2019 ■ Vol. 15 ■ no. 2

Longitudinal item response modeling and posterior predictive checking in R and Stan

Anna Scharl [a,] ✉ and Timo Gnambs [a,b]

[a]Leibniz Institute for Educational Trajectories
[b]Johannes Kepler University Linz

Abstract ■ Item response theory is widely used in a variety of research fields. Among others, it is the de facto standard for test development and calibration in educational large-scale assessments. In this context, longitudinal modeling is of great importance to examine developmental trajectories in competences and identify predictors of academic success. Therefore, this paper describes various multidimensional item response models that can be used in a longitudinal setting and how to estimate change in a Bayesian framework using the statistical software Stan. Moreover, model evaluation techniques such as the widely applicable information criterion and posterior predictive checking with several discrepancy measures suited for Bayesian item response modeling are presented. Finally, an empirical application is described that examines change in mathematical competence between grades 5 and 7 for $N = 1,371$ German students using a Bayesian longitudinal item response model.

Keywords ■ longitudinal, item response model, posterior predictive checking, Stan, Bayes, WAIC.
Tools ■ R, RStan.

✉ anna.scharl@lifbi.de

AS: 0000-0003-0081-1893; *TG*: 0000-0002-6984-1276

10.20982/tqmp.15.2.p075

Acting Editor ■ Denis Cousineau (Université d'Ottawa)

Reviewers
■ No reviewers.

Introduction

Empirical educational assessment has become increasingly important to educators and policy-makers who base their decisions on scientific knowledge (e.g., Educational Testing Service, 2018; Bundesministerium für Bildung und Forschung, 2017, 2018; Nuffield Foundation, 2018). Frequently, educational research relies on data collected in large-scale assessments, of which many have been around for decades such as the *National Assessment of Educational Progress* (NAEP) in the United States or the international *Trends In Mathematics and Science Study* (TIMSS) and the *Programme for International Student Assessment* (PISA). Lately, longitudinal measurement on an individual level has become more popular. In Germany, the *National Educational Panel Study* (NEPS) follows six distinct starting cohorts from birth to retirement over the course of their educational trajectories (Blossfeld & von Maurice, 2011) and the Organisation for Economic Co-operation and Development (OECD) has launched a longitudinal extension to PISA (Prenzel, Carstensen, Schöps, & Maurischat, 2006) and the *Programme for the International Assessment of Adult Competencies* (PIAAC) in Germany (PIAAC-L, Rammstedt, Martin, Zabal, Carstensen, & Schupp, 2017). Findings resulting from competence data collected in these studies have a huge impact on the perception of educational systems, their strengths and weaknesses, and, accordingly, how changes should be implemented (e.g., the German reaction to the PISA 2000 results: Kerstan, 2011; Finetti, 2010; Smolka, 2005).

Observed test data has to be calibrated, that is, latent competencies (e.g., reading comprehension, mathematical competence) are inferred from the responses of the participants. This inference is accomplished by using models of item response theory (IRT; OECD, 2012, 2014, 2017; Pohl & Carstensen, 2012; Martin, Mullis, & Hooper, 2016; Martin, Mullis, & Kennedy, 2007).

The next section presents an overview of popular item response models for longitudinal settings and describes a Bayesian approach for estimation, including posterior pre-

dictive checking (PPC) and the widely applicable information criterion (WAIC) as Bayesian approaches for model evaluation. In particular, the probabilistic programming language Stan is introduced as a way of implementing these Bayesian IRT models. Finally, an empirical example is presented to illustrate the longitudinal scaling of mathematical competence across two years in a sample of students from grade 5.

Item Response Modeling

Unidimensional Item Response Models

Item response models derive latent abilities of respondents and latent characteristics of items (e.g., difficulties) from the probability of correctly responding to an item or achieving a certain score on an item with more than two response categories. A basic item response model was introduced by Rasch (1960) and describes the binary response Y_{ij} of person i on item j with two parameters: the ability θ_i of person i and the difficulty β_j of item j:

$$P(Y_{ij} = 1|\theta_i, \beta_j) = logit^{-1}(\theta_i - \beta_j) \tag{1}$$

The Rasch model uses the logistic function to relate the observed responses to the item response model (Rasch, 1960; Adams, Wilson, & Wang, 1997; Patz & Junker, 1999a) although other link functions such as the cumulative distribution function (CDF) of the normal distribution can be used as well (Albert, 1992; Béguin & Glas, 2001; Aßmann, Gaasch, Pohl, & Carstensen, 2015, 2016). The logistic function can be transformed into the CDF of the normal distribution by adding a multiplicative constant $D = 1.7$ to the equation (Bowling, Khasawneh, Kaewkuekool, & Cho, 2009). Abilities and item difficulties are, thus, located on a common logit or probit scale.

The Rasch (1960) model assumes constant item slopes, that is, all items discriminate comparably between subjects with lower and higher competencies. In empirical applications, this assumption is frequently too strict (OECD, 2017). Therefore, this constraint can be relaxed to include different item slopes α_j, thus, resulting in the two parameter logistic model (2PL; Birnbaum, 1968).

$$P(Y_{ij}|\theta_i, \alpha_j, \beta_j) = logit^{-1}(\alpha_j\theta_i - \beta_j) \tag{2}$$

Considering multiple-choice items, that is, items with several response possibilities of which only one is correct, it is possible to guess the correct response without actually knowing the answer. The three parameter model (Birnbaum, 1968) takes guessing probabilities into account and models an additional parameter γ_j as a lower asymptote of the response probability.

$$\begin{aligned} P(Y_{ij}|\theta_i, \alpha_j, \beta_j, \gamma_j) =& \gamma_j + (1 - \gamma_j) \\ & \cdot \text{logit}^{-1}(\alpha_j\theta_i - \beta_j) \end{aligned} \tag{3}$$

So far, only item response models for dichotomous items have been considered. Frequently, aptitude tests also include items with more than two categories. Polytomous items can be modeled using so-called divide-by-total (e.g., the generalized partial credit model, GPCM; Muraki, 1992) or difference models (e.g., the graded response model, GRM; Samejima, 1969). Because difference and divide-by-total models are empirically largely indistinguishable (Naumenko, 2014), only the former will be considered. The GRM is formulated as

$$P(Y_{ij} = q|\omega) = P(Y_{ijq}) - P(Y_{ijq+1}) \tag{4}$$

where ω is a collective term for the parameters of the underlying model (e.g., the Rasch or 2PL model) and q is the respective item category with $q = 0, 1, \ldots, Q_j$. Additionally, an item category threshold κ is modeled. It is added to the item difficulty ($\beta_j + \kappa_{jq}$). The $Q_j + 2$ threshold parameters per item fulfill the ordering constraint

$$\kappa_{j0} = -\infty < \kappa_{j1} = 0 < \cdots < \kappa_{jQ_j} < \kappa_{jQ_j+1} = +\infty$$

From the ordering constraint follows

$$P(Y_{ij} = q|\omega) = \begin{cases} 1 - P(Y_{ij} = 1) & \text{if } q = 0, \\ P(Y_{ijq}) - P(Y_{ijq+1}) & \text{if } 1 \leq q < Q_j, \\ P(Y_{ijq}) & \text{else.} \end{cases} \tag{5}$$

As indicated by the name, difference models use the difference in cumulative probabilities to solve the item to model situations in which only partially correct answers were given.

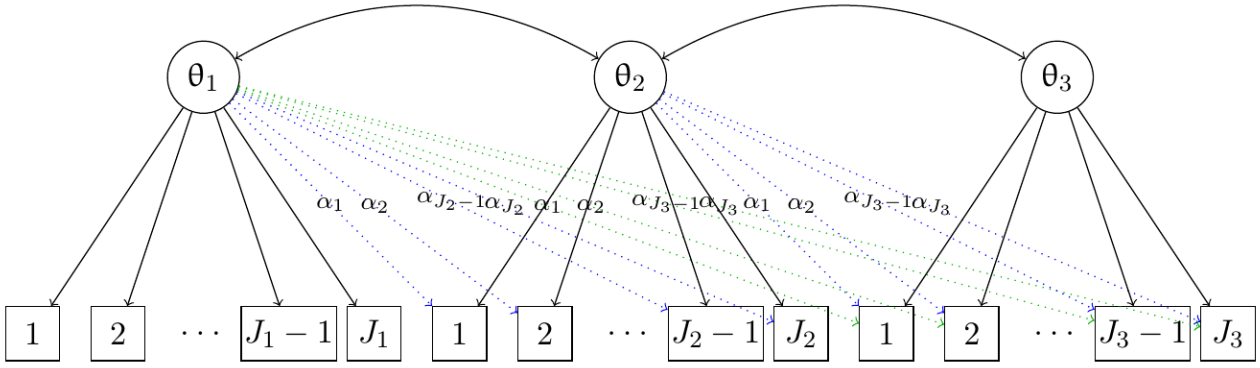
Longitudinal Item Response Modeling

Measuring change in abilities over time needs several prerequisites. There has to be overlapping information such as persons participating on several measurement occasions and test items or even complete test forms administered multiple times. Additionally, the latent correlation between the abilities can be used. Multidimensional IRT models are a natural way of incorporating all of this information (Ackerman, 1989; Adams et al., 1997).

$$P(Y_{ij} = 1|\vec{\theta}_i, \vec{\alpha}_j, \beta_j) = \text{logit}^{-1}\left(\sum_z^t \alpha_{jz}\theta_{iz} - \beta_j\right) \tag{6}$$

Abilities are now modeled as the vector $\vec{\theta}_i$. The vector contains T parameters, one for each measurement time point. Similarly, the item parameters are now vectors of length T. If measurement invariance holds for the item difficulties of common items, β_{jt} can be simplified to β_j. The item slopes, on the other hand, are now rows of the design matrix A of dimensions $J \times T$ with $J = J_1 + J_2 + \cdots + J_T$

Figure 1 ■ Longitudinal item response model for three measurement points with the total number of items $J = J_1 + J_2 + J_3$.



being the total number of items over all time points. The matrix describes which items were used at which measurement time point.

