




Combining Process Information and Item Response Modeling to Estimate Problem-Solving Ability

Yue Xiao,  Beijing Normal University, Bernard Veldkamp,  University of Twente, and Hongyun Liu,  Beijing Normal University

Abstract: *The action sequences of respondents in problem-solving tasks reflect rich and detailed information about their performance, including differences in problem-solving ability, even if item scores are equal. It is therefore not sufficient to infer individual problem-solving skills based solely on item scores. This study is a preliminary attempt to incorporate process data analysis into the measurement of problem-solving ability. The entire procedure consists of using information from process data as prior information for the estimation of problem-solving proficiency in an item response model. The purpose of this study is twofold: (1) to investigate the impact of adding process information on the estimation of latent ability; (2) to examine the extent to which the ability estimates obtained from the combination model can reflect the information of the problem-solving process. Seven problem-solving items from the Programme for International Assessment of Adult Competencies were used. Results indicate that the inclusion of process priors enhances the correlation between proficiency estimates and process information related to the problem-solving strategies adopted by respondents, as well as to their solution efficiency. The inclusion of process priors further reveals differences in the problem-solving performance of respondents exhibiting the same score pattern and increases precision of latent ability estimation.*

Keywords: ability estimation, Bayesian framework, item response theory, problem-solving, process data analysis

Problem-solving ability is defined as “an individual’s capacity to engage in cognitive processing to understand and resolve problem situations where a method of solution is not immediately obvious” (OECD, 2013, p. 122). It is regarded as one of the most sophisticated aspects of human cognition (Newell & Simon, 1972), and it has been recognized as one of the most important skills in the 21st century (Care et al., 2012; National Research Council, 2012). The measurement of this skill poses a major challenge, however, due to its complexity.

Advances in technology have promoted a new approach to educational and psychological assessment—computer-based assessment (CBA)—which further facilitates the growing interest in assessing problem-solving skills and knowledge in technology-related environments. The Programme for the International Assessment of Adult Competencies (PIAAC) was the first international assessment of adult skills to be administered predominantly by computer. It targets adults between the ages of 16 and 65 years, and it assesses three domains of cognitive skills: literacy, numeracy, and problem-solving in technology-rich environments (PSTRE; Schleicher, 2008). The PSTRE domain focuses on the ability to set goals, plan and monitor progress, as well as to acquire, evaluate and make use of information through digital technology, communication tools, and networks (OECD, 2016). Using computers as a delivery platform, the PSTRE assessment in the PIAAC consists of interactive scenario-based items that simulate real-life situations as closely as possible. The data collected include information on not only whether

respondents were able to solve tasks (i.e., item outcomes), but also how they approached the solution and how much time they spent doing so. Such data are known as process data (He & von Davier, 2016).

As a by-product of computer-based assessment, process data are detailed records of the behaviors of respondents in solving digital tasks. They are typically presented as time-stamped sequences of actions generated during problem-solving processes. The information in process data is particularly valuable when examining interactive problem-solving tasks, because the action sequences are detailed records of how test-takers achieve the success or failure of tasks, providing valid evidence for identifying the problem-solving strategies used by respondents (Goldhammer et al., 2013; He, Borgonovi, et al., 2019), as well as their cognitive processes (e.g., Arieli-Attali et al., 2019) and other aspects. Thus, process data allow the possibility to gain deeper insight into the latent construct measured in problem-solving items, which cannot be captured by item scores (Stadler et al., 2020). Accordingly, traditional measurement models inferring the latent trait based solely on item outcomes or scores may be inappropriate for problem-solving tasks, due to the use of insufficient information.

To estimate the latent trait more accurately, researchers have started to include timing data (a part of process data) into the construction of the measurement model (e.g., Guo et al., 2016; van der Linden et al., 2010; Wang & Xu, 2015). The exploitation of behavioral information is still in the early

stages, however, due to its complexity and unstructured nature. Most previous efforts have focused on extracting information from process behaviors (e.g., Goldhammer et al., 2013; Greiff et al., 2015; He, Veldkamp, et al., 2019). Only a few studies have explored how the information contained in response behaviors can be used to facilitate the estimation of latent traits (e.g., Lamar, 2018; Liu et al., 2018; Shu et al., 2017). The models proposed in these studies have different requirements for the application situation and the form of process information, as discussed later in this article. Therefore, this study aims to combine the extraction of process information with the measurement model to improve the estimation of latent ability, and subsequently to explore the impact of including process information in this manner on the estimation of problem-solving proficiency, as compared to the use of item scores alone.

A Brief Review of Process Data Analysis

Process data are more complex than traditional test responses, in which a univariate response is observed for each item. Each response process is a sequence of categorical actions, and its length varies across individuals. Due to the highly detailed information in the records, process data appear quite unstructured, such that traditional measurement models are largely inapplicable. In addition, analysis is complicated by a lack of understanding of the cognitive process underlying human-computer interaction and the noises in the response process.

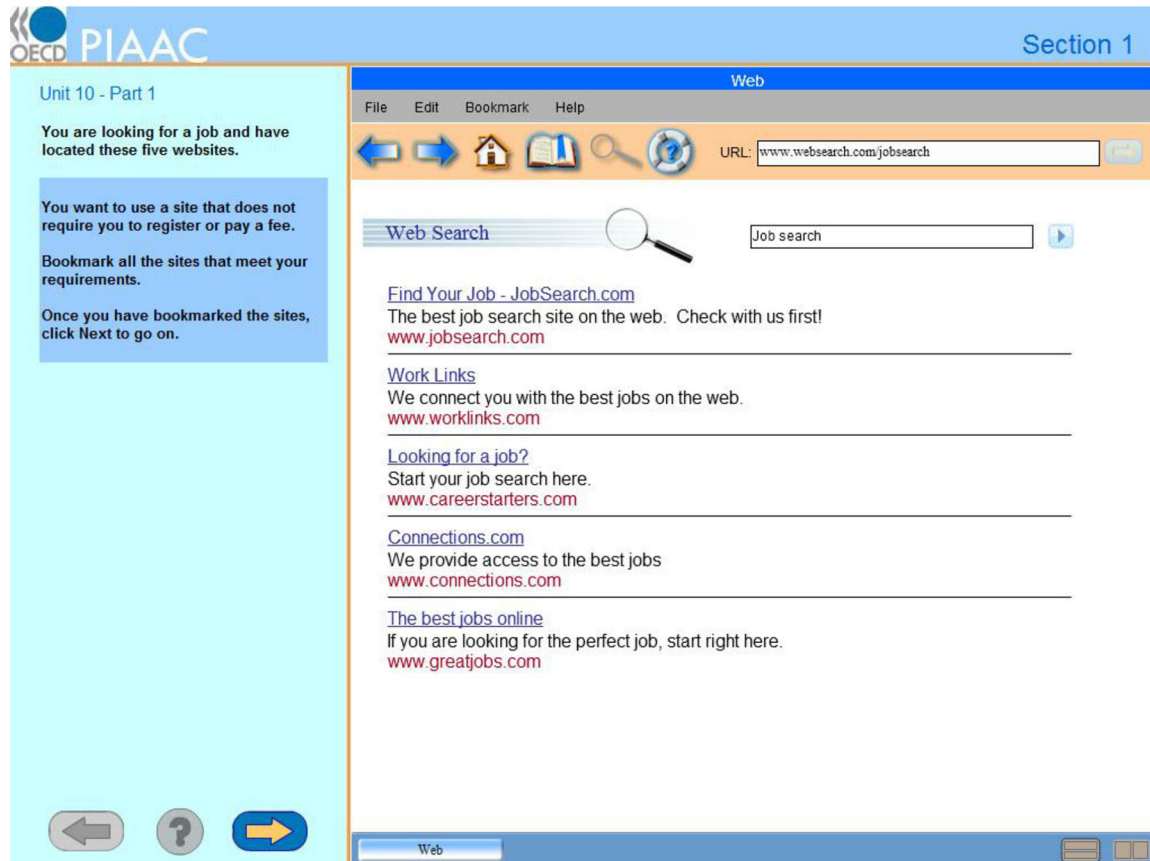
The utilization of process information can be divided roughly into the use of response time and the use of response behaviors. Measurement models have been proposed to estimate latent traits based on both traditional responses (i.e., item outcomes or item scores) and response time (e.g., Guo et al., 2016; van der Linden et al., 2010; Wang & Xu, 2015). In comparison, research on process behaviors is still in the exploratory stages, which can be roughly categorized into methods of information extraction and measurement models. Most investigations have been limited to the extraction of item-specific information to explore meaningful behavioral characteristics. For example, researchers may derive behavioral indicators from process data according to theory or expert input (such as the number of clicking “reset,” the time spent on one page, the number of times a certain strategy was used), and then examining the relationships between these indicators and other variables of interest (e.g., Greiff et al., 2015; Han et al., 2019; Lee & Haberman, 2016). Some researchers have also proposed bottom-up approaches to identify behavioral patterns directly from complete or short action sequences, for example, using n-grams (He & von Davier, 2016), hidden Markov models (e.g., Biswas et al., 2010), and social network analysis (Zhu et al., 2016). The information obtained through these approaches facilitates the understanding of the problem-solving process. However, it is difficult to aggregate the information across multiple items due to the item-specific nature, and it is also difficult to use this information in measurement models.

