non-centered parametrization. Items difficulty: (Left) trace plot, (Middle) trank plot, (Right) autocorrelation plot.

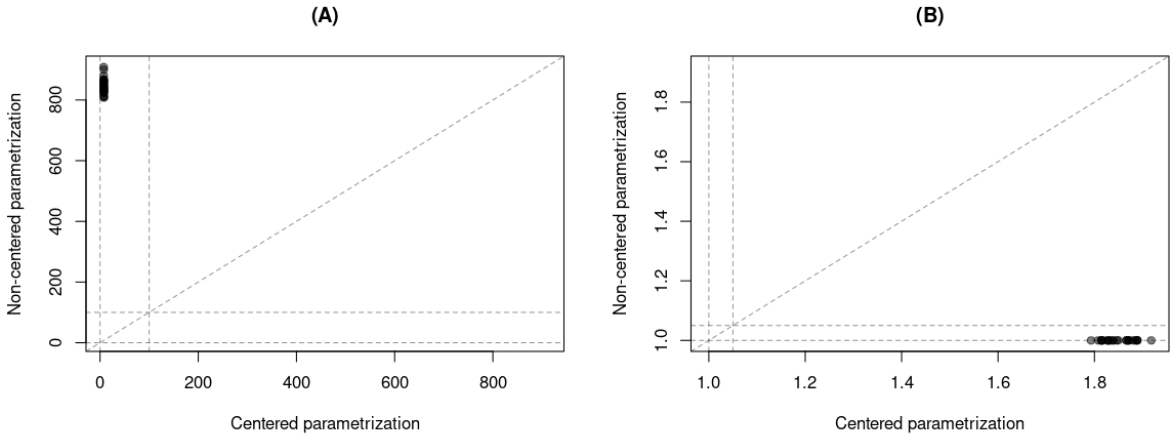

Figure A.50: Application's second-order latent variable model (SOLV). CP and NCP comparison plot. (A) `n_eff` for items' difficulties. (B) `Rhat` for items' difficulties. Diagonal discontinuous line describes equality between CP and NCP. Vertical and horizontal discontinuous lines set in A corresponds to `n_eff`= 100. Vertical and horizontal discontinuous lines set in B corresponds to `Rhat`= 1.05.

A.3.2 Retrodictive accuracy

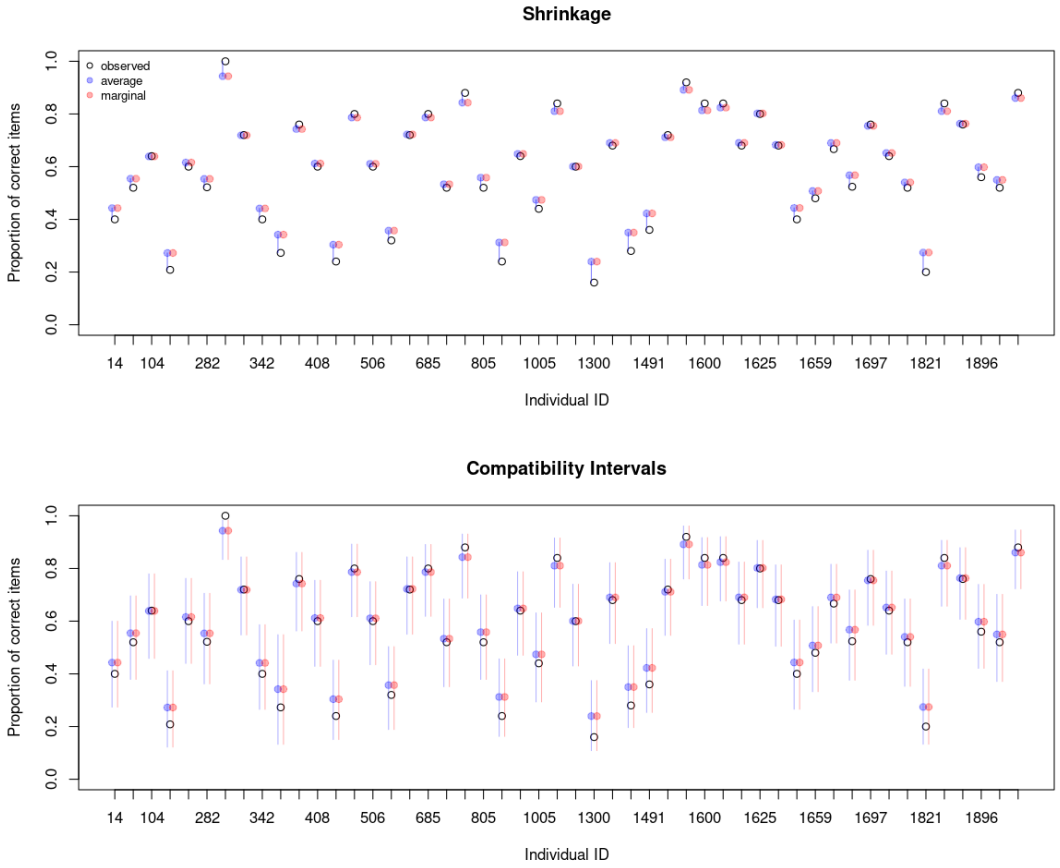

Figure A.51: Second-order latent variable model (SOLV). Non-centered parametrization. Individual predictive plot.

A.3.3 Psychometric properties

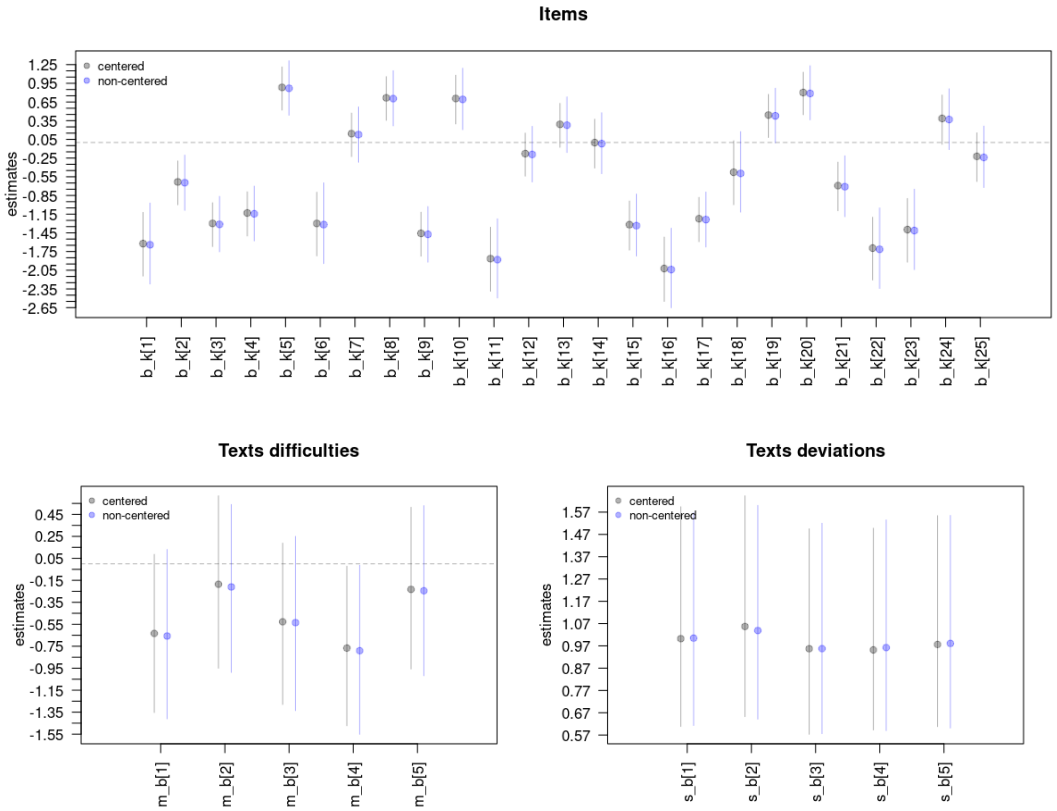

Figure A.52: Application's second-order latent variable model (SOLV). Centered and non-centered parametrization. Items, and texts difficulties, and texts deviations.