If distinct abilities are to be estimated for each time point, between multidimensional models should be employed, that is, all items load on a distinct latent ability (i.e., each row of A contains only one non-zero entry; Adams et al., 1997). If change is to be modeled, within multidimensional models should be used, that is, there are items that load on more than one latent ability (i.e., each row of A can contain multiple non-zero entries; Adams et al., 1997). Moreover, valid longitudinal models require an important constraint in the design matrix: all future latent abilities must not influence previous abilities, that is, all loadings of past test administration on future abilities must be constrained to zero (see Figure1).

For example, the design matrix for a within multidimensional model with two measurement time points and five items per time point can be defined as

$$\begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}$$

A basic longitudinal item response model is, for example, Embretson (1991)'s *Multidimensional Rasch Model for Learning and Change* (MRMLC). It assumes constant loadings α_j within and across time points (cf. Eq. 7) and test repetition (i.e., all items are common across time points and are assumed to be measurement invariant).

$$P(Y_{ij} = 1|\vec{\theta}_i, \beta_j) = \text{logit}^{-1}\left(\sum_z^t \theta_{iz} - \beta_j\right) \tag{7}$$

Similar to the MRMLC, which is a longitudinal extension of the Rasch model, the other models presented above can be formulated as multidimensional models. The longitudinal 3PL model is

$$P(Y_{ij}|\vec{\theta}_i, \vec{\alpha_j}, \beta_j, \gamma_j) = \gamma_j + (1 - \gamma_j) \cdot \text{logit}^{-1}\left(\sum_z^t \alpha_{jz}\theta_{iz} - \beta_j\right) \tag{8}$$

Note that all item parameters have the subscript j. This formulation assumes that the same test form has been administered multiple times and that all items function the same over several time points. By constraining the respective parameters, a longitudinal 2PL model or, again, the MRMLC is obtained.

IRT Scaling Procedure

IRT scaling can be described as a five-step procedure: First, proficiency test data has to be collected. Second, an IRT model, for example, one of those presented above, has to be agreed upon and estimated. Estimation methods can be divided into maximum likelihood and Bayesian methods. The latter are described below. Third, the estimated model has to be checked and compared to other models that might also be suitable in theory. This step may include model tweaking and re-estimation and re-evaluation of the models. Fourth, the best fitting model is chosen for the data. If several models fit equally well, the most parsimonious model is chosen. Fifth, if the aim of IRT scaling

Table 1 ■ Commonly used prior distributions in Bayesian item response modeling

Parameter	Distribution	Source (selection)
α	lognormal or truncated normal	Patz and Junker (1999a, 1999b), Béguin and Glas (2001), Sinharay, Johnson, and Stern (2006)
β	normal	Patz and Junker (1999a, 1999b), Béguin and Glas (2001), Sinharay, Johnson, and Stern (2006)
γ	beta[1]	Patz and Junker (1999b), Béguin and Glas (2001)
θ[2]	normal	Patz and Junker (1999a, 1999b), Béguin and Glas (2001), Sinharay, Johnson, and Stern (2006)
Z[3]	normal	Patz and Junker (1999a, 1999b), Béguin and Glas (2001), Sinharay, Johnson, and Stern (2006), Fox (2010), Albert (1992)

Note. Mostly, weakly-informative hyperparameters are chosen (limiting the support to sensible range within which the distribution is reasonably diffuse). It is recommended to assess the appropriateness of the hyperparameters and prior distributions empirically in sensitivity analyses. [1] For a more intuitive understanding of the functional shape, the parameters a and b can be transformed into mean ($\frac{a}{a+b}$) and weight ($a+b$). [2] To ensure model identification, mean and variance of the latent ability are usually fixed to 0 and 1 (at the first time point in the longitudinal case). [3] Z: latent response variable with realization Y (often used in data augmented Gibbs samplers; Béguin & Glas, 2001; Albert, 1992)

is producing ability estimates for further analysis, the estimates have to be extracted from the model.

Bayesian estimation of longitudinal item response models

In the last decades, Bayesian estimation of item response models has become more viable due to increased computational power and, consequently, more popular (Albert, 1992; Béguin & Glas, 2001; Santos, Moura, Andrade, & Gonçalves, 2016; Patz & Junker, 1999a, 1999b). Bayesian estimation of item response models means that, next to the information derived from the data collected in assessments, prior knowledge about the model parameters is incorporated into the statistical model in the form of prior distributions multiplied with the likelihood. Frequentist maximum likelihood estimation, which is widely used for IRT estimation in LSAs (Pohl & Carstensen, 2012; OECD, 2012), focuses solely on the optimization of the likelihood. Powerful methods of numerical analysis make maximum likelihood very efficient in lower-dimensional problems. In higher dimensional problems, on the other hand, optimization (especially numerical approximation of integrals) becomes quite inefficient and Bayesian approximation of integrals is more flexible and efficient (Betancourt, 2014).

Prior distributions

There are estimation schemes available for one to three parameter logistic and normal ogive models with multidimensional and multilevel extensions (e.g., Fox & Glas, 2001; Béguin & Glas, 2001; Santos et al., 2016). The aforementioned papers describe a variation of Markov Chain Monte Carlo (MCMC) algorithms. Commonly used prior distributions are summarized in Table 1. For example, for the item difficulty β and the proficiency θ typically normal distributions are adopted (Béguin & Glas, 2001; Patz & Junker, 1999a), whereas the item discrimination can be modeled using a lognormal prior (Béguin & Glas, 2001; Sinharay, Johnson, & Stern, 2006). Sensitivity analyses are recommended to decide on the appropriate prior distributions.

The competence data can be conceived as either binomially (binary data) or multinomially (ordered data) distributed. Assuming local independence, the likelihood of the 3PL item response model, for instance, can be formulated as follows

$$\mathcal{L} = P(Y|\Theta, A, \vec{\beta}, \Gamma) = \prod_i \prod_j P(Y_{ij})^{Y_{ij}} \cdot (1 - P(Y_{ij}))^{1-Y_{ij}} \quad (9)$$

The model is not identified (Béguin & Glas, 2001). To achieve model identification, the standard normal distribution is usually chosen as the prior distribution of the latent ability. In the longitudinal case, the means and standard deviations of the second and later time points are allowed to vary freely.

Following the separation strategy (Santos et al., 2016; Alvarez, Niemi, & Simpson, 2014; Barnard, McCulloch, & Meng, 2000), the co-variance structure of multidimensional models can be broken down to the individual variances S^2 and the correlation matrix R so that the covariance matrix $\Sigma = diag(S) \cdot R \cdot diag(S)$ is the product of the correlation matrix and the diagonal matrices with the standard deviations in the diagonal. The vector of standard deviations S can be modeled using a distribution with positive support (e.g., a lognormal, a truncated normal or

a uniform distribution with lower boundary 0). Barnard et al. (2000) proposed

$$f_d(R|\nu = T+1) \propto (\det R)^{\frac{T(T-1)}{2}-1}(\prod_i R_{ii})^{-\frac{(T+1)}{2}}, \quad (10)$$

with ν degrees of freedom, for the correlation matrix R. The distribution ensures that the marginal distributions of all individual correlation coefficients are uniform in the interval $[-1, 1]$.

Convergence checks

All MCMC algorithms require convergence checks. If the algorithms do not converge to a stationary posterior distribution, all inference based on the estimated data is invalid. Convergence checks can be done graphically or by using statistical indicators. Traceplots and autocorrelation plots give an overview of how well and fast the chain converged to a stationary distribution (Fox, 2010). Traceplots can also be used to check mixing if multiple chains have been initialized. The potential scale reduction factor $\hat{R}$ also serves this purpose (Gelman, Rubin, et al., 1992; Brooks & Gelman, 1998). It compares the estimated variances within and between chains. A detailed presentation on Bayesian convergence checkeing is beyond the scope of this paper. Interested readers are referred to introductory texts on Bayes modeling such as Gelman et al. (2014), Fox (2010), Kruschke (2014).

Model Evaluation

A number of errors can occur during the calibration of a test. Some of them may originate in the test development such as items that differentiate inappropriately among groups (differential item functioning (DIF); Dorans & Holland, 1992; Pohl & Carstensen, 2012) or do not discriminate sufficiently between different levels of competence. If those items are not excluded from the analysis and simply ignored, that is, not treated specially, the results of the calibration could be biased. Similarly, the model itself could have been a wrong choice from the multitude of models available to describe similar situations. Again, severely biased outcomes could be the result. The switch from Rasch modeling to two parameter modeling promoted by the PISA scientific board (OECD, 2017) is an example for the second source of error that was fixed by changing to a model that better fit the data. Hence, it is indispensable to check the applied models. In the following, two Bayesian ways of model checking will be described because this paper focuses on Bayesian IRT and many LSAs publish Bayesian competence scores.

Posterior Predictive Checking

Posterior predictive checking (PPC) is a powerful Bayesian model checking technique. There is rich literature on PPC in the context of item response theory (Sinharay, 2003, 2005; Sinharay et al., 2006; Sinharay, Guo, von Davier, & Veldkamp, 2009; Zhu & Stone, 2011, 2012; Li, Xie, & Jiao, 2017; Béguin & Glas, 2001; Fox, 2010). Also, posterior predictive model checking has been shown to be as accurate in selecting the most appropriate model from a range of candidate models as the DIC and the conditional predictive ordinate, but even more informative than the latter (Zhu & Stone, 2012). PPC draws on the property of a model describing the generating model accurately enough to also sufficiently describe future data from the same generating model. That is, the model shows good predictive performance. Accordingly, data is predicted from the posterior predictive distribution (PPD) of the model

$$p(y^{rep}|Y) = \int P(y^{rep}|\omega)P(\omega|Y)d\omega \quad (11)$$

The PPD consists of the posterior distribution of a model $P(\omega|Y)$ and the likelihood of the predicted data y^{rep} given the model parameters ω. In general, it is not necessary to solve the integral. As most Bayesian algorithms use Markov Chain Monte Carlo techniques, those algorithms can be extended to also simulate from the PPD (Fox, 2010; Rubin, 1984; Sinharay, 2006). Thus, the originally surveyed data can be located in the PPD. If it is typical, the model fits considerably well.