Various scholars have proposed new ideas of transforming behavioral sequences into single numbers or vectors to facilitate generalization of information extraction across items. In many cases, the distance between two or more sequences is computed. For instance, He, Borgonovi, et al. (2019)

apply the longest common subsequence (LCS) method to identify the distance between each observed action sequence for an item and the predefined sequence that subject-matter experts consider optimal for solving the item. A longer distance corresponds to less likelihood of a correct response. Tang, Wang, et al. (2020) propose extracting latent variables from process data through multidimensional scaling (MDS). Specifically, this method involves constructing a multidimensional space according to the pairwise dissimilarities between observed sequences. After proper rotation, the coordinates of this space can be treated as features that store process information. Both simulation and empirical studies have demonstrated that the latent variables extracted through MDS retain complete information about problem-solving processes (Tang, Wang, et al., 2020). Although MDS is applied to single items, the latent variables obtained are in standard numerical format, such that the features extracted from multiple items can be incorporated into many well-developed statistical methods to facilitate inferences related to the problem-solving skills of respondents.

In addition to extracting information, researchers have attempted to infer problem-solving ability or relevant latent constructs based on process behaviors. Given the complexity and categorical nature of action sequences, however, only a few such studies have been conducted. Moreover, the models that have been proposed are not well suited to broad use in practice, due to their respective application conditions. For example, Shu et al. (2017) proposed the Markov-IRT model by treating the transitions between observed actions as dichotomous indicators and using multidimensional item response theory (MIRT) models to estimate latent traits. The obtained two latent variables were respectively highly correlated with the systematicity and efficiency scores designed by the test developers, which are only relevant aspects of problem-solving. Although this procedure is generic, it can be applied to only one item at a time, and the number of all possible actions cannot be too many. Liu et al. (2018) developed a modified multilevel mixture IRT (MMixIRT) model that can estimate ability at the level of both process and student, based on process data from only one item. This model requires recoding process steps into dichotomous data according to the only correct solution, which can be difficult for some items. Lamar (2018) integrates a cognitive theory and the IRT approach based on the Markov decision process to develop a Markov decision process measurement model (MDP-MM) for estimating individual capability to solve a problem optimally. The model assumes that an individual's decision-making (i.e., probability of taking an action) in the current state depends on that individual's ability and the expected rewards obtained by taking a given action. Rooted in cognitive theory, this model can be regarded as a top-down approach, but its estimation depends on the extent to which the specification of the cognitive model deviates from reality (Lamar, 2018). The complexity of human behaviors often results in differences between predefined models and the processes performed by individuals. Although the MDP-MM can be updated iteratively based on these differences, doing so would require considerable effort, and it would be difficult for practitioners. This model further requires the definition of problem states in which the actions are taken—a process that can be difficult to realize for some problems.

Figure 1
 An example item in the PIAAC PSTRE assessment
 [Color figure can be viewed at wileyonlinelibrary.com]



Note. Available at www.oecd.org/skills/piaac/Problem%20Solving%20in%20IRE%20Sample%20Items.pdf.

The Present Study

Stadler et al. (2020) have revealed that individual differences in test-taking behavior sequences indicate differences in problem-solving ability, despite the same scores. In other words, process data contain more information about individuals' problem-solving proficiency than item scores. However, it is difficult to construct measurement models directly based on process data, due to the characteristics of process data and the unclear relationship between process information and problem-solving ability. For these reasons, we consider combining the information extraction of process data with the traditional measurement model based on item outcomes, so as to use process information to improve the estimation of problem-solving ability.

In psychological and educational assessments, supplementary prior information that is related to the observable outcome variables and latent variables in the measurement model can be added into these models in order to increase the accuracy of estimation, which is realized through Bayesian methods. For example, some background variables of students (e.g., demographic variables, scores on other tests) have been used as prior information to enhance the accuracy of their ability estimates (Matteucci & Veldkamp, 2013). Similarly, He, Veldkamp, et al. (2019) included the score from the textual assessment as input for a prior distribution of the latent trait measured according to questionnaire items using an IRT model, thereby enhancing accuracy in the detection of posttraumatic stress disorder (PTSD).

Since the process data are related to problem-solving item outcomes and individual problem-solving ability and can provide additional information to latent ability beyond item scores, in this study, we propose incorporating process data as prior information to improve the IRT estimation of the latent trait and focus on the impact of including process information on the estimation of latent traits, as compared to using item scores alone. More specifically, we focus on the extent to which the proficiency estimates obtained from the combination model can reflect the information of the problem-solving process. To extract process information, the multidimensional scaling approach was used. The proposed method was applied to process data of the United States sample on seven PSTRE items in PIAAC 2012.

Materials and Method

Instrument

The PSTRE assessment included a total of 14 items and it was administrated in two booklets. In this study, we focused on the seven items in the second booklet. An example of PSTRE items in the PIAAC is displayed in Figure 1. This item involves only one environment: the web. To solve the item, respondents need to click each link on the result page (displayed in Figure 1) and the associated pages and examine whether the site meets the requirements given in the left panel. Respondents can navigate using the back and forward arrows or the home icon in the toolbar, and they can bookmark websites

or manage bookmarks either by using the bookmark icon in the toolbar or by going through the bookmark menu item.

Data

The data used in this study include process data and item scores of seven PSTRE items. The process records used have been preprocessed from the raw PIAAC 2012 log file data, during which only records representing respondents' actions were kept and recoded, while other records were removed (such as system events activated by respondent's actions). A fragment of the process data for an item and the corresponding recoded action sequence are shown in Appendix Table A1. For the final responses, seven items were scored either dichotomously or polytomously.

Sample

This study focused on the US sample of participants who completed the second PSTRE booklet, including 1355 test-takers. Of these participants, 630 were male and 713 were female. For 125 test-takers, the highest level of education was lower than secondary school; 534 had secondary school as the highest level of education; and 682 had completed higher education. For the rest of the respondents, there were no relevant records for gender and education.