Appendix B

Code

B.1 Chapter 3: Bayesian estimation

B.1.1 To center or not to center

The devil's funnel, centered parametrization.

Stan

```
transformed data {
        int<lower=0> J;
        J = 1;
}
parameters {
        real theta[J];
        real v;
}
model {
        v ~ normal(0, 3);
        theta ~ normal(0, exp(v));
}
```

JAGS

```
model{
        v ~ dnorm(0,3)
        theta ~ dnorm(0, exp(v))
}
```

The devil's funnel, centered parametrization with priors.

Stan

```
transformed data {
        int<lower=0> J;
        J = 1;
}
parameters {
        real theta[J];
```

```
        real v;
}
model {
        v ~ normal(0, 1);
        theta ~ normal(0, exp(v));
}
```

JAGS

```
model{
        v ~ dnorm(0,1)
        theta ~ dnorm(0, exp(v))
}
```

The devil's funnel, non-centered parametrization.

Stan

```
transformed data {
        int<lower=0> J;
        J = 1;
}
parameters {
        real ztheta[J];
        real v;
}
transformed parameters{
        vector[J] theta;
        theta = exp(v) * to_vector( ztheta );
}
model {
        v ~ normal(0, 3);
        ztheta ~ normal(0, 1);
}
```

JAGS

```
model{
        v ~ dnorm(0,1)
        ztheta ~ dnorm(0,1)
        theta = v * ztheta
}
```

B.2 Chapter 4: Simulation study

This section provides the most important set of codes used throughout the current research. However, in case the reader wants to replicate the full simulation analysis, we provide the full set of codes in the "simulation" section of the accompanying github page: https://github.com/jriveraespejo/thesis/tree/master/simulation

B.2.1 Algorithm

Data generation

```
S = 10 # ten data sets
condition = expand_grid( J = c(100, 250, 500), load=0.95)
for(i in 1:nrow(condition)){
        for(s in 1:S){
                with(condition[i,],
                        data_generation( J=J, loads=rep(load, 3),
                                Ndata=s, seed=4587+s+i, # different seeds
                                file_dir=file.path(getwd(), 'data') ) )
        }
}
```

Data generation function

```
# function:
#     data_generation
# description:
#     To generate data based on different parameter settings.
#     Only two parameters are effectively controlled in the experimentation:
#     sample size (J), and the loading from the SOLV to the FOLV (loads).
# characteristics of the sample design:
#   - one instrument
#   - hierarchical measurement scales with FOLV and SOLV
#   - one evaluation time
#   - one sample of individuals
#   - testlet items, multiple items come from one text
#   - with covariates
#   - NO missingness
# arguments:
#     J = individual sample sizes
#     loads = loadings from SOLV to FOLV (it control correlation)
#     Ndata = defines the number of data generated
#     file_dir = path to save generated data
#     s_theta = sd to simulate SOLV and FOLV
#     s_text = sd in generating items from a specific text
#     D = number of dimensions (default 3)
#     K = number of items (default 25)
#     L = number of texts (default 5), it has to be a multiple of K
#     seed = seed used to generate the simulation (default 1)
#     prec = rounding in abilities and item parameters (default 3)
```

```
data_generation =function( J=100, loads=rep(0.95, 3), Ndata=1, file_dir,
        s_theta=0.5, s_text=0.5, D=3, K=25, L=5, seed=1, prec=3){

#_______________
# 1. generation
#_______________
set.seed(seed)


## 1.1. regression parameters
mom = expand_grid(a=loads[-3], b=loads[-1])
mom = mom[-3,]

betas = list( gender = c(0, 0.5),
        age = -0.02,
        edu = c(-0.5, 0.5, 0),
        exp = c(-0.5, 0, 0.35, 0.5),
        loads = loads, # loadings
        exp_corr = with(mom, a*b) ) # expected correlation


## 1.2 covariates
abilities = data.frame( IDind=1:J,
        gender = sample(c(1,2), size=J, replace=T),
        age = sample(30:65, size=J, replace=T),
        edu = sample(c(1,2,3), size=J, replace=T),
        exp = sample(c(1,2,3,4), size=J, replace=T),
        theta=rep(NA,J), theta1=rep(NA,J),
        theta2=rep(NA,J), theta3=rep(NA,J) )

# variable indices
SOLV_loc = which(names(abilities)=='theta')
FOLV_loc = which(names(abilities)=='theta1'):
        which(names(abilities)==paste0('theta', D))


## 1.3. abilities
# SOLV
m_theta = rep(NA, J)
for(j in 1:J){
        ageC = with(abilities, age[j] - min(age) + 1 )

        m_theta[j] = with(abilities,
                betas$gender[ gender[j] ] +
                betas$age * ageC +
                betas$edu[ edu[j] ] +
                betas$exp[ exp[j] ] )
}
abilities[, SOLV_loc] = round( rnorm(J, m_theta, s_theta), prec)
```

```
# FOLV
# independent after considering SOLV
s_mult = diag( rep(s_theta, D) )
m_mult = data.frame( theta1=rep(NA, J),
        theta2=rep(NA, J),
        theta3=rep(NA, J) )
for(j in 1:J){
        m_mult[j,] = betas$loads * abilities$theta[j]
        abilities[j, FOLV_loc] = round(
                mvrnorm(n=1, mu=unlist(m_mult[j,]), Sigma=s_mult), prec)
}


## 1.4 texts
texts = data.frame(IDtext=1:L, m_b=rep(NA, L), s_b=rep(NA, L))
texts$m_b = seq(-1.5, 1.5, length.out=L)
texts$s_b = rep(s_text, L)


## 1.5 items
items = data.frame(IDitem=1:K, IDtext=rep(NA,K),
        IDdim=rep(NA,K), b=rep(NA, K))
kl = K/L # items per text
for(k in 1:nrow(texts)){
        items$b[(1:kl) + (k-1)*kl] = with(texts,
                round( rnorm(kl, m_b[k], s_b[k]), prec ) )
        items$IDtext[(1:kl) + (k-1)*kl] = k
}


## 1.6 dimensions
items$IDdim = sample(1:3, K, replace=T)

#---------------
# 2. storage
#---------------

## 2.1 parameters
data_true = list(

        seed=seed,

        # indices
        J = J,
        D = D,
        K = K,
        L = L,

        # abilities
```

```
        betas = betas,
        abilities = abilities,

        # texts and items
        texts = texts,
        items = items)

        file_name = paste0('Parameters_J',J,'_l',loads[1],
                '_Ndata',Ndata,'.RData')
        save(data_true, file=file.path(file_dir, file_name) )



## 2.2 long format
data_eval = data.frame(

        # individual data
        IDind = rep(abilities$IDind, K),
        gender = rep(abilities$gender, K),
        age = rep(abilities$age, K),
        edu = rep(abilities$edu, K),
        exp = rep(abilities$exp, K),
        theta = rep(abilities$theta, K),
        theta1 = rep(abilities$theta1, K),
        theta2 = rep(abilities$theta2, K),
        theta3 = rep(abilities$theta3, K),

        # items
        IDitem = rep(items$IDitem, each=J),
        IDtext = rep(items$IDtext, each=J),
        IDdim = rep(items$IDdim, each=J),
        b = rep(items$b, each=J) )

# appropriate ability
data_eval$theta_jkld = NA
for(i in 1:nrow(data_eval) ){
        data_eval$theta_jkld[i] = data_eval[i, FOLV_loc[ data_eval$IDdim[i]
}

# linear predictor, probability, and outcome
data_eval$v_jkld = with(data_eval, theta_jkld - b)
data_eval$p_jkld = inv_logit( data_eval$v )
data_eval$y_jkld = rbinom( nrow(data_eval), 1, prob=data_eval$p)

file_name = paste0('LongFormat_J',J,'_l',loads[1],
        '_Ndata',Ndata,'.RData')
save(data_eval, file=file.path(file_dir, file_name) )


## 2.3 estimation data
```

```
data_post = list(

        # evaluation data
        N = nrow(data_eval),
        J = J,
        K = K,
        L = L,
        D = D,
        IDj = data_eval$IDind,
        IDk = data_eval$IDitem,
        IDl = data_eval$IDtext,
        IDd = data_eval$IDdim,
        GE = data_eval$gender,
        AG = data_eval$age,
        ED = data_eval$edu,
        XP = data_eval$exp,
        y = data_eval$y_jkld,

        # individual data
        IDind = abilities$IDind,
        G = abilities$gender,
        A = abilities$age,
        E = abilities$edu,
        X = abilities$exp,

        # items
        IDitem = items$IDitem,
        IDtext = items$IDtext,
        IDdim = items$IDdim )

file_name = paste0('ListFormat_J',J,'_l',loads[1],
        '_Ndata',Ndata,'.RData')
save(data_post, file=file.path(file_dir, file_name) )

}
```