Discrepancy measures

To assess typicality, so-called discrepancy measures are computed that summarize relevant characteristics of the data. Relevance is, of course, determined by the question at hand. In item response modeling, a number of discrepancy measures have been proposed and tested in simulation and field studies (e.g., Sinharay, 2006; Sinharay et al., 2009; Fox, 2010; Béguin & Glas, 2001; Zhu & Stone, 2011, 2012; Li et al., 2017). The most commonly used discrepancy measures are given below.

Odds ratio of item pairs The global odds ratio (OR) is a measure for binary item pairs. It has been shown useful in detecting local item dependence and multidimensionality, and even greater deviations from model implied item slopes (Chen & Thissen, 1997; Sinharay et al., 2006; Fox, 2010; Li et al., 2017; Zhu & Stone, 2011, 2012). It is calculated as

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

with $n_{jj'}$ the number of subjects scoring j on the first and j' on the second item, with $j, j' = 0, 1$.

Item-total correlation coefficient The item-total correlation coefficient (ITC) is usually used to assess if a slope parameter is missing in the analysis, but also whether local item dependence occurred (Sinharay et al., 2006; Sinharay

et al., 2009; Li et al., 2017; Zhu & Stone, 2011, 2012). Depending on the item format (i.e., whether the items are binary, ordered or even metric), the respective (i.e., biserial, polyserial or Pearson) correlation coefficient has to be used.

Observed Score Distribution The observed score distribution (OSD) is based on the total scores of the subjects. It is defined as

$$\chi^2_{NC} = \sum_{j=0}^{J} \frac{(NC_j - E(NC_j))^2}{E(NC_j)}$$

with $NC_{j=(0,1,\ldots,J)}$ as the number of subjects scoring j items correctly, and $NC = (NC_0, \ldots, NC_J)$. The OSD is frequently used to assess the fit of the prior chosen for the latent ability distribution, but also whether a pseudo-guessing parameter is missing from the model (Sinharay et al., 2006; Li et al., 2017; Béguin & Glas, 2001; Fox, 2010; Zhu & Stone, 2011, 2012). The discrepancy can either be quantified using the χ^2 statistic or by graphic display, that is, plotting the observed and predicted OSD in overlay (Li et al., 2017). Sinharay et al. (2009) successfully used the total score and grouped total score distributions in the context of a latent regression model.

Yen's Q_1 Yen's Q_1 is used to assess global fit of latent trait models (Yen, 1981, 1984). In PPC, it was used to detect violations to the functional form of the models (e.g., constant item thresholds in generalized partial credit models; Zhu & Stone, 2011, 2012; Li et al., 2017). After rank-ordering the subjects according to their latent trait and splitting them into ten evenly populated cells, it is calculated as

$$Q_{1j} = \sum_{r=1}^{10} \frac{N_r(O_{jr} - E_{jr})^2}{E_{jr}(1 - E_{jr})}$$

with N_r, the number of subjects in cell r, O_{jr}, the observed proportion of subjects scoring correctly on item j, and $E_{jr} = \frac{1}{N_r}\sum_{k\in r}^{N_r} P(Y_{kj} = 1|\vec{\theta}_k, \vec{\alpha}_j, \beta_j)$, the predicted proportion of subjects scoring correctly on item j. The global statistic $Q_1 = \sum_{j=1}^{J} Q_{1j}$ is the sum of the item statistics.

Yen's Q_3 of item pairs Yen's Q_3 is defined as the correlation of the residuals of an item pair across all individuals (Yen, 1981, 1984).

$$Q_{3jj'} = cor(d_j d_{j'})$$

with $d_{ij} = Y_{ij} - P(Y_{ij} = 1|\vec{\theta}_i, \vec{\alpha}_j, \beta_j)$, the residual term for item j. Unidimensionality is indicated by values of zero. The statistic has been shown to capture deviations from unidimensionality and local independence (Li et al., 2017; Zhu & Stone, 2011, 2012).

The aforementioned discrepancy measures generally worked well in the studies that employed them. Of course, they only worked as far as they are designed to work (e.g., observed score distributions did not deliver conclusive results in Zhu and Stone (2011) or Li et al. (2017) because neither study exhibited problems that were supposed to be detected by the observed score distribution). Similarly, under some conditions, the odds ratio statistic was more efficient than the Q_3 statistic and vice versa (Li et al., 2017).

In previous literature (Li et al., 2017) items with extreme posterior predictive p-values (PPP values), for example, less than 0.05 or greater than 0.95, were considered problematic. Ideally, PPP values, expressing the proportion of replicated discrepancy measures being more extreme than the original data's discrepancy measure, range around 0.5 (Li et al., 2017; Meng, 1994; Gelman et al., 2013). This signifies random deviation in the data and, thus, no systematic errors in the model.

Widely Applicable Information Criterion

The widely applicable information criterion (WAIC; Watanabe, 2010) is a fully Bayesian alternative to the deviance information criterion (DIC; Spiegelhalter, Best, Carlin, & Van Der Linde, 2002), remedying the DIC's weak points (Vehtari & Gelman, 2014; Vehtari, Gelman, & Gabry, 2017). The WAIC is defined as

$$WAIC = -2 \cdot (\widehat{lpd} - \hat{p}_{waic}) \tag{12}$$

where $\widehat{lpd}$ is the log pointwise predictive density computed from posterior simulations of the likelihood and $\hat{p}_{waic}$ is the estimated effective number of parameters which is calculated using the posterior variance of the lpd (Vehtari & Gelman, 2014). The WAIC is an approximation of leave-one-out cross-validation (Watanabe, 2010) and, thus, a measure of predictive accuracy of the model (Vehtari & Gelman, 2014).

Implementing Bayesian IRT in Stan

After defining a statistical model, it has to be implemented in some statistical software. In the field of Bayesian estimation, Stan (Stan Development Team, 2017) is a very powerful implementation of a Hamiltonian Monte Carlo algorithm (Betancourt, 2017; Gelman et al., 2014) that allows the fast and efficient exploration of posterior distributions even in higher dimensions. There is a number of articles giving tutorials on Stan (e.g., Luo & Jiao, 2018; Jiang & Carter, 2018; Sorensen, Hohenstein, & Vasishth, 2016). These are extended to multidimensional longitudinal IRT modeling in this article and an account of posterior predictive checking in Stan and R is given. Stan comes in a variety of different flavors, but because the analyses rely on R,

rstan, the R interface for Stan, is used (Stan Development Team, 2018). The installation of *rstan* is detailed on https://github.com/stan-dev/rstan/wiki/RStan-Getting-Started and is analogous to installing regular R packages.

In R, the models are estimated using the function `rstan::stan()`. Expected a posteriori estimators can be extracted from the final `stanfit` object using the function `rstan::get_posterior_mean()`. Usage details are described in the *rstan* documentation. Similarly, the Stan modeling language is detailed in the Stan user's guides. Both are accessible at https://mc-stan.org/users/documentation/. This paper focuses on the model specification in the Stan modeling language. The model itself can either be written as a character string in R or as a separate text file with the extension *.stan*. The latter is recommended for debugging purposes. If errors in the model syntax occur, the line numbers given in the error messages will match if the model is stored in a separate file, but probably not if it is part of a larger R script. A Stan model is composed of several blocks whose scope is limited by curly braces. It always ends with a blank line.

The longitudinal three parameter logistic model

In the following section, the longitudinal three parameter logistic model (Eq. 8) as the most complex model will be implemented in Stan model code. It will be detailed code block by code block. By imposing restrictions on the parameters, the two parameter logistic model (Eq. 2) or Rasch model (Eq. 1) can be obtained. To obtain the graded response model (Eq. 4), several major modifications need to be implemented. The respective code for these other models is given in the online supplement. Note that at some points generalizations (e.g., using the variable T for the number of time points instead of directly writing 2) are chosen. If the code is later to be adapted for more time points, not all parts of the code have to be replaced.

The functions block

The functions block is optional. To use the density function of Barnard et al. (2000) (Eq. 10) for correlation matrices, the block has to be specified using Listing 1 (all the listings are collected in the Appendix). Note that the log probability is returned because Stan operates only on log probabilities. Furthermore, the names of probability density functions have to end in `_log`.

The data block

In the data block given in Listing 2, the input for `rstan::stan()` is specified. Here, the observed test data is called *Y*. It is a matrix of dimensions *number of persons* $\times$ *number of items* over T time points.

The number of items per time point are modeled separately as elements of a vector to accommodate test repetitions with different numbers of items and perhaps only a subset of common items. If the exact same test is administered multiple times, the number of items J can be modeled as a scalar.

The parameters block

The parameters block given in Listing 3 contains all parameters that are estimated. If common items exist between time points, they have to be modeled by either estimating the common items only once or by estimating all items as unique items. In the latter case, the average of the common item parameters is later used in the model likelihood.

The transformed parameters block

In the optional transformed parameters block seen in Listing 4, the estimated parameters are transformed so that they can be used in the estimation of other parameters. This includes linear transformations, aggregations and reparameterizations. The order within this block is as follows: first, the transformed parameters have to be declared, then they can be defined.

Assuming test repetition, all item parameters are averaged over time points. The item discrimination parameters are estimated freely. The cross-loadings are stored in the vector $alpha[(sum(J)+1):(sum(J)+J[2])]$. Mean and variance of the latent ability of the first measurement time point are fixed to 0 and 1 to ensure model identification.

The model block

In the model block (Listing 5), the prior distributions and likelihood function are defined. Stan contains a number of predefined distributions for this purpose. As in the transformed parameters block, auxiliary variables have to be declared in the beginning and can later be defined.

The model syntax can be simplified in case of the Rasch and 2PL models by using `bernoulli_logit()` instead of `bernoulli(inv_logit())` in the likelihood function.