The shortest action sequence observed was "Start, Next, Next_OK," meaning that the respondent skipped to the next item immediately after starting and did not interact with the task, resulting in incorrect item outcomes. Because it is difficult to distinguish whether this was due to low engagement or insufficient proficiency and this study does not focus on how to address these cases, we regarded such process sequences as exhibiting the "Nonresponse (NR)" pattern and excluded them from the analysis. The corresponding outcomes were also coded as missing. When extracting process information, we used the process data of respondents who did not exhibit an NR sequence pattern for each item. Therefore, the sample sizes used for the seven items were different, ranging from 1127 to 1271 (listed in Appendix Table A2). When estimating ability based on data from seven items, we used data on the final 938 respondents whose process data did not reflect any NR sequence patterns.

The Combination Model of IRT and Process Information

In this section, we introduce our proposed approach: the combination model of IRT and process information. This approach involves two stages: (1) extracting features from the process data, and (2) combining process features and IRT within a Bayesian framework.

Stage 1: feature extraction from process data. This stage comprises two steps: (1) feature extraction using multidimensional scaling (MDS) and (2) feature selection based on the random forest algorithm. Tang, Wang, et al. (2020) introduce the MDS approach into the analysis of process data for single items and report that the extracted features retained sufficient information from the problem-solving process. In this study, therefore, we apply the MDS approach to extract continuous variables representing information from the process sequence for each item.

Briefly, the MDS procedure involves constructing a latent space with K raw latent dimensions based on the

dissimilarities between each pair of sequences, which requires the prespecification of K . For seeking interpretations, Tang, Wang, et al. (2020) further performed principal component analysis for these K dimensions, thus producing K features. Thereafter, cross-validation is adopted to compute the information loss (i.e., the discrepancy between the estimated and true similarities) for each possible K , and the K with the minimum loss (denoted as K_{loss} later) is ultimately selected. In this study, the possible K ranged from 2 to 30 for each of the seven items. This step was implemented directly using the *ProcData* package (Tang, Zhang, et al., 2020) in R. Table 1 lists the number of features determined based on the minimum loss (K_{loss}) for each item.

As shown in Table 1, a total of 171 features were extracted based on minimum loss. Adding all of these features into the prior of the model would result in too many parameters to be estimated for the current sample size. In addition, when K was large, the decreasing trend of the loss value with the increasing K was quite flat. It may thus be unnecessary to retain all of these features. Consider the fact that each process feature was extracted to account for the remaining dissimilarities between sequences for each item after excluding the part accounted for by the previous process features. In other words, the first feature accounted for the most important difference between sequences, and the second feature accounted for the second most important difference, and so on. Therefore, we retained the first three features for each item, which are expected to contain most of the information about the major differences between respondents' sequences. Based on the measure of information loss in MDS (see Tang, Wang, et al., 2020, for more details), for each item, we calculated an indicator to measure the amount of information retained by the first three features. Its formula is

$$\left(1 - \frac{\sum_{i,j} (d_{ij} - \|\mathbf{x}_i - \mathbf{x}_j\|)^2}{\sum_{i,j} (d_{ij})^2}\right) \times 100\%, 1 \leq i < j \leq n \quad (1)$$

in which $d_{ij} = d(s_i, s_j)$ is the pairwise dissimilarity between response process s_i and s_j , \mathbf{x}_i is the feature vector of s_i (including only the first three features here), and $\|\mathbf{x}_i - \mathbf{x}_j\| = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j)}$. The percentage of retained information for each item is listed in the last column of Table 1, which is higher than or about 90%. Therefore, it can be considered that the three features did represent most of the information about the differences between sequences. However, it should be noted that the number of features extracted for each item is not required to be the same, nor does it have to be three. If any, researchers can use other suitable methods to further select the extracted process features.

Although Tang, Wang, et al. (2020) introduced an approach to seeking possible interpretations for features extracted by MDS, the obtained interpretations and the criteria for validating them were subjective, and the procedure would require considerable effort. We did not devote much attention to labelling these extracted variables, as this was not the focus of the current study. As stated above, a total of 21 variables from process data on seven items were selected for use as prior information for estimating latent ability. The values of these process features were mostly between -0.6 and approximately 0.5 . Descriptive statistics of these variables and their correlations are shown in Appendix Tables A3 and A4.

Table 1**Number of Variables Extracted from Process Data for Seven PIAAC PSTRE Items**

Item	Score Level	N_{action}	K_{loss}	Percentage of Retained Information for the First 3 Features
U19a	0, 1	54	19	90.5%
U19b	0, 1, 2	254	29	92.9%
U07	0, 1	52	17	92.7%
U02	0, 1, 2, 3	142	29	92.4%
U11b	0, 1, 2, 3	359	30	92.8%
U16	0, 1	136	24	87.6%
U23	0, 1, 2, 3	85	23	96.8%
Sum			171	

Note. N_{action} = number of unique actions; K_{loss} = number of features extracted based on the minimum loss. The items are listed in order of administration. We finally retained only the first 3 features for each item, thus 21 features in total.

Stage 2: adding process features into IRT. After obtaining process information in the form of continuous variables, process features and item response modelling were combined within a Bayesian framework. The IRT model used here was the unidimensional generalized partial credit model (GPCM; Muraki, 1992), which is used in item calibration based solely on item outcomes in the PIAAC 2012 (OECD, 2016). For an item j with $m_j + 1$ ordered categories (0, 1, ..., m_j), the probability of the respondent i being in the response category k can be written as follows:

$$P(Y_{ij} = k) = \frac{\exp\left[\sum_{r=0}^k 1.7\alpha_j(\theta_i - \beta_j + d_{jr})\right]}{\sum_{u=0}^{m_j} \exp\left[\sum_{r=0}^u 1.7\alpha_j(\theta_i - \beta_j + d_{jr})\right]}, \quad (2)$$

where θ_i is the latent ability of respondent i ; α_j is the slope parameter of item j (i.e., the item discrimination parameter); β_j is the location parameter (i.e., item difficulty); and d_{jr} is the category threshold parameter.

The next step in our approach involved using the 21 variables obtained from process data in the previous step as prior information for the latent ability θ_i . The relationship between the latent ability θ of individual i , and the process information $x_1 \sim x_{21}$ is given by the following linear regression structure:

$$\theta_i = b_0 + b_1x_{1i} + \dots + b_{21}x_{21i} + \varepsilon_i, \quad (3)$$

where b_0 and $b_1 \sim b_{21}$ are the regression intercept and slopes, and ε_i is the error term that is assumed to follow a normal distribution $N(0, \sigma^2)$ with $i = 1, 2, \dots, N$ individuals. In other words, given the process variables, the prior distribution of θ_i is as follows:

$$\theta_i \sim N(b_0 + b_1x_{1i} + \dots + b_{21}x_{21i}, \sigma^2). \quad (4)$$

The regression parameters in the prior structure (b_0, b_1, \dots, b_{21} , and ε_i) are assumed to be independent of the item parameters in Equation 2.

Model Comparison and Specification

To examine the performance of the inclusion of process information, we compared the latent ability estimates from the combination model to those from the IRT model based on response outcomes alone. In practice, the Bayesian Expected a Priori (EAP) is often used to estimate the IRT ability scores after the item parameters are obtained. In addition, considering the ideas of the EAP and MCMC algorithms are different,

we considered both EAP and MCMC for the IRT model based on responses (denoted as IRT_EAP and IRT_MCMC), in order to make the ability estimation with and without process information more comparable. In all, we compared three approaches.

The IRT model used in the three approaches was the same GPCM, which is reduced to the two-parameter logistic (2PL) model for binary outcomes. To maintain the comparability of the three sets of estimates, for each of the three approaches, we employed the same set of fixed item parameters that had been calibrated using the GPCM and published in *Technical Report of the Survey of Adult Skills (PIAAC) (2nd Edition)* (OECD, 2016).