B.2.2 Results

Models

FOLV CP

```
data{
        // evaluation
        int N;
        int J;
        int K;
        int L;
        int D;
        int IDj[N];
```

```
        int IDk[N];
        int IDl[N];
        int IDd[N];
        int GE[N];
        real AG[N];
        int ED[N];
        int XP[N];
        int y[N];

        // individuals
        int IDind[J];
        int G[J];
        real A[J];
        int E[J];
        int X[J];

        // items
        int IDitem[K];
        int IDtext[K];
        int IDdim[K];
}
parameters{
        // items
        real m_b[L];
        real<lower=0> s_b[L];
        real b_k[K];

        // betas
        real a;
        real b_G[2];
        real b_A;
        real b_E[3];
        real b_X[4];

        // abilities
        // sub-dimensions: lit, inf, ref
        corr_matrix[D] Rho_theta_sub;
        vector[D] theta_sub[J];
}
model{
        // declare
        vector[D] m_mult[J];
        real v;
        real p;

        // items
        m_b ~ normal(0, 1);
        s_b ~ exponential(2);
        for(k in 1:K){            // priors
                b_k[k] ~ normal( m_b[ IDtext[k] ], s_b[ IDtext[k] ]);
```

```
        }

        // abilities
        a ~ normal(0, 0.5);
        b_G ~ normal(0, 0.5);
        b_A ~ normal(0, 0.5);
        b_E ~ normal(0, 1);
        b_X ~ normal(0, 0.5);
        Rho_theta_sub ~ lkj_corr(2);
        for(j in 1:J){
                m_mult[j,] = rep_vector( a + b_G[ G[j] ] +
                        b_A * ( A[j] - min(A) ) +
                        b_E[ E[j] ] +
                        b_X[ X[j] ], D);
                // using the same predictor for the three latents
        }
        theta_sub ~ multi_normal( m_mult, Rho_theta_sub );

        // model
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                y[i] ~ bernoulli(p);
        }
}
//# generated quantities{
//#     vector[N] log_lik;
//#     real v;
//#     real p;
//#
//#     // likelihood
//#     for( i in 1:N ) {
//#       v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
//#       p = inv_logit(v);
//#       log_lik[i] = bernoulli_lpmf( y[i] | p);
//#     }
//# }
```

FOLV NCP

```
data{
        // evaluation
        int N;
        int J;
        int K;
        int L;
        int D;
        int IDj[N];
        int IDk[N];
        int IDl[N];
        int IDd[N];
```

```
        int GE[N];
        real AG[N];
        int ED[N];
        int XP[N];
        int y[N];

        // individuals
        int IDind[J];
        int G[J];
        real A[J];
        int E[J];
        int X[J];

        // items
        int IDitem[K];
        int IDtext[K];
        int IDdim[K];
}
parameters{
        // items
        real m_b[L];
        real<lower=0> s_b[L];
        real zb_k[K];

        // betas
        real a;
        real b_G[2];
        real b_A;
        real b_E[3];
        real b_X[4];

        // abilities
        // sub-dimensions: lit, inf, ref
        cholesky_factor_corr[D] L_Rho_theta_sub;
        matrix[D, J] ztheta_sub;
}
transformed parameters{
        real b_k[K];
        matrix[D, D] Rho_theta_sub;
        matrix[J, D] m_mult;
        matrix[J, D] theta_sub;

        // items
        for(k in 1:K){
                b_k[k] = m_b[ IDtext[k] ] + s_b[ IDtext[k] ] * zb_k[k];
        }

        // abilities
        Rho_theta_sub = multiply_lower_tri_self_transpose(L_Rho_theta_sub);
        for(j in 1:J){
```

```
                m_mult[j,] = rep_row_vector( a + b_G[ G[j] ] +
                        b_A * (A[j] - min(A) ) +
                        b_E[ E[j] ] +
                        b_X[ X[j] ], D);
        }
        theta_sub = (L_Rho_theta_sub * ztheta_sub)';
        theta_sub = theta_sub + m_mult;
}
model{
        // declare
        real v;
        real p;

        // items
        m_b ~ normal(0, 1);
        s_b ~ exponential(2);
        zb_k ~ normal(0, 1);

        // abilities
        a ~ normal(0, 0.5);
        b_G ~ normal(0, 0.5);
        b_A ~ normal(0, 0.5);
        b_E ~ normal(0, 1);
        b_X ~ normal(0, 0.5);
        L_Rho_theta_sub ~ lkj_corr_cholesky(2);
        to_vector(ztheta_sub) ~ normal(0,1);

        // model
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                y[i] ~ bernoulli(p);
        }
}
//# generated quantities{
//#     vector[N] log_lik;
//#     real v;
//#     real p;
//#
//#     // likelihood
//#     for( i in 1:N ) {
//#       v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
//#       p = inv_logit(v);
//#       log_lik[i] = bernoulli_lpmf( y[i] | p);
//#     }
//# }
```