The generated quantities block

In the optional generated quantities block (Listing 6), the replication of data in the context of PPC is declared. Stan contains a number of random number generators for this purpose. Thus, data is simulated based on the random parameter draws of each post-warmup iteration of the HMC sampler. Note the nested looping over items and persons. The random number generator does not support vectorization. Furthermore, the log likelihood for the WAIC has to be declared here because Stan generally does not differentiate between likelihood and prior distribution in its computations (Vehtari & Gelman, 2014).

Again, the code can be simplified in the Rasch and 2PL case to `bernoulli_logit_rng()` instead of `bernoulli_rng(inv_logit())` and `bernoulli_logit_lpmf()` instead of `bernoulli_lpmf(inv_logit())`. The next steps are to run the model using the Stan interface of one's choice and then evaluate it.

Implementing Posterior Predictive Checking in R

The code that is presented in this section is based on the data structures – replicated and estimated – returned by the $\mathrm{stan}()$ function. The parameter rep used in the functions denotes the number of replicated data sets per estimated model (i.e., the number of iterations minus the warmup iterations and divided by the thinning interval). The R code can be simplified to accommodate the original data in wide format easily.

Odds ratio of item pairs

To speed up the computation, R's vectorization was used. Instead of looping over all replicated data sets, the property that a three-dimensional array held constant in one dimension becomes a matrix was used. Because the first dimension of parameters estimated in Stan is always the iteration, if the items are held constant, summing over the rows will result in aggregated statistics of the persons for each replicated data set. See Listing 7.

Item-total correlation coefficient

Again, vectorization speeds up the calculation. Furthermore, computation is more efficient when using the apply function instead of for loops. Thus, first the person total score is calculated, then correlated with each items responses. This results in a matrix of which the diagonal holds the correlation of the respective iteration of the replicated data sets. See Listing 8.

Observed Score Distribution

The observed score distribution is actually nothing more than the total score distribution. Thus, the procedure is similar to that used in the ITC computation (see Listing 9).

Yen's Q statistics

For Yen's Q_1 and Q_3, the probabilities of a correct solution to the responses have to be computed. Furthermore, both statistics need auxiliary variables that, for example, contain the expected values per group for Q_1. To keep things concise, only the final estimation functions for the statistics are given below. How to initialize the auxiliary variables and the function for calculating the solution probability can be taken from the online supplement.

Yen's Q_1

Other than the discrepancy measures used so far, Yen's Q_1 cannot be computed from the replicated data alone. The expected values are calculated from the posterior means of the estimated parameters extracted from the $\mathrm{stanfit}$ object. The groups are formed by assigning indexes in rank order of the latent ability, as seen in Listing 10.

Yen's Q_3 of item pairs

Like Yen's Q_1, Q_3 has need of auxiliary variables calculated from the posterior means of the parameter estimates. Again, the expected values are computed as solution probabilities. First, the differences in observed and expected values are calculated and then correlated for each item pair. See Listing 11.

Real Data Example

Sample

This study used data from mathematical competence tests administered to a sample of the National Educational Panel Study[1] that is representative for German fifth graders in 2010 (Blossfeld & von Maurice, 2011). The analyses are limited to $N = 1,371$ students (43% female) that had no missing data in grades 5 and 7.

Instruments

Mathematical competence was assessed using a test with different response formats for the items including dichotomous and polytomous multiple choice items (Schnittjer & Duchhardt, 2015). The present analyses are limited to the dichotomous items. Thus, 21 of the 24 items of the grade 5 test and 22 of the 23 items of the grade 7 test were used in the analyses.

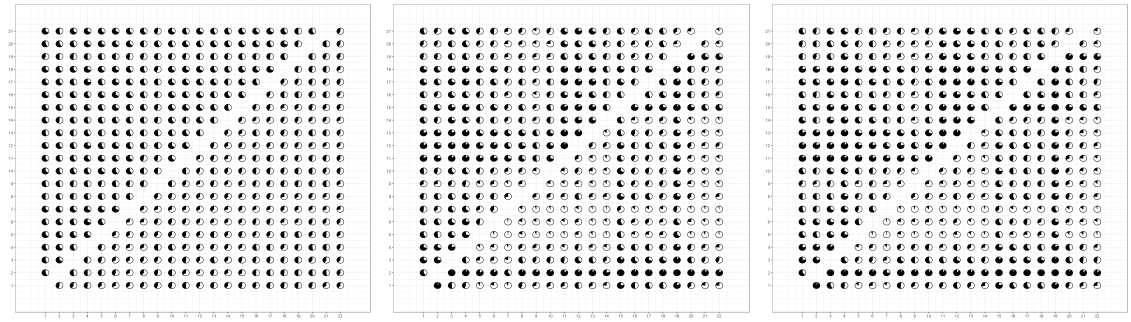
Software and packages

All analyses were performed in Stan (version 2.17.0; Stan Development Team, 2017) and R (version 3.5.1; R Core Team, 2018). The R packages rstan (version 2.17.3) and edstan (version 1.0.6; Stan Development Team, 2018; Furr, 2017) were used as the Stan interface and for convergence diagnostics of the HMC sampler. The R packages haven (version 2.0.0) and tidyr (version 0.8.1; Wickham & Miller, 2018; Wickham & Henry, 2018) were used for data read-in and cleaning. The packages ggplot2 (version 3.0.0) and scatterpie (version 0.1.2; Wickham, 2016; Yu, 2018) were used for graphical display of the PPC results, whereas the package loo (version 2.0.0; Vehtari, Gabry, Yao, & Gelman,

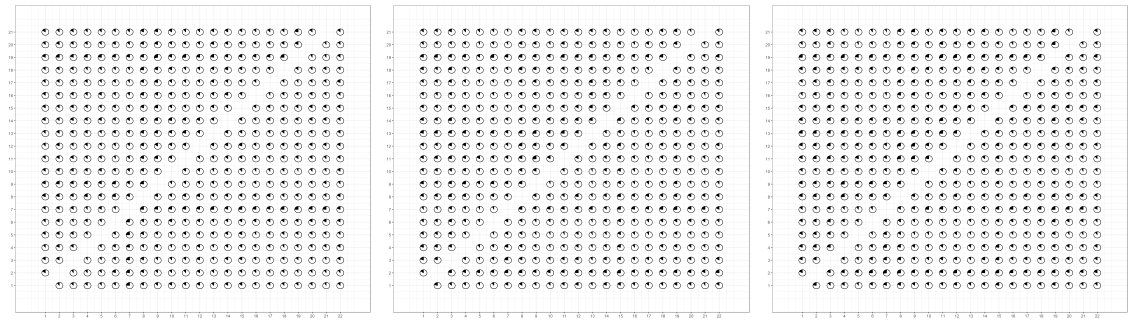
[1]The data can be downloaded free of charge via the NEPS homepage (https://www.neps-data.de/). Please note that a data use agreement with the NEPS research data center is a prerequisite for data access.

Figure 2 ■ PPP values for items and item pairs. (a) OR – from left to right: Rasch, 2PL, and 3PL model. Upper triangle: first time point (tp), lower triangle: second tp; (b) Q_3 – from left to right: Rasch, 2PL, and 3PL model. Upper triangle: first tp, lower triangle: second tp; (c) Q_1 – from left to right: Rasch, 2PL, and 3PL model; (d) Q_1 – from left to right: Rasch, 2PL, and 3PL model.

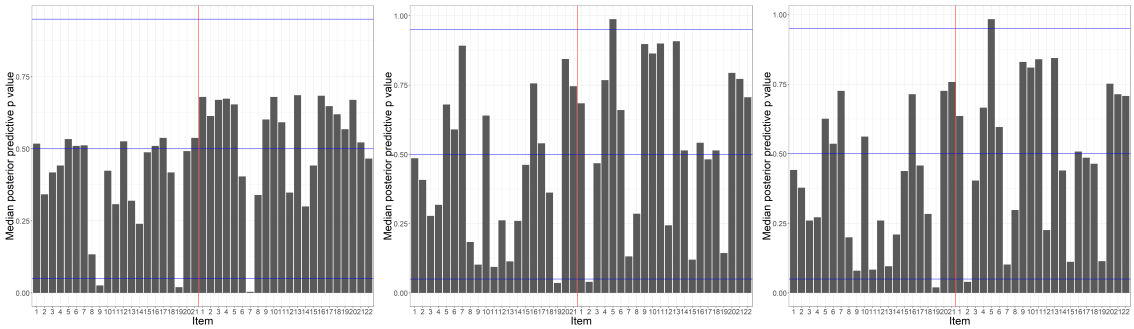
(a)



(b)



(c)



(d)

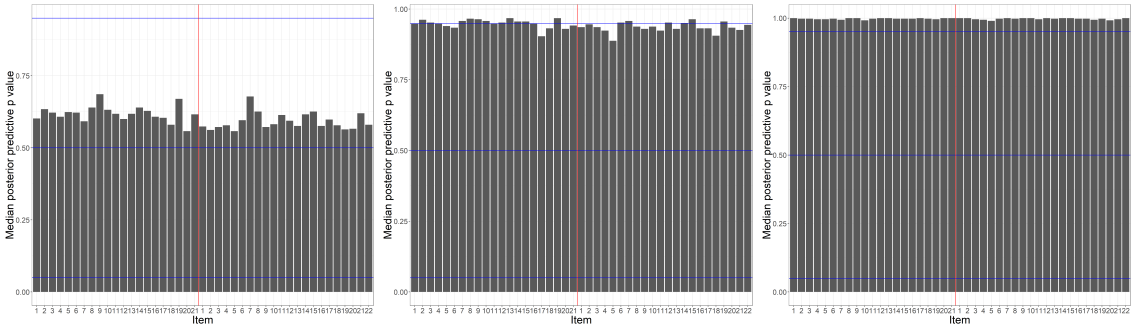


Table 2 ■ Potential scale reduction factor $\hat{R}$ for the longitudinal Rasch model.