The MCMC estimation procedures were run in JAGS 4.3.0 (Plummer, 2017) through the *rjags* package (Plummer, 2019) in R. In the combination model, we specified relatively diffuse prior distributions for the parameters in the regression of θ : $b_0 \sim N(0, 10^2)$, $b_1, \dots, b_{21} \sim N(0, 2^2)$, and $\sigma^2 \sim IG(1, 1)$, since we had no prior knowledge for the distribution of θ or those parameters. For the MCMC estimation without process information, we assigned a common normal prior with relatively large variance, $\theta_i \sim N(0, 3^2)$, to the person parameter (latent ability). The EAP estimation was performed using the *irtplay* package in R (Lim & Wells, 2020), also with the normal prior $N(0, 3^2)$.

The convergence of the MCMC estimation for each parameter was monitored using the potential scale reduction factor \hat{R} (Brooks & Gelman, 1998; Gelman & Rubin, 1992), as well as the trace plot of MCMC. Generally, approximate convergence is diagnosed when \hat{R} is close to 1 (often operationalized as being <1.1). We also used the plot of the evolution of \hat{R} as the number of iterations increases to check whether the \hat{R} happened to be close to 1 by chance or had really converged (Brooks & Gelman, 1998). The estimation of the latent ability in both IRT_MCMC and the combination model was performed using two MCMC chains. Each chain was run for 20,000 iterations. As suggested by Brooks and Gelman (1998), the first half of the iterations can be discarded for each MCMC chain to avoid the burn-in (i.e., running the chain until stationarity is reached; Patz & Junker, 1999) period. Combined with the convergence results, the first 10,000 iterations were discarded for each chain, yielding a total of $10,000 \times 2 = 20,000$ iterations that served to empirically approximate the posterior distribution. We used the median of the distribution as a posterior summary of each parameter.

As an empirical study, the true values of the latent ability are unavailable. To examine the influence of adding prior

information from process data, we conducted two investigations.

Investigation 1

The first investigation aimed at examining the association between ability estimates and several indicators of the problem-solving process that are related to the latent construct. A higher correlation suggests that the set of ability estimates obtained by the corresponding method is more reflective of the characteristics of the response process. This analysis was based on the entire sample of 938 respondents.

Specifically, we adopted the indicators of similarity and efficiency to quantify the problem-solving information across multiple items, as proposed by He, Borgonovi, et al. (2019). These indicators are based on the distances between observed sequences (OS) and the optimal or reference sequences (RS), that is, the length of the longest common subsequence (LCS) between OS and RS. As mentioned in the article of He, Borgonovi, et al. (2019), reference sequences represent the theoretical range of sequences that should be the most efficient way to solve the problem. Therefore, there may be more than one RS for an item. We invited three experts in the field of cognition who are familiar with PIAAC and problem solving to discuss and identify the reference sequences for each PSTRE item. To access the non-released items, they also signed the confidentiality agreement.

The similarity for each item is calculated as the ratio between the length of LCS and the length of RS (Formula 5). The larger the ratio, the more closely the individual follows the reference strategy to solve the problem, thus indicating higher ability. The efficiency for each item is the ratio of the LCS length to the OS length (Formula 6). A large value implies that the individual solved the problem without many unnecessary behaviors, also implying high ability (Stadler et al., 2020). In order to synthesize information across items, we calculated the mean of the similarity (or efficiency) across items, generating two indicators: average similarity (SM) and average efficiency (EM) (Formulas 7 and 8). In addition, we calculated the standard deviation of similarity across items (SSD) to denote the consistency of similarity (Formula 9), with larger values corresponding to inconsistent problem-solving behaviors. For additional details, see He, Borgonovi, et al. (2019).

$$\text{Similarity for each item : Sim} = \text{length (LCS)} / \text{length (RS)} , \quad (5)$$

$$\text{Efficiency for each item : Eff} = \text{length (LCS)} / \text{length (OS)} , \quad (6)$$

$$\text{Average Similarity : SM} = \text{Mean (Sim}_1, \text{Sim}_2, \dots, \text{Sim}_7) , \quad (7)$$

$$\text{Average Efficiency : EM} = \text{Mean (Eff}_1, \text{Eff}_2, \dots, \text{Eff}_7) , \quad (8)$$

$$\text{Consistency of Similarity : SSD} = (\text{Sim}_1, \text{Sim}_2, \dots, \text{Sim}_7) . \quad (9)$$

Given that each item may have more than one optimal sequence, an observed sequence was compared to each optimal sequence, thus generating multiple LCSs. The longest LCS and the corresponding reference sequence are used in the calculation. In our research, the presence of two or more longest LCSs for a single observed sequence indicates that the

observed sequence might not contain some key operations that could match the corresponding reference sequences. In these cases, we selected the length of the longest reference sequences that produce the longest LCS as the denominator of the similarity. More specifically, we chose to generate a smaller similarity to reduce the information contained in the value of this indicator. In addition, when computing the length of LCS for each item, three actions were excluded from both observed and reference sequences: START at the beginning, and NEXT and NEXT_OK at the end of each sequence. These actions were excluded, as they were required by each item and did not contain any useful information. After obtaining the process indicators for all respondents, we calculated the correlation between the latent trait estimates and the indicators.

Results

Table 2 presents descriptive statistics and correlations among ability estimates from three models. Logically, the IRT_EAP and IRT_MCMC approaches produced highly consistent estimates with a correlation of 0.983 and a wider range. The correlation between estimates of the combination model and those without process priors appeared to be slightly lower, indicating that the inclusion of process priors led to some differences in ability estimates.

In statistics, the precision of a parameter estimate is measured by the variability of the estimates around the value of the parameter (Baker, 2001). More about the parameter value could be known if the corresponding estimate is obtained with higher precision. Accordingly, the *SDs* were computed according to the posterior draws for the MCMC estimation (IRT_MCMC and the combination model). Precision was measured by the *SE* for the IRT model with EAP estimation. The smallest average posterior SD of the combination model indicates that the addition of the process priors helped to improve the precision of the estimates.

The distributions of three process indicators (SM, EM, and SSD) are presented in Appendix Figure A1. We expected that the estimates produced by the combination model would contain more information about the process performance than those of the outcome-based IRT models, which should be reflected by a higher correlation between estimates of the combination model and the process indicators.

The Spearman's correlation was used to explore the relationship, thereby avoiding the influence of extreme values in estimates. We first calculated the Spearman's correlations between SM/EM and the three sets of estimates. Given the finding of He, Borgonovi, et al. (2019) that the relationship between the consistency of similarity and problem-solving proficiency differed by level of average similarity, we decided not to consider SSD here and to examine its relationship with ability estimates later.

In general, the estimates without using process information were already highly correlated with average similarity (0.879), and were moderately correlated with average efficiency (0.663), in which there was no difference between IRT_EAP and IRT_MCMC estimates. The inclusion of process priors enhanced the relationship between ability estimates and SM/EM. Specifically, the correlation of the combination model estimates with the similarity is 0.905 and the correlation with the efficiency is 0.739. These findings imply that the estimates using process information contain more information related to problem-solving strategies and efficiency.

Table 2*Descriptive Statistics and Correlations among Ability Estimates with and without Process Priors*

Approach	M	SD	Range	Average	1	2	3
				Posterior SD or SE			
1. Combination model	-0.040	0.804	[-2.874, 1.772]	0.204	1.000		
2. IRT_EAP	-0.024	1.270	[-3.500, 3.397]	0.467	0.931	1.000	
3. IRT_MCMC	-0.030	1.164	[-3.264, 3.196]	0.519	0.898	0.983	1.000

Note. IRT_EAP = the IRT model with EAP estimation; IRT_MCMC = the IRT model without process information using MCMC estimation.