SOLV CP

```
data{
        // evaluation
```

```
        int N;
        int J;
        int K;
        int L;
        int D;
        int IDj[N];
        int IDk[N];
        int IDl[N];
        int IDd[N];
        int GE[N];
        real AG[N];
        int ED[N];
        int XP[N];
        int y[N];

        // individuals
        int IDind[J];
        int G[J];
        real A[J];
        int E[J];
        int X[J];

        // items
        int IDitem[K];
        int IDtext[K];
        int IDdim[K];
}
parameters{
        // items
        real m_b[L];
        real<lower=0> s_b[L];
        real b_k[K];

        // betas
        real a;
        real b_G[2];
        real b_A;
        real b_E[3];
        real b_X[4];

        // abilities
        vector[J] theta;              // reading comprehension
        real<lower=0> loads[D];       // loadings
        corr_matrix[D] Rho_theta_sub; // sub-dimensions: lit, inf, ref
        vector[D] theta_sub[J];
}
model{
        // declare
        vector[J] m_theta;
        vector[D] m_mult[J];
```

```
        real v;
        real p;

        // items
        m_b ~ normal(0, 1);
        s_b ~ exponential(2);
        for(k in 1:K){              // priors
                b_k[k] ~ normal( m_b[ IDtext[k] ], s_b[ IDtext[k] ]);
        }

        // SOLV
        a ~ normal(0, 0.5);
        b_G ~ normal(0, 0.5);
        b_A ~ normal(0, 0.5);
        b_E ~ normal(0, 1);
        b_X ~ normal(0, 0.5);
        for(j in 1:J){
                m_theta[j] = a + b_G[ G[j] ] +
                        b_A * ( A[j] - min(A) ) +
                        b_E[ E[j] ] +
                        b_X[ X[j] ];
        }
        theta ~ normal(m_theta, 1);
        // setting the scale at this level

        // FOLV
        loads ~ lognormal(0, 0.5);
        for(j in 1:J){
                m_mult[j,] = [ loads[1]*theta[j],
                        loads[2]*theta[j],
                        loads[3]*theta[j] ]';
        }
        Rho_theta_sub ~ lkj_corr(2);
        theta_sub ~ multi_normal( m_mult, Rho_theta_sub );
        // also setting scale here

        // model
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                y[i] ~ bernoulli(p);
        }
}
//# generated quantities{
//#     vector[N] log_lik;
//#     real v;
//#     real p;
//#
//#     // likelihood
//#     for( i in 1:N ) {
```

```
//#       v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
//#       p = inv_logit(v);
//#       log_lik[i] = bernoulli_lpmf( y[i] | p);
//#     }
//# }
```

SOLV NCP

```
data{
        // evaluation
        int N;
        int J;
        int K;
        int L;
        int D;
        int IDj[N];
        int IDk[N];
        int IDl[N];
        int IDd[N];
        int GE[N];
        real AG[N];
        int ED[N];
        int XP[N];
        int y[N];

        // individuals
        int IDind[J];
        int G[J];
        real A[J];
        int E[J];
        int X[J];

        // items
        int IDitem[K];
        int IDtext[K];
        int IDdim[K];
}
parameters{
        // items
        real m_b[L];
        real<lower=0> s_b[L];
        real zb_k[K];

        // betas
        real a;
        real b_G[2];
        real b_A;
        real b_E[3];
        real b_X[4];

        // abilities
```

```
        vector[J] ztheta;              // reading comprehension
        real<lower=0> loads[D];        // loadings
        cholesky_factor_corr[D] L_Rho_theta_sub; // sub-dimensions
        matrix[D, J] ztheta_sub;
}
transformed parameters{
        real b_k[K];
        vector[J] m_theta;
        vector[J] theta;              // reading comprehension
        matrix[D, D] Rho_theta_sub;
        matrix[J, D] m_mult;
        matrix[J, D] theta_sub;

        // items
        for(k in 1:K){
                b_k[k] = m_b[ IDtext[k] ] + s_b[ IDtext[k] ] * zb_k[k];
        }

        // SOLV
        for(j in 1:J){
                m_theta[j] = a + b_G[ G[j] ] +
                b_A * ( A[j] - min(A) ) +
                b_E[ E[j] ] +
                b_X[ X[j] ];
        }
        theta = m_theta + 1 * ztheta;
        // setting the scale at this level

        // FOLV
        Rho_theta_sub = multiply_lower_tri_self_transpose(L_Rho_theta_sub);
        for(j in 1:J){
                m_mult[j,] = to_row_vector( [ loads[1]*theta[j],
                loads[2]*theta[j],
                loads[3]*theta[j] ] );
        }
        theta_sub = m_mult + (L_Rho_theta_sub * ztheta_sub)';
        // also setting scale here
}
model{
        // declare
        real v;
        real p;

        // items
        m_b ~ normal(0, 1);
        s_b ~ exponential(2);
        zb_k ~ normal(0, 1);

        // abilities
        a ~ normal(0, 0.5);
```

```
        b_G ~ normal(0, 0.5);
        b_A ~ normal(0, 0.5);
        b_E ~ normal(0, 1);
        b_X ~ normal(0, 0.5);
        ztheta ~ normal(0,1);
        L_Rho_theta_sub ~ lkj_corr_cholesky(2);
        loads ~ lognormal(0, 0.5);
        to_vector(ztheta_sub) ~ normal(0,1);

        // model
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                y[i] ~ bernoulli(p);
        }
}
//# generated quantities{
//#     vector[N] log_lik;
//#     real v;
//#     real p;
//#
//#     // likelihood
//#     for( i in 1:N ) {
//#       v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
//#       p = inv_logit(v);
//#       log_lik[i] = bernoulli_lpmf( y[i] | p);
//#     }
//# }
```

Prior predictive investigation

Prior predictive

```
model_path = file.path(getwd(), 'models_prior')
model_list = list.files( model_path )
model_list = model_list[ str_detect(model_list, '.stan') ]

run_prior( models = model_list,
model_path = model_path,
model_out = file.path(getwd(), 'chains_prior'),
data_path = file.path(getwd(), 'data') )
```

Prior predictive function

```
# function:
#     run_prior
# description:
#     To run each prior simulation for each model
# arguments:
#     models = list of model to run located in model_path
#     model_path = path where all models are located
```

```
#     model_out = path where chains have to be saved
#     data_path = path where data per condition is located

run_prior = function(models, model_path, model_out, data_path){

# data list
data_list = list.files( data_path )
data_list = data_list[str_detect(data_list, 'ListFormat')]

for(j in 1:length(models) ){

        # compile model
        set_cmdstan_path('~/cmdstan')
        mod = cmdstan_model( file.path(model_path, models[j] ) )

        # generate base name
        base_nam = str_replace( data_list[1], 'ListFormat_',
        str_replace( models[j], '.stan', '_') )
        base_nam = str_replace( base_nam, '.RData', '' )

        # load data
        load( file.path(data_path, data_list[1] ) )

        # run model
        mod$sample(data = data_post,
                output_dir = model_out,
                output_basename = base_nam,
                chains=1, parallel_chains=1, adapt_delta=0.99)
        }
}
```

Posterior predictive

Posterior predictive

```
model_path = file.path(getwd(), 'models_post')
model_list = list.files( model_path )
model_list = model_list[ str_detect(model_list, '.stan') ]
model_list = model_list[c(1:2,5:6,3:4)]

run_post( models = model_list,
model_path = model_path,
model_out = file.path(getwd(), 'chains_post'),
data_path = file.path(getwd(), 'data') )
```