	Grade 5	Grade 7			
Item number	β	β	μ	SD	ρ
1	1.00	1.00	1.01	1.14	1.13
2	1.01	1.00			
3	1.00	1.01			
4	1.00	1.00			
5	1.00	1.00			
6	1.00	1.00			
7	1.00	1.00			
8	1.00	1.00			
9	1.00	1.00			
10	1.00	1.00			
11	1.00	1.00			
12	1.00	1.00			
13	1.00	1.01			
14	1.00	1.00			
15	1.00	1.00			
16	1.01	1.00			
17	1.00	1.00			
18	1.00	1.00			
19	1.01	1.02			
20	1.00	1.00			
21	1.00	1.00			
22		1.00			

Note. β item difficulty; μ mean of the latent ability; SD standard deviation of the latent ability; ρ correlation of the latent abilities

2018) was used for WAIC evaluation.

Statistical Analyses

The longitudinal versions of the Rasch model (Eq. 1), the 2PL model (Eq. 2) and the 3PL model (Eq. 8) were applied to the data. All code for data preparation and calculating and processing the discrepancy measures is given in the online supplement. The Stan code for all models described in this article can also be found in the online supplement. All models were invoked with 3000 iterations, 2000 warmup iterations and thinning of four for two chains, resulting in 500 posterior draws.

The discrepancy measures described in this paper were computed for the original and replicated data sets. PPP values are obtained as the proportion of replicated measures more extreme (e.g., larger) than the original measure, that is, they can be computed as means of the binary evaluation of the extremeness of the measure. Graphical display gives a comprehensive overview of the models in view of the large number of PPP values.

Results

Model fit

Convergence checks Convergence diagnostics results were mixed for the different models. The longitudinal Rasch model converged well as indicated by graphical checks of the traceplots and the potential scale reduction factor $\hat{R}$ (values less than 1.1 indicate convergence, Table 2). The longitudinal 2PL and 3PL model did similarly well, only the cross-loadings had trouble converging (Table 3, 4). For all models, the ability hyperparameters have slight troubles converging.

Evaluation using PPC All estimated models were evaluated using odds ratio, Yen's Q_1, Yen's Q_3, the item-total correlation and the observed score distribution as discrepancy measures. For the item-based measures, PPP values were calculated and plotted (see Figure 2). For the OSD, a subset of ten randomly drawn subject's distributions were inspected graphically (one of which is shown for each model in Figure 3). The replicated observed score distributions did not show any systematic deviations. Next to graphical analysis, the item-based measures were eval-

Table 3 ■ Potential scale reduction factor $\hat{R}$ for the longitudinal 2PL model.

	Grade 5		Grade 7					
Item number	α	β	α	β	α	μ	SD	ρ
1	1.01	1.00	1.01	1.01	1.12	1.00	1.03	1.27
2	1.00	1.00	1.03	1.03	1.06			
3	1.00	1.00	1.03	1.03	1.03			
4	1.00	1.00	1.01	1.01	1.08			
5	1.00	1.00	1.00	1.00	1.12			
6	1.00	1.00	1.03	1.03	1.10			
7	1.00	1.00	1.00	1.00	1.08			
8	1.00	1.00	1.00	1.00	1.10			
9	1.00	1.00	1.04	1.04	1.10			
10	1.00	1.00	1.04	1.03	1.13			
11	1.00	1.00	1.01	1.02	1.07			
12	1.00	1.00	1.02	1.02	1.12			
13	1.00	1.00	1.02	1.02	1.17			
14	1.00	1.00	1.00	1.00	1.03			
15	1.00	1.00	1.01	1.01	1.16			
16	1.00	1.00	1.01	1.02	1.13			
17	1.00	1.00	1.01	1.02	1.13			
18	1.00	1.01	1.00	1.00	1.11			
19	1.00	1.00	1.01	1.02	1.08			
20	1.00	1.00	1.00	1.01	1.08			
21	1.00	1.00	1.01	1.01	1.12			
22			1.00	1.00	1.11			

Note. β item difficulty; α item discrimination; μ mean of the latent ability; SD standard deviation of the latent ability; ρ correlation of the latent abilities

uated using the rule of thumb for extreme PPP values (Figure 4).

The evaluation revealed irregularities about several different aspects in the respective models. The 3PL and 2PL models exhibit striking values on Yen's Q_1, both for each item and globally (cf. Figure 2). This indicates problems with the functional form. Both models perform similarly poor when evaluated using the odds ratio, Yen's Q_3 and ITC statistics (cf. Figure 2). The longitudinal Rasch model, on the other hand, performs well under Yen's Q_1. The odds ratio do not seem alarmingly biased. Yen's Q_3, on the other hand, exhibits more extreme values although no items are flagged by the statistic for any of the models (Figure 4). The item-total correlation flags the items 9 and 19 (first time point) and 7 (second time point) when the Rasch model is applied. The items 7 and 9 are repeated items, thus, the measure might indicate problems with measurement invariance or, also, that some items might benefit from changed discrimination parameters.

Evaluation using WAIC The model with the smallest WAIC can be considered the best-fitting model. Table 5 shows the WAIC and respective standard errors for the different models. The WAIC, contrary to PPC, seems to favor the 2PL model over the 3PL and the Rasch model although the standard errors overlap and the result is, thus, not conclusive.

Model selection

Considering convergence and PPC results, it seems appropriate to select the most parsimonious model, the longitudinal Rasch model, although the WAIC favors this model least. But because the trouble seemed to lie predominantly with the cross-loadings, a 2PL or 3PL model without or with constant cross-loadings (i.e., fixed to one as in the MRMLC) might be a valid solution as well. A brief comparison of the percentage of flagged items, on the other hand (Figure **??**), shows that the 2PL model with constant cross-loadings could be chosen. This is supported by the WAIC values (Table 6) although the standard errors overlap here as well.

Discussion

This paper gave an overview of Bayesian longitudinal IRT, WAIC and PPC as well as the implementation of IRT and

Table 4 ■ Potential scale reduction factor $\hat{R}$ for the longitudinal 3PL model.

	Grade 5			Grade 7						
Item number	α	β	γ	α	β	γ	α	μ	SD	ρ
1	1.00	1.00	1.00	1.00	1.00	1.00	1.07	1.01	1.16	1.20
2	1.00	1.00	1.00	1.00	1.00	1.00	1.01			
3	1.00	1.00	1.00	1.00	1.00	1.00	1.09			
4	1.01	1.00	1.00	1.00	1.00	1.00	1.04			
5	1.01	1.00	1.00	1.00	1.00	1.00	1.09			
6	1.00	1.00	1.00	1.00	1.00	1.00	1.03			
7	1.00	1.00	1.00	1.00	1.00	1.00	1.09			
8	1.00	1.00	1.00	1.00	1.00	1.00	1.12			
9	1.00	1.00	1.00	1.00	1.00	1.00	1.03			
10	1.00	1.00	1.00	1.00	1.00	1.00	1.05			
11	1.00	1.00	1.00	1.01	1.01	1.00	1.04			
12	1.01	1.01	1.01	1.00	1.00	1.00	1.10			
13	1.00	1.00	1.00	1.00	1.00	1.01	1.08			
14	1.00	1.00	1.00	1.00	1.01	1.00	1.05			
15	1.00	1.00	1.00	1.00	1.01	1.00	1.04			
16	1.00	1.00	1.00	1.01	1.01	1.00	1.08			
17	1.01	1.00	1.00	1.00	1.01	1.00	1.08			
18	1.00	1.00	1.00	1.01	1.00	1.00	1.11			
19	1.00	1.00	1.00	1.00	1.00	1.00	1.05			
20	1.00	1.00	1.00	1.00	1.00	1.00	1.07			
21	1.00	1.00	1.00	1.01	1.01	1.00	1.06			
22				1.00	1.00	1.00	1.10			

Note. β item difficulty; α item discrimination; γ guessing parameter of items; μ mean of the latent ability; SD standard deviation of the latent ability; ρ correlation of the latent abilities

Table 5 ■ WAIC and standard error of WAIC per model

Model	WAIC	SE
Rasch	59667.43	243.89
2PL	59346.42	245.78
3PL	59410.47	242.60

PPC in R and Stan. There are R packages that facilitate the use of WAIC for Stan models. Thus, the hurdle to use those techniques has become low.

Many studies have shown that PPC is able to detect model deviations and estimation problems in IRT. Applied to a real data example, PPC identified the longitudinal Rasch model as the most fitting model. This is in line with competence test construction in the NEPS which aims at Rasch model conform test forms. The WAIC, on the other hand, did not deliver clear results, but slightly favored the 2PL model. Because both model diagnostic methods need additional data output, they are quite memory intensive. If working memory is a critical bottleneck, it might be necessary to choose one of the methods. Because PPC are more informative than WAIC and, thus, can be used for more detailed investigation of the models, they might be preferred in initial model evaluations.

Assuming model misfit was detected, one of several steps could be taken: Firstly, the model could be modified (e.g., by leaving out problematic items or by scoring them differently if the item-total correlation indicated under- or over-estimation). Secondly, the model could be changed as a whole (e.g., by adding a discrimination or guessing parameter, or by choosing the most parsimonious model of a range of models) if problems are detected with a larger number of the items investigated. Thirdly, the model could be used as it is while reporting the detected problems of the model.

In real data applications, the HMC sampler could be started with more chains (e.g., four instead of two) and a

Figure 3 ■ Observed Score distribution. (a) Rasch model: subject 15; (b) Two parameter logistic model: subject 191; (c) Three parameter logistic model: subject 129.