Table 3*Spearman's Correlations between Process Indicators and Latent Ability Estimates with Process Priors for Score Patterns with More than 15 Respondents*

Score Pattern	Number of Respondents	Ability Estimates	Correlation with Process Indicators	
			SM	EM
0000000	42	-1.856 (0.385)	0.471**	0.325*
1213313	30	1.144 (0.287)	0.288	0.696**
1212313	23	0.972 (0.219)	-0.141	0.291
1212013	20	0.678 (0.194)	0.519*	0.689**
1000000	19	-1.535 (0.316)	0.672**	0.291
1200010	17	-0.398 (0.183)	0.422 [†]	0.213

Note. Numbers in the parentheses are the standard deviations of the ability estimates in each score pattern.

SM = average similarity; EM = average efficiency.

[†] $p < 0.10$.

* $p < 0.05$.

** $p < 0.01$.

Table 4*Spearman's Correlations between Consistency of Similarity (SSD) and Latent Ability Estimates from Three Approaches under Different Levels of Average Similarity*

Approaches	High Similarity	Medium Similarity	Low Similarity
Combination model	-0.361**	-0.188**	0.460**
IRT_EAP	-0.256**	-0.063	0.463**
IRT_MCMC	-0.256**	-0.059	0.461**

Note. Consistency of similarity is calculated as the standard deviation of similarity, with larger values corresponding to lower consistency.

IRT_EAP = the IRT model with EAP estimation; IRT_MCMC = the IRT model without process information using MCMC estimation.

** $p < 0.01$.

We conducted a further examination of the correlations between the estimates with process priors and the average similarity or average efficiency for respondents within the same score pattern (see Table 3). For groups with too few respondents, the correlation may not accurately capture the relationship between variables. For this reason, we selected the score patterns with at least 15 respondents. Because the item scores provided the same information, the estimates with no prior information were/should be the same for individuals having the same score pattern. Therefore, IRT_EAP and IRT_MCMC estimates were not included. According to Table 3, the estimates with process priors were often strongly correlated with similarity and/or efficiency. For the high score patterns ("1213313" and "1212013"), the combination model estimates were strongly correlated with efficiency. Their small to moderate correlations with the similarity may be because their average similarity was already high with a narrow range of variation. The low correlation of estimates with efficiency for the high-score pattern "1212313" could be attributed to the low variance of average efficiency amongst these respondents. This response mode indicates that respondents received partial credit on the fourth item and full

scores on all other items, possibly implying high proficiency. The combination-model estimates for the low score pattern ("1000000" and "1200010") were more strongly correlated with average similarity than they were with average efficiency. For the zero-score group ("0000000"); however, the correlations of the combination model estimates with average similarity or efficiency were both moderate and significant.

The correlations between the consistency of similarity (SSD) and latent ability estimates from three approaches under different levels of the average similarity are listed in Table 4. In general, the association between SSD and ability estimates was not very strong. For respondents with high and low similarity, the direction of this association was opposite. Given that higher SSD values indicate lower consistency, the negative correlation for the high similarity group means that, for this group, respondents performing more consistent behaviors tended to obtain higher estimates. The opposite situation applied for the low similarity group. The influence of including process priors was reflected primarily in respondents with high similarity, for whom the correlation increased from 0.256 to 0.335.

Table 5

Spearman's Correlations between Proficiency Estimates from Three Approaches and Average Efficiency/Similarity under Different Levels of Average Similarity/Efficiency

Approach	Correlation with Average Efficiency		
	High similarity	Medium similarity	Low similarity
Combination model	0.465**	0.262**	0.481**
IRT_EAP	0.196**	0.111	0.382**
IRT_MCMC	0.199**	0.109	0.379**
	Correlation with Average Similarity		
	High efficiency	Medium efficiency	Low efficiency
Combination model	0.742**	0.837**	0.903**
IRT_EAP	0.702**	0.806**	0.847**
IRT_MCMC	0.703**	0.806**	0.846**

IRT_EAP = the IRT model with EAP estimation; IRT_MCMC = the IRT model without process information using MCMC estimation.

** $p < 0.01$.

Logically, the interpretation of the efficiency index depends on whether the respondents adopt the correct problem-solving strategies. We, therefore, divided the average similarity and efficiency values into low, medium, and high levels, based on the percentiles 33.3 and 66.7. We then investigated the association between estimates and average similarity (efficiency) at different levels of average efficiency (similarity) (see Table 5). According to these results, the correlation between efficiency and estimates under each level of similarity was much lower than the correlation for the whole sample. In addition, for different levels of similarity (or efficiency), the estimates reflected efficiency (or similarity) to differing degrees, while the inclusion of process information enhanced the degree of reflection in all cases. Of these results, the effect of the combination model was the most salient at the high similarity level. More specifically, for respondents with high average similarity, the estimates of the combination model were able to reflect much more information related to their solution efficiency.

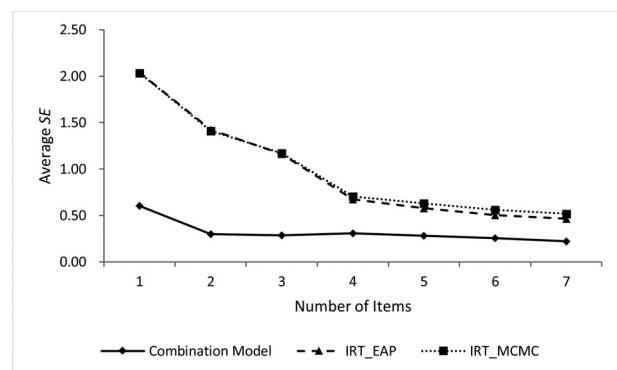
Investigation 2

The second investigation aimed to explore the efficiency of the combination model. Given that process information can be regarded as supplementary information for estimating problem-solving proficiency, we expected that adding process information would improve the precision of the latent variable estimates, thereby helping to reduce the number of items without compromising the precision of the estimation. Therefore, we fitted the combination model, as well as the two IRT models based on only item outcomes, to the data of different numbers of items. We expected that the combination model would produce more precise estimates with fewer items than the models without process priors. The precision of θ -estimates was indicated by the posterior standard deviation (*SD*) in the MCMC estimation and the standard error (*SE*) in EAP estimation.

Due to the unavailability of the prior knowledge for the regression parameters in the combination model, we randomly split the sample into two halves and fitted the combination model with large-variance priors ($b_0 \sim N(0, 10^2)$, $b_1, \dots, b_{21} \sim N(0, 2^2)$, and $\sigma^2 \sim IG(1, 1)$) to the first half of the sample. We then used the obtained posteriors as priors and fitted the combination model to the remaining half of the sample. The other two IRT models were also fitted to the same second half of the sample, using the same priors as in the first investigation. These procedures were repeated for

Figure 2

Relationship between precision of proficiency estimates and number of items with or without process priors



IRT_EAP = the IRT model with EAP estimation; IRT_MCMC = the IRT model without process information using MCMC estimation.

data of different numbers of items. For example, when one item (U19a) was used, only three process variables from the process data of item U19a were included in the prior for the combination model. By checking the estimation precision of three approaches when different numbers of items were used, the effect of process information in improving estimation precision and thus reducing test length could be better shown. Given that there appear to be no criteria for selecting items, the items were ranked in the order in which they had been administered in the assessment. When using the combination model, the priors of the parameters in the regression of θ for the second half of the sample (i.e., the posteriors obtained from the first half of the sample) when one to seven items were used are shown in Appendix Table A5.

Note that the two investigations were based on different prior specifications used in the combination model, that is, the large-variance priors with no prior information and informative priors from half of the sample. To check the sensitivity of the results to priors, we also made a brief comparison between the ability estimates for the same subsample resulting from the two sets of priors.