Posterior predictive function

```
# function:
#     run_post
# description:
#     To run each model for each data in all conditions
```

```
# arguments:
#     models = list of model to run located in model_path
#     model_path = path where all models are located
#     model_out = path where chains have to be saved
#     data_path = path where data per condition is located

run_post = function(models, model_path, model_out, data_path){

# data list
data_list = list.files( data_path )
data_list = data_list[str_detect(data_list, 'ListFormat')]

for(j in 1:length(models) ){

        # compile model
        set_cmdstan_path('~/cmdstan')
        mod = cmdstan_model( file.path(model_path, models[j] ) )

        for(i in 1:length(data_list) ){

                # generate base name
                base_nam = str_replace( data_list[i], 'ListFormat_',
                str_replace(models[j], '.stan', '_') )
                base_nam = str_replace( base_nam, '.RData', '' )

                # load data
                load( file.path(data_path, data_list[i] ) )

                # run model
                t0 = proc.time()
                mod$sample(data = data_post,
                        output_dir = model_out,
                        output_basename = base_nam,
                        chains=3, parallel_chains=3, adapt_delta=0.99)
                t1 = proc.time()

                # saving time
                start = str_locate(base_nam, 'J')[2]
                m = str_sub( base_nam, start=1, end=(start-2) )
                J = str_sub( base_nam, start=start+1, end=(start+3) )

                start = str_locate(base_nam, 'l')[2]
                l = str_sub( base_nam, start=start+1, end=(start+3) )

                start = str_locate(base_nam, 'Ndata')[2]
                S = str_sub( base_nam, start=start+1, end=(start+2) )

                elapsed = (t1-t0)

                if(j==1 & i==1){
```

```
                        time_elap = data.frame(
                                Model=m,
                                J=J,
                                load=l,
                                data=S,
                                time=unlist(elapsed['elapsed']) )
                } else{
                        time_elap = rbind(time_elap,
                                c(m, J, l, S,
                                        unlist(elapsed['elapsed']) ) )
                }

                # save time elapsed (saved at each iteration)
                write.csv( time_elap, row.names=F,
                file=file.path(model_out, 'time_elapsed.csv' ) )

        }
}

}
```

B.3 Chapter 5: Application

B.3.1 Models

FOLV CP

```
data{
        // evaluation
        int N;
        int J;
        int K;
        int L;
        int D;
        int IDj[N];
        int IDk[N];
        int IDl[N];
        int IDd[N];
        int G[N];
        real A[N];
        int E[N];
        int S[N];
        int Xpu[N];
        int Xpr[N];
        int Di[N];
        int y[N];

        // individuals
        int IDind[J];
        int GE[J];
        real AG[J];
        int ED[J];
        int SP[J];
        int XPpu[J];
        int XPpr[J];
        int DI[J];

        // items
        int IDitem[K];
        int IDtext[K];
        int IDdim[K];
}
transformed data {
        real min_AG;
        min_AG = min(AG);
}
parameters{
        // items
        real m_b[L];
        real<lower=0> s_b[L];
        real b_k[K];
```

```
        // betas
        real a;
        real b_G[2];
        real b_A;
        real b_E[3];
        real b_S[3];
        real b_Xpu[4];
        real b_Xpr[4];
        real b_Di[4];

        // abilities
        corr_matrix[D] Rho_theta_sub; // sub-dimensions: lit, inf, ref
        vector[D] theta_sub[J];
}
model{
        // declare
        vector[D] m_mult[J];
        real v;
        real p;

        // items
        m_b ~ normal(0, 1);
        s_b ~ exponential(2);
        for(k in 1:K){              // priors
                b_k[k] ~ normal( m_b[ IDtext[k] ], s_b[ IDtext[k] ]);
        }

        // abilities
        a ~ normal(0, 0.5);
        b_G ~ normal(0, 0.5);
        b_A~ normal(0, 0.5);
        b_E ~ normal(0, 1);
        b_S ~ normal(0, 0.5);
        b_Xpu ~ normal(0, 0.5);
        b_Xpr ~ normal(0, 0.5);
        b_Di ~ normal(0, 0.5);
        Rho_theta_sub ~ lkj_corr(2);
        for(j in 1:J){
                m_mult[j,] = rep_vector( a + b_G[ GE[j] ] +
                        b_A*(AG[j] - min_AG) +
                        b_E[ ED[j] ] + b_S[ SP[j] ] + b_Di[ DI[j] ] +
                        b_Xpu[ XPpu[j] ] + b_Xpr[ XPpr[j] ] , D);
                // using the same predictor for the three latents
        }
        theta_sub ~ multi_normal( m_mult, Rho_theta_sub );

        // model
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
```

```
                y[i] ~ bernoulli(p);
        }
}
generated quantities{
        vector[N] log_lik;
        real v;
        real p;

        // likelihood
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                log_lik[i] = bernoulli_lpmf( y[i] | p);
        }
}
```

FOLV NCP

```
data{
        // evaluation
        int N;
        int J;
        int K;
        int L;
        int D;
        int IDj[N];
        int IDk[N];
        int IDl[N];
        int IDd[N];
        int G[N];
        real A[N];
        int E[N];
        int S[N];
        int Xpu[N];
        int Xpr[N];
        int Di[N];
        int y[N];

        // individuals
        int IDind[J];
        int GE[J];
        real AG[J];
        int ED[J];
        int SP[J];
        int XPpu[J];
        int XPpr[J];
        int DI[J];

        // items
        int IDitem[K];
        int IDtext[K];
```

```
        int IDdim[K];
}
transformed data {
        real min_AG;
        min_AG = min(AG);
}
parameters{
        // items
        real m_b[L];
        real<lower=0> s_b[L];
        real zb_k[K];

        // betas
        real a;
        real b_G[2];
        real b_A;
        real b_E[3];
        real b_S[3];
        real b_Xpu[4];
        real b_Xpr[4];
        real b_Di[4];

        // abilities
        cholesky_factor_corr[D] L_Rho_theta_sub;
        matrix[D, J] ztheta_sub;
}
transformed parameters{
        real b_k[K];
        matrix[D, D] Rho_theta_sub;
        matrix[J, D] m_mult;
        matrix[J, D] theta_sub;

        // items
        for(k in 1:K){
                b_k[k] = m_b[ IDtext[k] ] + s_b[ IDtext[k] ] * zb_k[k];
        }

        // abilities
        Rho_theta_sub = multiply_lower_tri_self_transpose(L_Rho_theta_sub);
        for(j in 1:J){
                m_mult[j,] = rep_row_vector( a + b_G[ GE[j] ] +
                        b_A*(AG[j] - min_AG) +
                        b_E[ ED[j] ] + b_S[ SP[j] ] + b_Di[ DI[j] ] +
                        b_Xpu[ XPpu[j] ] + b_Xpr[ XPpr[j] ] , D);
        }
        theta_sub = m_mult + (L_Rho_theta_sub * ztheta_sub)';
}
model{
        // declare
        real v;
```

```
        real p;

        // items
        m_b ~ normal(0, 1);
        s_b ~ exponential(2);
        zb_k ~ normal(0, 1);

        // abilities
        a ~ normal(0, 0.5);
        b_G ~ normal(0, 0.5);
        b_A~ normal(0, 0.5);
        b_E ~ normal(0, 1);
        b_S ~ normal(0, 0.5);
        b_Xpu ~ normal(0, 0.5);
        b_Xpr ~ normal(0, 0.5);
        b_Di ~ normal(0, 0.5);
        L_Rho_theta_sub ~ lkj_corr_cholesky(2);
        to_vector(ztheta_sub) ~ normal(0,1);

        // model
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                y[i] ~ bernoulli(p);
        }
}
generated quantities{
        vector[N] log_lik;
        real v;
        real p;

        // likelihood
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                log_lik[i] = bernoulli_lpmf( y[i] | p);
        }
}
```