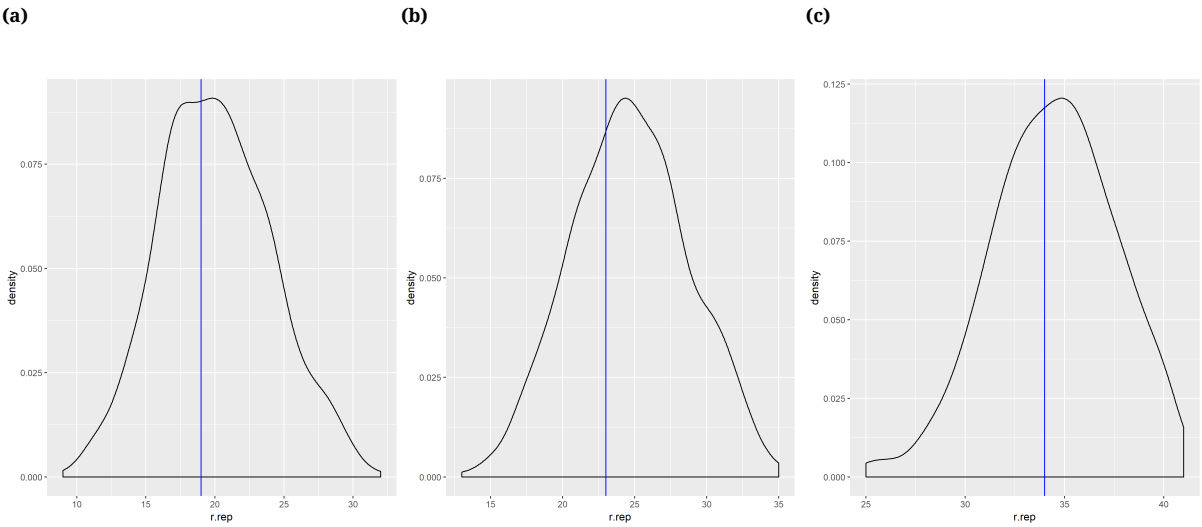


Table 6 ■ WAIC and standard error of WAIC per model

Model	WAIC	SE
Rasch nc	59663.16	243.88
2PL cc	59482.85	248.08
2PL nc	59361.60	245.65
3PL cc	59455.36	242.66
3PL nc	59417.05	242.72

Note. "cc" means constant cross-loadings, "nc" means no cross-loadings.

larger number of iterations. This would lead to better measurement precision, but aggravate memory problems. It would also allow larger thinning intervals to counter possible issues because of autocorrelation in the Markov chains.

Future research should again broaden the scope. For example, more than two time points should be modeled. Also, several different IRT models should be applied (e.g., routines for models for ordered data with different maximum categories, or different link functions in a longitudinal setting). A mixture of hierarchical and multidimensional modeling could be employed to fully capture the features of longitudinal data. Furthermore, the ever-present problem of missing data in large-scale assessments has not been addressed in this study. Combinations of IRT estimation and multiple imputation strategies should be investigated. Another subject could be the difference in estimation schemes as most large-scale assessments are currently using a two-step approach combining maximum likelihood and Bayesian estimation (Sinharay et al., 2009; OECD, 2017) instead of fully Bayesian approaches.

Regardless of the acceleration of computational speed and power of personal computers, the limitations still encompass the computational costs of this study. While they are extended by the repetitive nature of testing a larger variety of competing models, it has to be stressed that Bayesian computation is expensive and, especially if sample sizes increase, hours might have to be invested into the estimation of the model and also into the model evaluation (e.g., the aggregation of the PPC information). More parsimonious models, especially with smaller sample sizes, will take much less time than their more complex counterparts. On the other hand, much less information is available in small sample situations which entails a different set of challenges.

Authors' note

This paper uses data from the National Educational Panel Study (NEPS): Starting Cohort Grade 5, doi:10.5157/NEPS:SC3:7.0.1. From 2008 to 2013, NEPS data was collected as part of the Framework Program for the Promotion of Empirical Educational Research funded by the German Federal Ministry of Education and Research

Figure 4 ■ Percentage of items with extreme PPP values (<.05 or >.95).

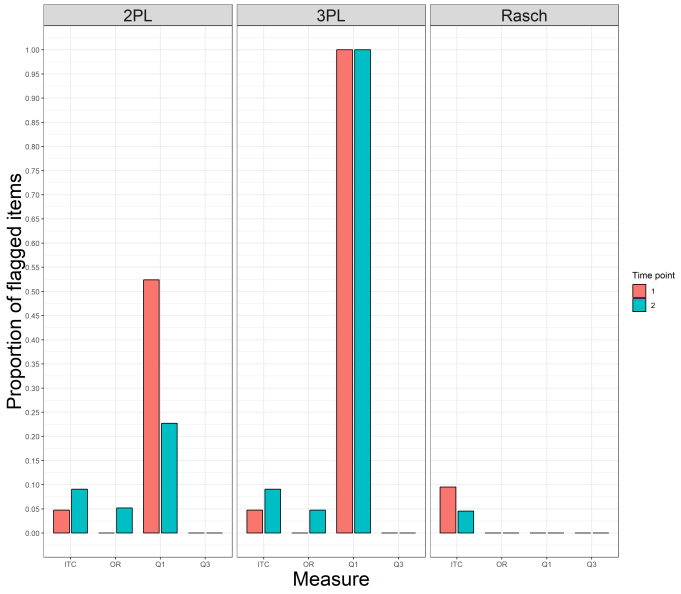
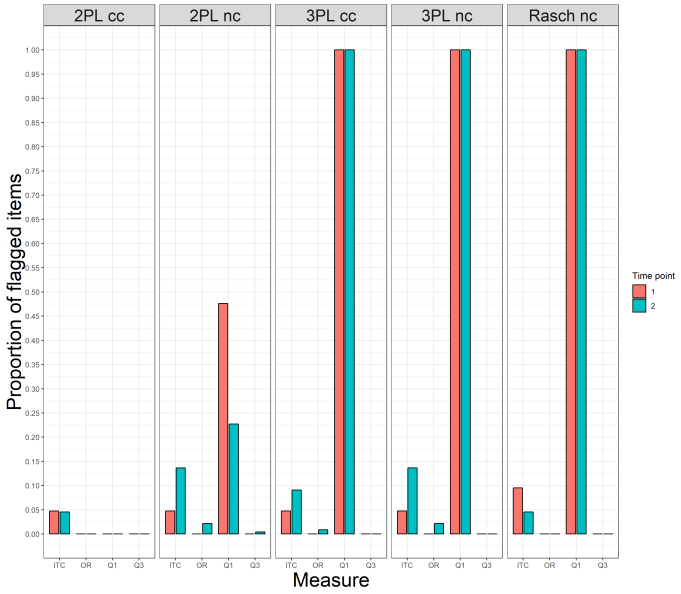


Figure 5 ■ Percentage of items with extreme PPP values (<.05 or >.95). "cc" denotes constant cross-loadings, "nc" no cross-loadings in the model.



(BMBF). As of 2014, NEPS is carried out by the Leibniz Institute for Educational Trajectories (LIfBi) at the University of Bamberg in cooperation with a nationwide network.

Correspondence concerning this article should be addressed to Anna Scharl, Leibniz Institute for Educational Trajectories, Wilhelmsplatz 3, 96047 Bamberg, Germany, E-mail: anna.scharl@lifbi.de.

References

Ackerman, T. A. (1989). Unidimensional IRT calibration of compensatory and noncompensatory multidimensional items. *Applied Psychological Measurement*, *13*(2), 113–127. doi:10.1177/014662168901300201

Adams, R. J., Wilson, M., & Wang, W.-c. (1997). The multidimensional random coefficients multinomial logit model. *Applied Psychological Measurement*, *21*(1), 1–23. doi:10.1177/0146621697211001

Albert, J. H. (1992). Bayesian Estimation of Normal Ogive Item Response Curves Using Gibbs Sampling. *Journal of Educational Statistics*, *17*(3), 251–269. doi:10.2307/1165149

Alvarez, I., Niemi, J., & Simpson, M. (2014). Bayesian inference for a covariance matrix. *arXiv preprint arXiv:1408.4050*.

Aßmann, C., Gaasch, C., Pohl, S., & Carstensen, C. H. (2015). Bayesian estimation in IRT models with missing values in background variables. *Psychological Test and Assessment Modeling*, *57*(4), 595–618.

Aßmann, C., Gaasch, C., Pohl, S., & Carstensen, C. H. (2016). Estimation of Plausible Values Considering Partially Missing Background Information: A Data Augmented MCMC Approach. In *Methodological Issues of Longitudinal Surveys* (pp. 503–521). Springer.

Barnard, J., McCulloch, R., & Meng, X.-L. (2000). Modeling covariance matrices in terms of standard deviations and correlations, with application to shrinkage. *Statistica Sinica*, 1281–1311.

Béguin, A. A., & Glas, C. A. (2001). MCMC estimation and some model-fit analysis of multidimensional IRT models. *Psychometrika*, *66*(4), 541–561. doi:10.1007/BF02296195

Betancourt, M. (2014). *Efficient Bayesian inference with Hamiltonian Monte Carlo*. MLSS Iceland 2014. Retrieved from https://www.youtube.com/watch?v=pHsuIaPbNbY

Betancourt, M. (2017). A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*.

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. *Statistical Theories of Mental Test Scores*.

Blossfeld, H.-P., & von Maurice, J. (2011). Education as a lifelong process. *Zeitschrift für Erziehungswissenschaft*, *14*(2), 19–34. doi:10.1007/s11618-011-0179-2

Bowling, S. R., Khasawneh, M. T., Kaewkuekool, S., & Cho, B. R. (2009). A logistic approximation to the cumulative normal distribution. *Journal of Industrial Engineering and Management*, *2*(1). doi:10.3926/jiem.2009.v2n1.p114-127

Brooks, S. P., & Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of computational and graphical statistics*, *7*(4), 434–455. doi:10.2307/1390675

Bundesministerium für Bildung und Forschung. (2017). BMBF stärkt Bildungsforschung in Deutschland. Retrieved from https://www.bmbf.de/de/forschung-fuer-gute-bildung-4524.html

Bundesministerium für Bildung und Forschung. (2018). Bildungsforschung. Retrieved from https://www.bmbf.de/de/bildungsforschung-1225.html

Chen, W.-H., & Thissen, D. (1997). Local dependence indexes for item pairs using item response theory. *Journal of Educational and Behavioral Statistics*, *22*(3), 265–289. doi:10.2307/1165285

Dorans, N. J., & Holland, P. W. (1992). DIF detection and description: Mantel-Haenszel and standardization. *ETS Research Report Series*, *1992*(1), i–40. doi:10.1002/j.2333-8504.1992.tb01440.x

Educational Testing Service. (2018). About. Retrieved from https://www.ets.org/about

Embretson, S. E. (1991). A multidimensional latent trait model for measuring learning and change. *Psychometrika*, *56*(3), 495–515. doi:10.1007/BF02294487

Finetti, M. (2010). Schock mit Folgen. Retrieved from https://www.sueddeutsche.de/karriere/fuenf-jahre-pisa-schock-mit-folgen-1.558990

Fox, J.-P. (2010). *Bayesian item response modeling: Theory and applications*. Springer Science & Business Media.