Results

As shown in Figure 2, IRT_EAP and IRT_MCMC had highly similar estimation precision in all situations: the SE or posterior SD of estimation without process priors started

at about 2 (when only the first item was included) and dropped to about 0.5 (when all seven items were included). By comparison, the posterior SD of the combination model stayed at the lowest level and was much less affected by the number of items, decreasing from 0.53 to 0.21 as the number of items increased. The estimation precision of the combination model using only one item was close to that of the IRT models based on item scores of seven items. Further, with more items, the difference between the estimation precision of the three approaches decreased at the beginning, but then remained relatively stable after reaching four items.

We investigated the standard error of estimation for respondents at different proficiency levels in greater detail. The proficiency levels were divided based on the 20th, 40th, 60th, and 80th percentiles of the IRT_MCMC estimates, given the very high correlation between ability estimates. Proficiency levels 1~5 correspond to very low, low, medium, high, and very high levels, respectively. From Figure 3, the decreasing trend of the standard errors without using process information differed between different groups. When fewer than four items were used, the standard errors obtained from IRT_MCMC and IRT_EAP were high for all proficiency levels. When 4~7 items were used, the standard errors obtained without process information were higher for the very low and very high proficiency group than they were for the other three proficiency groups. In contrast, the standard errors from the combination model were not affected by proficiency levels and remained the lowest in the comparison between approaches.

These results suggest that the inclusion of process information improved the precision of latent trait estimation, with the effect being more apparent when using fewer items and for respondents with very low and very high levels of proficiency.

Considering that two sets of prior specifications (i.e., the large-variance priors and the posterior-informed priors) were used respectively for two investigations, we compared the θ -estimates of the second half sample that resulted from them, in order to check the influence of these priors on the results. The differences between the two sets of estimates fell between -0.067 and 0.071 , and the differences between the two sets of posterior SDs were mostly between 0.010 and 0.032 . These results show that the ability estimation was not sensitive to the priors for the regression coefficients of the process variables.

Discussion

Summary and Implications

Process data record rich details about how respondents solve problems while providing additional information related to problem-solving ability beyond item outcomes. However, the construction of measure models directly based on the behavioral sequences is a big challenge, considering the lack of definition of how process data are related to item outcomes and latent ability, as well as the possible dependency between process variables. Therefore, starting from the other perspective, this study is a preliminary attempt to incorporate data-driven process information into the IRT measurement model in order to improve the estimation of problem-solving ability. Specifically, information from process sequences is extracted using MDS and used as prior information for the estimation of latent ability based on item outcomes. Because process in-

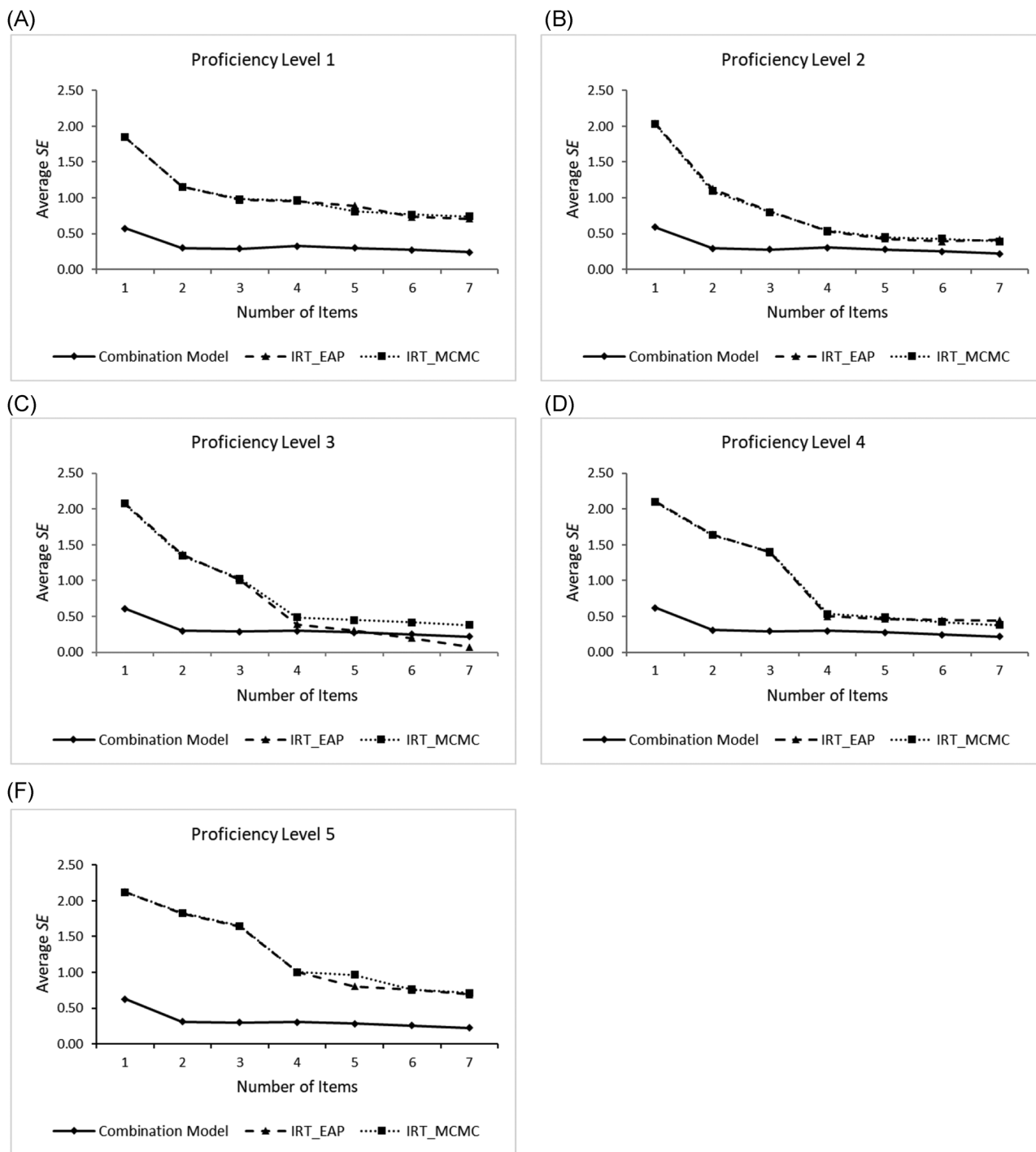
formation provides additional information about latent ability beyond item outcomes, more informative and precise ability estimates can be obtained using the combination model. The whole procedure does not require prior knowledge of the items or response processes.

To examine the impact of including process information into the measurement model, we compared the ability estimates with and without using process information, based on their correlations with process indicators, along with the precision for the estimates. The process indicators used in this study—similarity and efficiency—are important manifestations of problem-solving ability. The similarity indicator describes the consistency between the observed action sequence and the optimal solution, which can be regarded as an indicator of strategy use (He, Borgonovi, et al., 2019). The efficiency of different sequences also indicates differences in problem-solving ability (Stadler et al., 2020).

Results based on data from the US sample on seven PSTRE items in the PIAAC indicate that the inclusion of process priors renders the ability estimates more informative, reflected by a stronger correlation with similarity and/or efficiency. Examining the correlation between the estimates and one process indicator at different levels of another process indicator reveals the process information contained in the estimates with process priors from a more detailed perspective. The inclusion of process priors resulted in a larger increase in the correlation between the estimates and the consistency of similarity for respondents with high similarity than it did for those with medium or low similarity. It is conceivable that respondents who consistently perform sequences that are close to optimal solutions tend to have higher abilities than do those exhibiting performance that is high, on average, but unstable. Process information also caused a much higher increase in the correlation between the estimates and the average efficiency of respondents with high similarity than it did for those with medium or low similarity. This result is meaningful for the high similarity group, as the sequences with high similarity are more similar to each other. For these respondents, efficiency can provide a better reflection of differences in ability. The weaker effect for the medium or low similarity group may have occurred because sequences with medium or low similarity exhibit much more variation in other aspects, in addition to similarity and efficiency. The efficiency indicator may not be able to capture the reflection of process information for these sequences, especially for those with low similarity. In addition, low similarity indicates that, in general, the respondents did not use appropriate solutions. In this regard, efficiency may not provide a good reflection of differences in ability.