SOLV CP

```
data{
        // evaluation
        int N;
        int J;
        int K;
        int L;
        int D;
        int IDj[N];
        int IDk[N];
        int IDl[N];
```

```
        int IDd[N];
        int G[N];
        real A[N];
        int E[N];
        int S[N];
        int Xpu[N];
        int Xpr[N];
        int Di[N];
        int y[N];

        // individuals
        int IDind[J];
        int GE[J];
        real AG[J];
        int ED[J];
        int SP[J];
        int XPpu[J];
        int XPpr[J];
        int DI[J];

        // items
        int IDitem[K];
        int IDtext[K];
        int IDdim[K];
}
transformed data {
        real min_AG;
        min_AG = min(AG);
}
parameters{
        // items
        real m_b[L];
        real<lower=0> s_b[L];
        real b_k[K];

        // betas
        real a;
        real b_G[2];
        real b_A;
        real b_E[3];
        real b_S[3];
        real b_Xpu[4];
        real b_Xpr[4];
        real b_Di[4];

        // abilities
        vector[J] theta;                // reading comprehension
        real<lower=0> loads[D];         // loadings
        corr_matrix[D] Rho_theta_sub; // sub-dimensions: lit, inf, ref
        vector[D] theta_sub[J];
```

```
}
model{
        // declare
        vector[J] m_theta;
        vector[D] m_mult[J];
        real v;
        real p;

        // items
        m_b ~ normal(0, 1);
        s_b ~ exponential(2);
        for(k in 1:K){              // priors
                b_k[k] ~ normal( m_b[ IDtext[k] ], s_b[ IDtext[k] ]);
        }

        // SOLV
        a ~ normal(0, 0.5);
        b_G ~ normal(0, 0.5);
        b_A~ normal(0, 0.5);
        b_E ~ normal(0, 1);
        b_S ~ normal(0, 0.5);
        b_Xpu ~ normal(0, 0.5);
        b_Xpr ~ normal(0, 0.5);
        b_Di ~ normal(0, 0.5);
        for(j in 1:J){
                m_theta[j] = a + b_G[ GE[j] ] + b_A*(AG[j] - min_AG) +
                        b_E[ ED[j] ] + b_S[ SP[j] ] + b_Di[ DI[j] ] +
                        b_Xpu[ XPpu[j] ] + b_Xpr[ XPpr[j] ];
        }
        theta ~ normal(m_theta, 1);
        // setting the scale at this level

        // FOLV
        loads ~ lognormal(0, 0.5);
        for(j in 1:J){
                m_mult[j,] = [ loads[1]*theta[j],
                                loads[2]*theta[j],
                                loads[3]*theta[j] ]';
        }
        Rho_theta_sub ~ lkj_corr(2);
        theta_sub ~ multi_normal( m_mult, Rho_theta_sub );
        // also setting scale here

        // model
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                y[i] ~ bernoulli(p);
        }
}
```

```
generated quantities{
        vector[N] log_lik;
        real v;
        real p;

        // likelihood
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                log_lik[i] = bernoulli_lpmf( y[i] | p);
        }
}
```

SOLV NCP

```
data{
        // evaluation
        int N;
        int J;
        int K;
        int L;
        int D;
        int IDj[N];
        int IDk[N];
        int IDl[N];
        int IDd[N];
        int G[N];
        real A[N];
        int E[N];
        int S[N];
        int Xpu[N];
        int Xpr[N];
        int Di[N];
        int y[N];

        // individuals
        int IDind[J];
        int GE[J];
        real AG[J];
        int ED[J];
        int SP[J];
        int XPpu[J];
        int XPpr[J];
        int DI[J];

        // items
        int IDitem[K];
        int IDtext[K];
        int IDdim[K];
}
transformed data {
```

```
        real min_AG;
        min_AG = min(AG);
}
parameters{
        // items
        real m_b[L];
        real<lower=0> s_b[L];
        real zb_k[K];

        // betas
        real a;
        real b_G[2];
        real b_A;
        real b_E[3];
        real b_S[3];
        real b_Xpu[4];
        real b_Xpr[4];
        real b_Di[4];

        // abilities
        vector[J] ztheta;              // reading comprehension
        real<lower=0> loads[D];        // loadings
        cholesky_factor_corr[D] L_Rho_theta_sub;
        matrix[D, J] ztheta_sub;
}
transformed parameters{
        real b_k[K];
        vector[J] m_theta;
        vector[J] theta;              // reading comprehension
        matrix[D, D] Rho_theta_sub;
        matrix[J, D] m_mult;
        matrix[J, D] theta_sub;

        // items
        for(k in 1:K){
                b_k[k] = m_b[ IDtext[k] ] + s_b[ IDtext[k] ] * zb_k[k];
        }

        // SOLV
        for(j in 1:J){
                m_theta[j] = a + b_G[ GE[j] ] + b_A*(AG[j] - min_AG) +
                        b_E[ ED[j] ] + b_S[ SP[j] ] + b_Di[ DI[j] ] +
                        b_Xpu[ XPpu[j] ] + b_Xpr[ XPpr[j] ];
        }
        theta = m_theta + 1 * ztheta;
        // setting the scale at this level

        // FOLV
        Rho_theta_sub = multiply_lower_tri_self_transpose(L_Rho_theta_sub);
        for(j in 1:J){
```

```
                m_mult[j,] = to_row_vector(
                                [ loads[1]*theta[j],
                                loads[2]*theta[j],
                                loads[3]*theta[j] ] );
        }
        theta_sub = m_mult + (L_Rho_theta_sub * ztheta_sub)';
        // also setting scale here
}
model{
        // declare
        real v;
        real p;

        // items
        m_b ~ normal(0, 1);
        s_b ~ exponential(2);
        zb_k ~ normal(0, 1);

        // abilities
        a ~ normal(0, 0.5);
        b_G ~ normal(0, 0.5);
        b_A~ normal(0, 0.5);
        b_E ~ normal(0, 1);
        b_S ~ normal(0, 0.5);
        b_Xpu ~ normal(0, 0.5);
        b_Xpr ~ normal(0, 0.5);
        b_Di ~ normal(0, 0.5);
        ztheta ~ normal(0,1);
        L_Rho_theta_sub ~ lkj_corr_cholesky(2);
        loads ~ lognormal(0, 0.5);
        to_vector(ztheta_sub) ~ normal(0,1);

        // model
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                y[i] ~ bernoulli(p);
        }
}
generated quantities{
        vector[N] log_lik;
        real v;
        real p;

        // likelihood
        for( i in 1:N ) {
                v = theta_sub[ IDj[i], IDd[i] ] - b_k[ IDk[i] ];
                p = inv_logit(v);
                log_lik[i] = bernoulli_lpmf( y[i] | p);
        }
```

```
}
```

Bibliography

[1] Azevedo, C. [2003]. *Métodos de estimação na teoria de resposta ao item*, Master's thesis, Universidade de São Paulo (USP).
url: https://teses.usp.br/teses/disponiveis/45/45133/tde-05102004-163906/pt-br.php.

[2] Baker, F. [1998]. An investigation of the item parameter recovery characteristics of a gibbs sampling procedure, *Applied Psychological Measurement* **22**(22): 153–169.
doi: https://doi.org/10.1177/01466216980222005.

[3] Baker, F. [2001]. The basic of item response theory, *Technical report*, ERIC Clearinghouse on Assessment and Evaluation.

[4] Baker, F. and Kim, S. [1992]. *Item Response Theory: Parameter Estimation Techniques*, Statistics for the Social and Behavioral Sciences, CRC Press, Taylor and Francis Group.
doi: https://www.doi.org/10.1201/9781482276725.