Fox, J.-P., & Glas, C. A. (2001). Bayesian estimation of a multilevel IRT model using Gibbs sampling. *Psychometrika*, *66*(2), 271–288. doi:10.1007/BF02294839

Furr, D. C. (2017). *edstan: Stan Models for Item Response Theory*. R package version 1.0.6. Retrieved from https://CRAN.R-project.org/package=edstan

Gelman, A. et al. (2013). Two simple examples for understanding posterior p-values whose distributions are far from uniform. *Electronic Journal of Statistics*, *7*, 2595–2602. doi:10.1214/13-EJS854

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2014). *Bayesian data analysis*. CRC press Boca Raton, FL.

Gelman, A., Rubin, D. B. et al. (1992). Inference from iterative simulation using multiple sequences. *Statistical science*, *7*(4), 457–472. doi:10.1214/ss/1177011136

Jiang, Z., & Carter, R. (2018). Using Hamiltonian Monte Carlo to estimate the log-linear cognitive diagnosis model via Stan. *Behavior Research Methods*, 1–12. doi:10.3758/s13428-018-1069-9

Kerstan, T. (2011). Der heilsame Schock. Retrieved from https://www.zeit.de/2011/49/C-Pisa-Rueckblick

Kruschke, J. (2014). *Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan*. Academic Press.

Li, T., Xie, C., & Jiao, H. (2017). Assessing fit of alternative unidimensional polytomous IRT models using posterior predictive model checking. *Psychological Methods*, *22*(2), 397. doi:10.1037/met0000082

Luo, Y., & Jiao, H. (2018). Using the Stan program for Bayesian item response theory. *Educational and Psychological Measurement*, *78*(3), 384–408. doi:10.1177/0013164417693666

Martin, M. O., Mullis, I. V. S., & Hooper, M. (Eds.). (2016). *Methods and Procedures in TIMSS 2015*. Boston College, TIMSS & PIRLS International Study Center. Retrieved from http://timssandpirls.bc.edu/publications/timss/2015-methods.html

Martin, M. O., Mullis, I. V. S., & Kennedy, A. M. (2007). *Progress in International Reading Literacy Study (PIRLS): PIRLS 2006 Technical Report*. ERIC.

Meng, X.-L. (1994). Posterior predictive p-values. *The Annals of Statistics*, 1142–1160. doi:10.1214/aos/1176325622

Muraki, E. (1992). A generalized partial credit model: Application of an EM algorithm. *ETS Research Report Series*, *1992*(1), i–30. doi:10.1002/j.2333-8504.1992.tb01436.x

Naumenko, O. (2014). Comparison of various polytomous item response theory modeling approaches for task based simulation cpa exam data. *AICPA 2014 Summer Internship Project*.

Nuffield Foundation. (2018). Education. Retrieved from http://www.nuffieldfoundation.org/education

OECD. (2012). PISA 2009 Technical Report. Retrieved from http://dx.doi.org/10.1787/9789264167872-en

OECD. (2014). PISA 2012 Technical Report. Retrieved from https://www.oecd.org/pisa/pisaproducts/PISA-2012-technical-report-final.pdf

OECD. (2017). PISA 2015 Technical Report. Retrieved from https://www.oecd.org/pisa/data/2015-technical-report/

Patz, R. J., & Junker, B. W. (1999a). A straightforward approach to Markov chain Monte Carlo methods for item response models. *Journal of Educational and Behavioral Statistics*, *24*(2), 146–178. doi:10.2307/1165199

Patz, R. J., & Junker, B. W. (1999b). Applications and extensions of MCMC in IRT: Multiple item types, missing data, and rated responses. *Journal of Educational and Behavioral Statistics*, *24*(4), 342–366. doi:10.2307/1165367

Pohl, S., & Carstensen, C. H. (2012). *NEPS Technical Report – Scaling the Data of the Competence Tests*. Otto-Friedrich-Universität, National Educational Panel Study. Bamberg.

Prenzel, M., Carstensen, C. H., Schöps, K., & Maurischat, C. (2006). Die Anlage des Längsschnitts bei PISA 2003 [The longitudinal design in PISA 2003]. In P. K. Deutschland (Ed.), *PISA 2003: Untersuchungen zur Kompetenzentwicklungim Verlauf eines Schuljahres [PISA 2003: Investigations of the development of competencies across one school year]* (pp. 29–62). Münster, Germany: Waxmann.

R Core Team. (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. Retrieved from https://www.R-project.org/

Rammstedt, B., Martin, S., Zabal, A., Carstensen, C., & Schupp, J. (2017). The PIAAC longitudinal study in Germany: Rationale and design. *Large-scale Assessments in Education*, *5*(1), 4. doi:10.1186/s40536-017-0040-z

Rasch, G. (1960). Probabilistic models for some intelligence and achievement tests. *Copenhagen: Danish Institute for Educational Research*.

Rubin, D. B. (1984). Bayesianly justifiable and relevant frequency calculations for the applied statistician. *The Annals of Statistics*, *12*(4), 1151–1172. doi:10.1214/aos/1176346785

Samejima, F. (1969). Estimation of latent ability using a response pattern of graded scores. *Psychometrika Monograph Supplement*. doi:10.1007/BF03372160

Santos, V. L. F., Moura, F. A., Andrade, D. F., & Gonçalves, K. C. (2016). Multidimensional and longitudinal item response models for non-ignorable data. *Computational Statistics & Data Analysis*, *103*, 91–110. doi:10.1016/j.csda.2016.05.002

Schnittjer, I., & Duchhardt, C. (2015). Mathematical competence: Framework and exemplary test items. *Leibniz Institute for Educational Trajectories (LIfBi)*.

Sinharay, S. (2003). Practical applications of posterior predictive model checking for assessing fit of common item response theory models. *ETS Research Report Series*, *2003*(2). doi:10.1002/j.2333-8504.2003.tb01925.x

Sinharay, S. (2005). Assessing Fit of Unidimensional Item Response Theory Models Using a Bayesian Approach.

Journal of Educational Measurement, *42*(4), 375–394. doi:10.1111/j.1745-3984.2005.00021.x

Sinharay, S. (2006). Bayesian item fit analysis for unidimensional item response theory models. *British Journal of Mathematical and Statistical Psychology*, *59*(2), 429–449. doi:10.1348/000711005X66888

Sinharay, S., Guo, Z., von Davier, M., & Veldkamp, B. P. (2009). Assessing fit of latent regression models. *ETS Research Report Series*, *2009*(2), i–25. doi:10.1002/j.2333-8504.2009.tb02207.x

Sinharay, S., Johnson, M. S., & Stern, H. S. (2006). Posterior Predictive Assessment of Item Response Theory Models. *Applied Psychological Measurement*, *30*(4), 298–321. doi:10.1177/0146621605285517

Smolka, D. (2005). PISA - Konsequenzen für die Bildung und Schule. Retrieved from http://www.bpb.de/apuz/29164/pisa-konsequenzen-fuer-bildung-und-schule?p=all

Sorensen, T., Hohenstein, S., & Vasishth, S. (2016). Bayesian linear mixed models using Stan: A tutorial for psychologists, linguists, and cognitive scientists. *The Quantitative Methods for Psychology*, *12*(3), 175–200. doi:10.20982/tqmp.12.3.p175

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *64*(4), 583–639. doi:10.1111/1467-9868.00353

Stan Development Team. (2017). *Stan Modeling Language Users Guide and Reference Manual*. Version 2.17.0. Retrieved from http://mc-stan.org

Stan Development Team. (2018). RStan: The R interface to Stan. R package version 2.17.3. Retrieved from http://mc-stan.org/

Vehtari, A., Gabry, J., Yao, Y., & Gelman, A. (2018). Loo: Efficient leave-one-out cross-validation and waic for bayesian models. R package version 2.0.0. Retrieved from https://CRAN.R-project.org/package=loo

Vehtari, A., & Gelman, A. (2014). WAIC and cross-validation in Stan. *Helsinki: Aalto University*.

Vehtari, A., Gelman, A., & Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, *27*(5), 1413–1432. doi:10.1007/s11222-016-9696-4

Watanabe, S. (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, *11*(Dec), 3571–3594.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. Retrieved from http://ggplot2.org

Wickham, H., & Henry, L. (2018). *tidyr: Easily Tidy Data with 'spread()' and 'gather()' Functions*. R package version 0.8.1. Retrieved from https://CRAN.R-project.org/package=tidyr

Wickham, H., & Miller, E. (2018). *haven: Import and Export 'SPSS', 'Stata' and 'SAS' Files*. http://haven.tidyverse.org, https://github.com/tidyverse/haven, https://github.com/WizardMac/ReadStat.

Yen, W. M. (1981). Using simulation results to choose a latent trait model. *Applied Psychological Measurement*, *5*(2), 245–262. doi:10.1177/014662168100500212

Yen, W. M. (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:10.1177/014662168400800201

Yu, G. (2018). *scatterpie: Scatter Pie Plot*. R package version 0.1.2. Retrieved from https://CRAN.R-project.org/package=scatterpie

Zhu, X., & Stone, C. A. (2011). Assessing fit of unidimensional graded response models using Bayesian methods. *Journal of Educational Measurement*, *48*(1), 81–97. doi:10.1111/j.1745-3984.2011.00132.x

Zhu, X., & Stone, C. A. (2012). Bayesian comparison of alternative graded response models for performance assessment applications. *Educational and Psychological Measurement*, *72*(5), 774–799. doi:10.1177/0013164411434638

Appendix follows.