The higher estimation precision of the combination model also indicates the positive impact of using process priors. When process priors were introduced, even one item was sufficient to make latent trait estimations as precise as those obtained using only the outcomes of all seven items. This result is similar to findings reported in He, Veldkamp, et al. (2019), in which the textual assessment information that was used as prior information increased the precision of PTSD latent trait estimates based on IRT, thereby eliminating the need for the follow-up items. Likewise, the findings of our study suggest that the use of process priors can help to shorten test length, improving the efficiency of test development and implementation. In addition, decreases in the standard errors were more apparent for respondents of very low and very high ability levels. This indicates that the use of

Figure 3
 Relationship between precision of proficiency estimates and number of items for different proficiency levels



Note. Panels A to E show the changes of average standard errors with increasing number of items for Proficiency Levels 1–5, which correspond to very low, low, medium, high, and very high levels, respectively. These levels were divided based on the 20th, 40th, 60th, and 80th percentiles of the IRT_MCMC estimates. IRT_EAP = the IRT model with EAP estimation; IRT_MCMC = the IRT model without process information using MCMC estimation.

process priors could help to address the problem of low estimation precision associated with the IRT model for respondents at both ends of the ability scale.

In the proposed approach, process information was included into the priors of θ through the Bayesian approach, thus requiring to specify the priors of the regression coefficients. In this study, we used both diffuse priors and posterior-informed priors and found that the informativeness of the

priors of the regression coefficients had little effect on the latent trait estimation. In addition, as extracted and selected by data-driven approaches, the process variables of different samples may be different and difficult to match. If researchers want to get the prior information of those coefficients of process variables, they need to derive the process variables based on the whole sample first, so as to keep these variables consistent between the split-half samples.

Therefore, considering the insensitivity of estimation to the priors and the efficiency of implementation, large-variance priors may be a better choice for the combination model.

One thing that may cause confusion is that the ranges of ability estimates for the three models are different (see Table 2). The range of estimates for the combination model is smaller than those for the other two models. This is probably due to the different priors of the latent ability. In the IRT_MCMC and IRT_EAP models, a weak-informative prior $N(0, 3^2)$ was specified for the latent ability, while an informative prior based on the process features was used in the combination model. According to the value ranges of the process features, it can be easily inferred that the prior variance is much smaller than 3^2 , leading to a smaller range of the estimates. Actually, the estimated abilities of individuals resulting from different estimation procedures should not be directly compared. It may be more appropriate to focus on the relative positions of individuals based on the estimates. For this reason, we used correlation in the first investigation to make comparison between methods.

Limitations and Future Directions

First, this combination model is only a preliminary attempt to include process information in the estimation of latent traits for problem-solving items. The ability estimates from the combination model need further validation by using other external criteria and more process indicators related to problem-solving skills.

Second, in the current study, we used the precalibrated item parameters, referring to the study of He, Veldkamp, et al. (2019) in which the information from the textual assessment was used as prior information in the item response modelling for the questionnaire. In our approach, however, it is not necessary to fix item parameters. Future studies could investigate the impact of fixing or freely estimating item parameters.

Third, the selection of the extracted process features and the construction of the regression-like prior structure need to be further explored. In this study, we directly used the first three features for each item, which retained most of the information about the major differences between respondents' sequences. However, there are other options for choosing the number of features to take into account. For example, the selection of features based on the contribution of each feature can be considered. Also, as suggested by Levy and Mislevy (2016), the covariates in the regression-like structure of the latent variable should be collateral information related to the trait captured by the latent variable. This means that additional information reflecting respondents' latent traits may be needed to select the process features. However, if this is the case, the test efficiency improved by the use of process information may be greatly reduced because of the need to collect additional information. Therefore, how to select the process features to be used as prior information remains to be studied in detail. Besides, given that the relationship between process data and ability is not yet known, we adopted a simple linear form of adding process information in this study. The actual relationship is likely to be more complicated. These may partially explain the limited increase in the correlation between estimates and the process indicators due to the use of process priors. Further research could consider other methods to use process variables as prior information and focus on achieving further improvements in the impact of using process information, for example, by constructing more complex relationships


between process information and the latent ability, as well as by using more information from the process data.


Acknowledgments

The authors thank the OECD-PIAAC (Programme for the International Assessment of Adult Competencies) team for granting access to the data source in this study.

Orcid

Yue Xiao  <https://orcid.org/0000-0002-2836-2906>

Bernard Veldkamp  <https://orcid.org/0000-0003-3543-2164>

Hongyun Liu  <https://orcid.org/0000-0002-3472-9102>

Author Details

Correspondence concerning this article should be addressed to Bernard P Veldkamp, Department of Learning, Data Analytics and Technology, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands. Email: b.p.veldkamp@utwente.nl. Yue Xiao is affiliated with the Faculty of Psychology, Beijing Normal University, Beijing, China; Bernard Veldkamp is affiliated with the Department of Learning, Data Analytics and Technology, University of Twente, Enschede, The Netherlands; and Hongyun Liu is affiliated with Beijing Key Laboratory of Applied Experimental Psychology, Faculty of Psychology, Beijing Normal University, Beijing, China.

Conflict of Interest

We have no known conflict of interest to disclose.

References

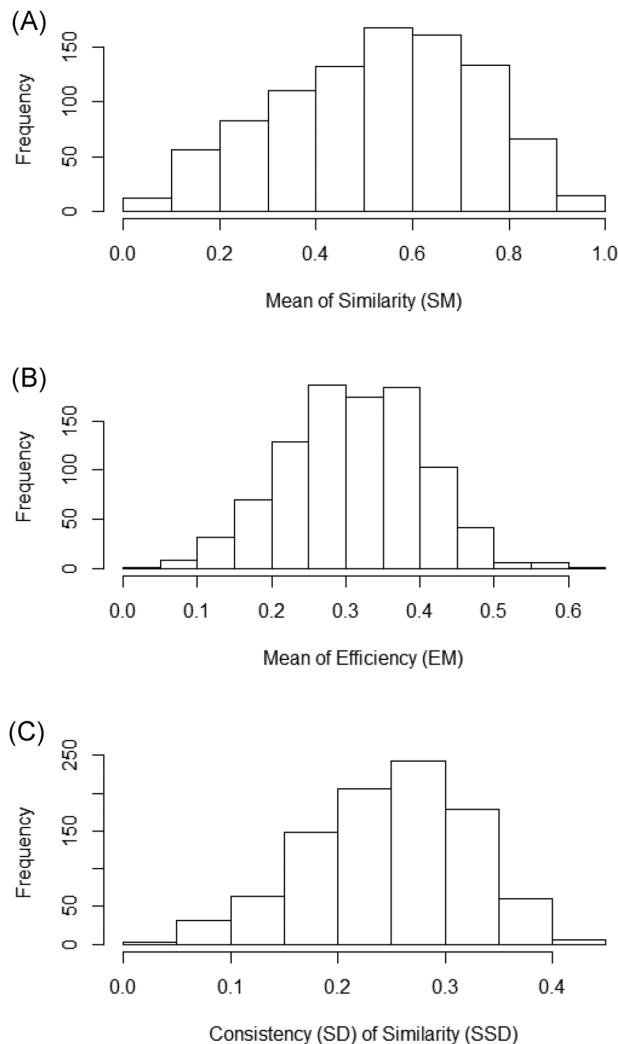
- Arieli-Attali, M., Ou, L., & Simmering, V. R. (2019). Understanding test takers' choices in a self-adapted test: A hidden Markov modeling of process data. *Frontiers in Psychology, 10*, Article 83. <https://doi.org/10.3389/fpsyg.2019.00083>
- Baker, F. B. (2001). The basics of item response theory. *ERIC Clearinghouse on Assessment and Evaluation*, <https://files.eric.ed.gov/fulltext/ED458219.pdf>
- Biswas, G., Jeong, H., Kinnebrew, J. S., Sulcer, B., & Roscoe, R. O. D. (2010). Measuring self-regulated learning skills through social interactions in a teachable agent environment. *Research and Practice in Technology Enhanced Learning, 5*(02), 123–152. <https://doi.org/10.1142/S1793206810000839>
- Brooks, S. P., & Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics, 7*, 434–455. <https://doi.org/10.1080/10618600.1998.10474787>
- Care, E., Griffin, P., & McGaw, B. (2012). *Assessment and teaching of 21st century skills*. Springer.
- Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science, 7*(4), 457–472. <https://doi.org/10.1214/ss/1177011136>
- Goldhammer, F., Naumann, J., & Keßel, Y. (2013). Assessing individual differences in basic computer skills. *European Journal of Psychological Assessment, 29*(4), 263–275. <https://doi.org/10.1027/1015-5759/a000153>
- Greiff, S., Wüstenberg, S., & Avvisati, F. (2015). Computer-generated log-file analyses as a window into students' minds? A showcase study based on the PISA 2012 assessment of problem solving. *Computers & Education, 91*, 92–105. <https://doi.org/10.1016/j.compedu.2015.10.018>