[5] Beaujean, A. [2014]. *Latent Variable Modeling Using R. A Step-by-Step Guide.*, Routledge.

[6] Betancourt, M. and Girolami, M. [2012]. Hamiltonian monte carlo for hierarchical models.
url: arxiv.org/abs/1312.0906v1.

[7] Bock, R. [1972]. Estimating item parameters and latent ability when responses are scored in two or more nominal categories, *Psychometrika* **37**(1).
doi: https://doi.org/10.1007/BF02291411.

[8] Bradlow, E., Wainer, H. and Wang, X. [1999]. A bayesian random effects model for testlets, *Psychometrika* **64**(2): 153–168.
doi: https://doi.org/10.1007/BF02294533.

[9] Chen, W. and Thissen, D. [1997]. Local dependence indexes for item pairs using item response theory, *Journal of Educational and Behavioral Statistics* **22**(3): 265–289.
doi: https://doi.org/10.3102/10769986022003265.

[10] Depaoli, S. [2021]. *Bayesian Structural Equation Modeling*, Methodology in the Social Sciences, The Guilford Press.

[11] Duane, S., Kennedy, A., Pendleton, B. and Roweth, D. [1987]. Hybrid monte carlo, *Physics Letters B* **195**(2): 216–222.

doi: https://doi.org/10.1016/0370-2693(87)91197-X.
url: https://www.sciencedirect.com/science/article/pii/037026938791197X.

[12] Edwards, J. and Bagozzi, R. [2000]. On the nature and direction of relationships between constructs and measures, *Psychological Methods* **5**(2): 155–174.
doi: https://www.doi.org/10.1037/1082-989X.5.2.155.

[13] Flores, S. [2012]. *Modelos testlet logísticos y logísticos de exponente positivo para pruebas de compresión de textos*, Master's thesis, Pontificia Universidad Católica del Perú.

[14] Fox, J. [2010]. *Bayesian Item Response Modeling, Theory and Applications*, Statistics for Social and Behavioral Sciences, fienberg, s. and van der linden, w. edn, Springer Science+Business Media, LLC.

[15] Fujimoto, K. [2018a]. The bayesian multilevel trifactor item response theory model, *Educational and Psychological Measurement* **79**(3): 462–494.
doi: https://doi.org/10.1177/0013164418806694.

[16] Fujimoto, K. [2018b]. A general bayesian multilevel multidimensional irt model for locally dependent data, *Br J Math Stat Psychol* **71**(3): 536–560.
doi: https://doi.org/10.1111/bmsp.12133.

[17] Fujimoto, K. [2020]. A more flexible bayesian multilevel bifactor item response theory model, *Journal of Educational Measurement* **57**(2): 255–285.
doi: https://doi.org/10.1111/jedm.12249.

[18] Gelfand, A., Sahu, S. and Carlin, B. [1995]. Efficient parametrisations for normal linear mixed models, *Biometrika* **82**(3): 479–488.
doi: https://doi.org/10.1093/biomet/82.3.479.

[19] Gelfand, A., Sahu, S. and Carlin, B. [1996]. Efficient parameterizations for generalised linear models (with discussion), *in* J. Bernardo, J. Berger, A. Dawid and a. Smith (eds), *Bayesian Statistics*, Vol. 5, pp. 165–180.

[20] Gelman, A. [1996]. Bayesian model building by pure thought: some principles and examples, *Statistica Sinica* **6**(1): 215–232.
url: https://www.jstor.org/stable/24306008.

[21] Gelman, A., Bois, F. and Jiang, J. [1996]. Physiological pharmacokinetic analysis using population modeling and informative prior distributions, *Journal of the American Statistical Association* **91**(436): 1400–1412.
doi: https://doi.org/10.2307/2291566.
url: https://www.jstor.org/stable/2291566.

[22] Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehtari, A. and Rubin, D. [2014]. *Bayesian Data Analysis*, Texts in Statistical Science, third edn, Chapman and Hall/CRC.

[23] Gelman, A. and Rubin, D. [1996]. Markov chain monte carlo methods in biostatistics, *Statistical Methods in Medical Research* **5**(4): 339–355.
doi: https://doi.org/10.1177/096228029600500402.

[24] Geman, S. and Geman, D. [1984]. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-6**(6): 721–741.
doi: https://doi.org/10.1109/TPAMI.1984.4767596.

[25] Gorinova, M., Moore, D. and Hoffman, M. [2019]. Automatic reparameterisation of probabilistic programs.
url: https://arxiv.org/abs/1906.03028.

[26] Hambleton, R. and Swaminathan, H. [1991]. *Item Response Theory*, Evaluation in Education and Human Services series, Springer Science+Business Media, LLC.

[27] Hambleton, R., Swaminathan, H. and Rogers, H. [1991]. *Fundamentals of Item Response Theory*, SAGE Publications Inc.

[28] Hastings, W. K. [1970]. Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* **57**(1): 97–109.
doi: https://doi.org/10.1093/biomet/57.1.97.
url: https://academic.oup.com/biomet/article-pdf/57/1/97/23940249/57-1-97.pdf.

[29] Hernán, M. and Robins, J. [2020]. *Causal Inference: What If*, 1 edn, Chapman and Hall/CRC.
url: https://www.hsph.harvard.edu/miguel-hernan/causal-inference-book/.

[30] Hsieh, M., Proctor, T., Hou, J. and Teo, K. [2010]. A comparison of bayesian mcmc and marginal maximum likehood methods in estimating the item parameters for the 2pl irt model, *International Journal of Innovative Management, Information and Production* **1**(1): 81–89.
url: http://ismeip.org/IJIMIP/contents/imip1011/10IN15T.pdf.

[31] Jaynes, E. [1985]. Highly informative priors, *Bayesian Statistics* **2**: 329–360.

[32] Jiao, H., Kamata, A., Wang, S. and Jin, Y. [2012]. A multilevel testlet model for dual local dependence, *Journal of Educational Measurement* **49**(1): 82–100.
doi: https://doi.org/10.1111/j.1745-3984.2011.00161.x.

[33] Keane, M. [1992]. A note on identification in the multinomial probit model, *Journal of Business and Economic Statistics* **10**(2): 193–200.
doi: https://doi.org/10.2307/1391677.
url: https://www.jstor.org/stable/1391677.

[34] Kieftenbeld, V. and Natesan, P. [2012]. Recovery of graded response model parameters: A comparison of marginal maximum likehood and markov chain monte carlo estimation, *Applied Psychological Measurement* **36**(5): 399–419.
doi: https://www.doi.org/10.1177/0146621612446170.

[35] Kim, S. and Cohen, A. [1999]. Accuracy of parameter estimation in gibbs sampling under the two-parameter logistic model, *Annual Meeting of the American Educational Research Association*, American Educational Research Association.
url: https://eric.ed.gov/?id=ED430012.

[36] Kline, R. [2012]. Assumptions in structural equation modeling, *in* R. Hoyle (ed.), *Handbook of Structural Equation Modeling*, The Guilford Press, chapter 7, pp. 111–125.

[37] Kullback, S., . L. R. [1951]. On information and sufficiency, *The Annals of Mathematical Statistics* **22**(1): 79–86.
url: http://www.jstor.org/stable/2236703.

[38] Lewandowski, D., Kurowicka, D. and Joe, H. [2009]. Generating random correlation matrices based on vines and extended onion method, *Journal of Multivariate Analysis* **100**(9): 1989–2001.
doi: https://doi.org/10.1016/j.jmva.2009.04.008.