Appendix A: The listing mentionned in the text.

Listing 1: Custom density function for correlation matrices

```
functions {
    real corr_mat_pdf_log(matrix R, real k) {
        real log_dens;
        log_dens = ((k * (k - 1)) / 2) - 1) * log_determinant(R) + (-((k + 1) / 2)) * sum(
    log(diagonal(R)));
        return log_dens;
```

```
    }}
```

Listing 2: Input data

```
data {
    int<lower=1> I; // number of persons
    int<lower=1> T; // number of time points
    int<lower=1> J[T]; // number of items per time point
    int<lower=0, upper=1> Y[I, sum(J)]; // binary item response data
}
```

Listing 3: Parameters to be estimated

```
parameters {
    matrix[I, T] theta; // latent ability
    vector<lower=0>[sum(J)+J[2]] alpha; // item discrimination
    vector[sum(J)] beta; // item difficulty
    vector<lower=0, upper=1>[sum(J)] gamma; // guessing
    real mu; // prior mean of latent ability (time point 2)
    corr_matrix[T] R; // correlation matrix of latent ability
    real<lower=0> SD; // std. deviation of latent ability (time point 2)
}
```

Listing 4: Transformed parameters for the likelihood function

```
transformed parameters {
    vector[T] mutheta;
    vector[T] S;
    cov_matrix[T] sigmatheta;
    vector[sum(J)] BETA; // item difficulty
    vector<lower=0>[sum(J)+sum(J[2:T]))] ALPHA; // item discrimination
    vector<lower=0, upper=1>[sum(J)] GAMMA; // guessing

    // set hyperparameters for proficiency
    mutheta[1] = 0;
    mutheta[2] = mu;
    S[1] = 1;
    S[2] = SD;
    sigmatheta = diag_matrix(S) * R * diag_matrix(S);

    // average separately estimated item parameters
    for (j in 1:J[1]) ALPHA[j] = (alpha[j]+alpha[j+J[1]]) / 2;
    ALPHA[(J[1]+1):(sum(J))] = ALPHA[1:J[1]];
    ALPHA[(sum(J)+1):(sum(J)+sum(J[2:T]))] = alpha[(sum(J)+1):(sum(J)+sum(J[2:T]))];
    for (j in 1:J[1]) BETA[j] = (beta[j]+beta[j+J[1]]) / 2;
    BETA[(J[1]+1):sum(J)] = BETA[1:J[1]];
    BETA[(sum(J)+1):(sum(J)+sum(J[2:T]))] = beta[(sum(J)+1):(sum(J)+sum(J[2:T]))];
    for (j in 1:J[1]) GAMMA[j] = (gamma[j]+gamma[j+J[1]]) / 2;
    GAMMA[(J[1]+1):sum(J)] = GAMMA[1:J[1]];
    GAMMA[(sum(J)+1):(sum(J)+sum(J[2:T]))] = gamma[(sum(J)+1):(sum(J)+sum(J[2:T]))];}
```

Listing 5: Prior distributions and likelihood of the model

```
model {
```

```
    // prior distributions on the hyperparameters
    mu ~ normal(1, 3);
    R ~ corr_mat_pdf(T);
    SD ~ normal(1, 3) T[0, ];
    // prior distributions on the parameters
    for (i in 1:I) {
        theta[i, ] ~ multi_normal(mutheta, sigmatheta);
    }
    beta ~ normal(0, 3);
    for (j in 1:(sum(J)+sum(J[2:T])))) alpha[j] ~ normal(1, 1) T[0, ];
    gamma ~ beta(12.5, 37.5);
    // likelihood of the data
    for (j in 1:sum(J)) {
        if (j > J[1]) {
           Y[, j] ~ bernoulli(GAMMA[j] + (1 - GAMMA[j]) * inv_logit(ALPHA[j+J2] * theta[, 1]
                + ALPHA[j] * theta[, 2] - BETA[j]));
        } else {
           Y[, j] ~ bernoulli(GAMMA[j] + (1 - GAMMA[j]) * inv_logit(ALPHA[j] * theta[, 1]
               - BETA[j]));
        }}}
```

Listing 6: Replicating data for posterior predictive checking

```
generated quantities {
    int y_rep[I, sum(J)];
    real log_lik[I, sum(J)];
    // replicated data
    for (i in 1:I) {
        for (j in 1:sum(J)) {
            if (j > J[1]) {
                y_rep[i, j] = bernoulli_rng(GAMMA[j] + (1 - GAMMA[j]) *
                inv_logit(ALPHA[j+J[2]] * theta[i, 1] + ALPHA[j] * theta[i, 2] - BETA[j]));
            } else {
                y_rep[i, j] = bernoulli_rng(GAMMA[j] + (1 - GAMMA[j]) * inv_logit(ALPHA[j] *
                theta[i, 1] - BETA[j]));}}}
    // individual log-likelihood
    for (i in 1:I) {
        for (j in 1:sum(J)) {
            if (j > J[1]) {
                log_lik[i, j] = bernoulli_lpmf(Y[i, j] | GAMMA[j] + (1 - GAMMA[j]) *
                inv_logit(ALPHA[j+J[2]] * theta[i, 1] + ALPHA[j] * theta[i, 2] - BETA[j]));
           } else {
               log_lik[i, j] = bernoulli_lpmf(Y[i, j] | GAMMA[j] + (1 - GAMMA[j]) *
               inv_logit(ALPHA[j] * theta[i, 1] - BETA[j]));
           }}}}
```

Listing 7: Odds ratio calculated in R

```
#' @param y_rep (rep) x (pers) x (item) array; replicated data
#' @param n (patterns) x (rep); number of persons solving item pairs in pattern xy
#' @param J total number of items
#' @param or (rep) x (no. item pairs)
create_odds_ratio <- function(y_rep, n, J, or) {
    count <- 1
    for (j in seq(J)) {
```

```
        i <- 1
        while (i<j) {
            n[1,] <- rowSums(y_rep[, , i] == 1 & y_rep[, , j] == 1)
            n[2,] <- rowSums(y_rep[, , i] == 0 & y_rep[, , j] == 0)
            n[3,] <- rowSums(y_rep[, , i] == 1 & y_rep[, , j] == 0)
            n[4,] <- rowSums(y_rep[, , i] == 0 & y_rep[, , j] == 1)
            or[, count] <- (n[1,]*n[2,])/(n[3,]*n[4,])
            colnames(or)[count] <- paste0('ItemPair', i, '_', j)
            count <- count + 1
            i <- i + 1}}
    return(or)}
```

Listing 8: Item-total correlation implemented in R

```
#' @param y_rep (rep) x (pers) x (item) array; replicated data
#' @param r (rep) x (items) matrix
#' @param J total number of items
create_r <- function(y_rep, J, r) {
    x <- apply(y_rep, 1, rowSums) # rows: pers, cols: reps
    for (j in seq(J)) {
        r[, j] <- diag(apply(y_rep[, , j], 1,
                        FUN = function(y) {
                            apply(x, 2, FUN = function(xx) {
                                cor(y, xx, method = "pearson")
                        }))))}
    return(r)}
```

Listing 9: Observed score distribution implemented in R

```
#' @param y_rep (rep) x (pers) x (item) array; replicated data
#' @param J total number of items
#' @param J2 number of items at time point 2
create_osd <- function(y_rep, J, J2){
    p <- list()
    # for each matrix holds: rows: persons, cols: reps
    p[["overall"]] <- apply(y_rep, 1, rowSums)
    p[["t1"]] <- apply(y_rep, 1, function(x) rowSums(x[, 1:(J-J2)]))
    p[["t2"]] <- apply(y_rep, 1, function(x) rowSums(x[, (J-J2+1):ncol(x)]))
    return(p)}
```

Listing 10: Yen's Q1 implemented in R

```
#' @param y_rep (rep) x (pers) x (item) array; replicated data
#' @param E list of (pers per group) x (items) matrices; expected values
#' @param J number of items
#' @param q (rep) x (items); initialized to 0
#' @param s list of length 10; each list element contains person indexes of the resp. group
create_yens_q1 <- function(y_rep, E, J, q, s) {
    # observed values
    O <- replicate(length(s), matrix(0, dim(y_rep)[1], J), simplify = FALSE)
    for (r in seq(dim(y_rep)[1])) {
```

```
        for (i in seq(length(s))) {
            O[[i]][r, ] <- colMeans(y_rep[r, s[[i]], ])}}
    # Q1
    for (i in seq(length(s))) {
        for (j in seq(J)) {
            q[, j] <- q[, j] + (length(s[[i]]) * (O[[i]][, j] -
                E[[i]][j]))^2 / (E[[i]][j] * (1 - E[[i]][j]))}}
    return(q)}
```

Listing 11: Yen's Q3 implemented in R

```
#' @param y_rep (rep) x (pers) x (item) array; replicated data
#' @param d (rep) x (pers) x (dim) array; differences y−p
#' @param p (pers) x (items) matrix; solution probabilities
#' @param J number of items
#' @param q (rep) x (no. item pairs)
create_yens_q3 <- function(y_rep, d, p, J, q) {
    count <-1
    for (j in seq(J)) {
        i <- 1
        while (i<j) {
            d[, , 1] <- t(t(y_rep[, , i]) - p[, i])
            d[, , 2] <- t(t(y_rep[, , j]) - p[, j])
            q[, count] <- apply(d, 1, function(x) cor(x)[1, 2])
            colnames(q)[count] <- paste0('ItemPair', i, '_', j)
            count <- count + 1
            i <- i + 1}}
    return(q)}
```

Open practices

The *Open Material* badge was earned because supplementary material(s) are available on the journal's web site.

Citation

Scharl, A., & Gnambs, T. (2019). Longitudinal item response modeling and posterior predictive checking in R and Stan. *The Quantitative Methods for Psychology*, *15*(2), 75–95. doi:10.20982/tqmp.15.2.p075

Received: 23/05/2019 ~ Accepted: 18/03/2019