[39] Linacre, J. [2021]. *Winsteps® (Version 5.1.0) [Computer Software]*, Portland, Oregon.
url: https://www.winsteps.com/.

[40] Lord, F. and Novik, M. [2008]. *Statistical Theories of Mental Test Scores*, Information Age Publishing.

[41] Lunn, D., Spiegelhalter, D., Thomas, A. and Best, N. [2009]. The bugs project: Evolution, critique and future directions, *Statistics in Medicine* **28**(25): 3049–3067.

[42] Lunn, D., Thomas, A., Best, N. and Spiegelhalter, D. [2000]. Winbugs - a bayesian modelling framework: Concepts, structure, and extensibility, *Statistics and Computing* (10): 325–337.
doi: https://www.doi.org/10.1023/A:1008929526011.

[43] Martin, J. and McDonald, R. [1975]. Bayesian estimation in unrestricted factor analysis: A treatment for heywood cases, *Psychometrika* (40): 505–517.
doi: https://doi.org/10.1007/BF02291552.

[44] McCullagh, P. and Nelder, J. [1989]. *Generalized Linear Models*, Monographs on Statistics Applied Probability, Chapman Hall/CRC Press.

[45] McElreath, R. [2020]. *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*, Texts in Statistical Science, 2 edn, Chapman and Hall/CRC.
doi: https://doi.org/10.1201/9780429029608.

[46] Metropolis, N., Rosenbluth, A., Rosenbluth, M. and Teller, A. [1953]. Equation of state calculations by fast computing machines, *The Journal of Chemical Physics* **21**: 1087–1092.
doi: https://doi.org/10.1063/1.1699114.

[47] Muller, P. [1991]. A generic approach to posterior integration and gibbs sampling, *Technical Report 91-09*, Department of Statistics, Purdue University.
url: https://www.stat.purdue.edu/docs/research/tech-reports/1991/tr91-09.pdf.

[48] Muthén, L. and Muthén, B. [1998-2011]. *Mplus User's Guide*, CA: Muthén Muthén.

[49] Neal, R. [2011]. Mcmc using hamiltonian dynamics, *in* S. Brooks, A. Gelman, G. Jones and X. Meng (eds), *Handbook of Markov Chain Monte Carlo*, Chapman Hall/CRC Press, chapter 5, pp. 113–162.

[50] Nelder, J. and Wedderburn, W. [1972]. Generalized linear models, *Royal Statistical Society* **135**(3): 370–384.
doi: https://doi.org/10.2307/2344614.
url: https://www.jstor.org/stable/2344614.

[51] Papaspiliopoulos, O., Roberts, G. and Skold, M. [2003]. Non-centered parameterisations for hierarchical models and data augmentation, *Bayesian Statistics* **7**: 307–326.
url: http://econ.upf.edu/ omiros/papers/val7.pdf.

[52] Papaspiliopoulos, O., Roberts, G. and Skold, M. [2007]. A general framework for the parametrization of hierarchical models, *Statistical Science* **22**(1): 59–73.
doi: https://www.doi.org/10.1214/088342307000000014.

[53] Patz, R. J. and 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.

[54] Plummer, M. [2003]. Jags: A program for analysis of bayesian graphical models using gibbs sampling.

[55] R Core Team [2015]. *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria.
url: http://www.R-project.org/.

[56] Rabe-Hesketh, S., Skrondal, A. and Pickles, A. [2004a]. Generalized multilevel structural equation modeling, *Psychometrika* **69**(2): 167–190.
doi: https://www.doi.org/10.1007/BF02295939.

[57] Rabe-Hesketh, S., Skrondal, A. and Pickles, A. [2004b]. *GLLAMM Manual*, UC Berkeley Division of Biostatistics.
url: http://www.biostat.jhsph.edu/ fdominic/teaching/bio656/software-gllamm.manual.pdf.

[58] Rabe-Hesketh, S., Skrondal, A. and Pickles, A. [2004c]. Maximum likelihood estimation of limited and discrete dependent variable models with nested random effects, *Journal of Econometrics* **128**(2): 301–323.
doi: https://www.doi.org/10.1016/j.jeconom.2004.08.017.
url: http://www.sciencedirect.com/science/article/pii/S0304407604001599.

[59] Rabe-Hesketh, S., Skrondal, A. and Zheng, X. [2012]. Multilevel structural equation modeling, *in* R. Hoyle (ed.), *Handbook of Structural Equation Modeling*, The Guilford Press, chapter 30, pp. 512–531.

[60] Rasch, G. [1980]. *Probabilistic Models for Some Intelligence and Attainment Tests*, University of Chicago Press.

[61] Raudenbush, S. and Bryk, A. [2002]. *Hierarchical linear models: Applications and data analysis methods (Vol. 1)*, Advanced Quantitative Techniques in the Social Sciences, SAGE Publications Inc.

[62] Reckase, M. [2009]. *Multidimensional Item Response Theory*, Statistics for Social and Behavioral Sciences, Springer Science+Business Media, LLC.

[63] Rivera, J. [2019]. *El modelo de respuesta nominal: Aplicación a datos educacionales*, Master's thesis, Pontificia Universidad Católica del Peru.
url: http://hdl.handle.net/20.500.12404/14600.

[64] Seaman, J., Seaman jr., J. and Stamey, J. [2011]. Hidden dangers of specifying noninformative priors, *The American Statistician* **66**(2): 77–84.
doi: https://www.doi.org/10.1080/00031305.2012.695938.

[65] Skrondal, A. and Rabe-Hesketh, S. [2004]. *Generalized Latent Variable Modeling: Multilevel, Longitudinal, and Structural Equation Models*, Interdisciplinary Statistics, Chapman Hall/CRC Press.

[66] Stan Development Team [2020]. RStan: the R interface to Stan. R package version 2.21.2.
url: http://mc-stan.org/.

[67] Stan Development Team. [2021]. *Stan Modeling Language Users Guide and Reference Manual, version 2.26*, Vienna, Austria.
url: https://mc-stan.org.

[68] Tarazona, E. [2013]. *Modelos alternativos de respuesta graduada con aplicaciones en la calidad de servicios*, Master's thesis, Pontificia Universidad Católica del Perú (PUCP).
url: http://hdl.handle.net/20.500.12404/6175.

[69] Vehtari, A., Simpson, D., Gelman, A., Yao, Y. and Gabry, J. [2021]. Pareto smoothed importance sampling.
url: https://arxiv.org/abs/1507.02646.

[70] Wainer, H., Bradlow, E. and Wang, X. [2007]. *Testlet response theory and its applications*, Cambridge University Press.

[71] Watanabe, S. [2013]. A widely applicable bayesian information criterion, *Journal of Machine Learning Research 14* **14**: 867–897.
url: https://jmlr.org/papers/volume14/watanabe13a/watanabe13a.pdf.

[72] Wollack, J. A., Bolt, D. M., Cohen, A. S. and 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: https://www.doi.org/10.1177/0146621602026003007.

[73] Yen, W. [1984]. Effects of local item dependence on the fit and equating performance of the three-parameter logistic model, *Applied Psychological Measurement* **8**(2): 125–145.
doi: https://doi.org/10.1177/014662168400800201.

AFDELING
Straat nr bus 0000
3000 LEUVEN, BELGIË
tel. + 32 16 00 00 00
fax + 32 16 00 00 00
www.kuleuven.be