- Guo, H., Rios, J. A., Haberman, S., Liu, O. L., Wang, J., & Paek, I. (2016). A new procedure for detection of students' rapid guessing responses using response time. *Applied Measurement in Education, 29*, 173–183. <https://doi.org/10.1080/08957347.2016.1171766>
- Han, Z., He, Q., & von Davier, M. (2019). Predictive feature generation and selection using process data from PISA simulation-based environment: An application of tree-based ensemble methods. *Frontiers in Psychology, 10*, Article 2461. <https://doi.org/10.3389/fpsyg.2019.02461>
- He, Q., Borgonovi, F., & Paccagnella, M. (2019). *Using process data to understand adults' problem-solving behaviour in the programme for the international assessment of adult competencies (PIAAC): Identifying generalised patterns across multiple tasks with sequence mining* (OECD Education Working Papers, No. 205). OECD Publishing. <https://doi.org/10.1787/650918f2-en>
- He, Q., & von Davier, M. (2016). Analyzing process data from problem-solving items with n-grams: Insights from a computer-based large-scale assessment. In Y. Rosen, S. Ferrara, & M. Mosharraf (Eds.), *Handbook of research on technology tools for real-world skill development* (pp. 750–777). IGI Global.
- He, Q., Veldkamp, B. P., Glas, C. A., & van den Berg, S. M. (2019). Combining text mining of long constructed responses and item-based measures: A hybrid test design to screen for posttraumatic stress disorder (PTSD). *Frontiers in Psychology, 10*, Article 2358. <https://doi.org/10.3389/fpsyg.2019.02358>
- LaMar, M. M. (2018). Markov decision process measurement model. *Psychometrika, 83*(1), 67–88. <https://doi.org/10.1007/s11336-017-9570-0>
- Lee, Y. H., & Haberman, S. J. (2016). Investigating test-taking behaviors using timing and process data. *International Journal of Testing, 16*(3), 240–267. <https://doi.org/10.1080/15305058.2015.1085385>
- Levy, R., & Mislevy, R. J. (2016). *Bayesian psychometric modeling*. CRC Press.
- Lim, H., & Wells, C. S. (2020). irtplay: an r package for online item calibration, scoring, evaluation of model fit, and useful functions for unidimensional IRT. *Applied Psychological Measurement, 44*(7–8), 563–565. <https://doi.org/10.1177/0146621620921247>
- Liu, H., Liu, Y., & Li, M. (2018). Analysis of process data of PISA 2012 computer-based problem solving: Application of the modified Multilevel Mixture IRT model. *Frontiers in Psychology, 9*, Article 1372. <https://doi.org/10.3389/fpsyg.2018.01372>
- Matteucci, M., & Veldkamp, B. P. (2013). On the use of MCMC computerized adaptive testing with empirical prior information to improve efficiency. *Statistical Methods & Applications, 22*(2), 243–267. <http://doi.org/10.1007/s10260-012-0216-1>
- Muraki, E. (1992). A generalized partial credit model: Application of an EM algorithm. *Applied Psychological Measurement, 16*(2), 159–176. <https://doi.org/10.1177/014662169201600206>
- National Research Council. (2012). *Education for life and work: Developing transferable knowledge and skills in the 21st century*. National Academies Press.
- Newell, A., & Simon, H. A. (1972). *Human problem solving* (Vol., 104). Prentice-Hall.
- OECD. (2013). *PISA 2012 assessment and analytical framework: Mathematics, reading, science, problem solving and financial literacy*. OECD Publishing. <http://doi.org/10.1787/9789264190511-en>
- OECD. (2016). *Technical report of the survey of adult skills (PIAAC) (2nd Edition)*. OECD Publishing. https://www.oecd.org/skills/piaac/PIAAC_Technical_Report_2nd_Edition_Full_Report.pdf
- Patz, R., & Junker, B. (1999). A straightforward approach to Markov chain Monte Carlo methods for item response models. *Journal of Educational and Behavioral Statistics, 24*(2), 146–178.
- Plummer, M. (2017). *JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling* (Version 4.3.0) [Computer software]. <https://sourceforge.net/projects/mcmc-jags/files/JAGS>
- Plummer, M. (2019). rjags: Bayesian Graphical Models using MCMC. R package version 4–9. <http://CRAN.R-project.org/package=rjags>
- Schleicher, A. (2008). PIAAC: A new strategy for assessing adult competencies. *International Review of Education, 54*(5–6), 627–650. <https://doi.org/10.1007/s11159-008-9105-0>
- Shu, Z., Bergner, Y., Zhu, M., Hao, J., & von, & Davier, A. A. (2017). An item response theory analysis of problem-solving processes in scenario-based tasks. *Psychological Test and Assessment Modeling, 59*(1), 109–131.
- Stadler, M., Hofer, S., & Greiff, S. (2020). First among equals: Log data indicates ability differences despite equal scores. *Computers in Human Behavior, 111*, Article 106442. <https://doi.org/10.1016/j.chb.2020.106442>
- Tang, X., Wang, Z., He, Q., Liu, J., & Ying, Z. (2020). Latent feature extraction for process data via multidimensional scaling. *Psychometrika, 85*, 378–397. <https://doi.org/10.1007/s11336-020-09708-3>
- Tang, X., Zhang, S., Wang, Z., Liu, J., & Ying, Z. (2020). *ProcData: An R package for process data analysis*. arXiv. <https://arxiv.org/abs/2006.05061>
- van der Linden, W. J., Klein Entink, R. H., & Fox, J. P. (2010). IRT parameter estimation with response times as collateral information. *Applied Psychological Measurement, 34*(5), 327–347. <http://doi.org/10.1177/0146621609349800>
- Wang, C., & Xu, G. (2015). A mixture hierarchical model for response times and response accuracy. *British Journal of Mathematical and Statistical Psychology, 68*(3), 456–477. <https://doi.org/10.1111/bmsp.12054>
- Zhu, M., Shu, Z., & von Davier, A. A. (2016). Using networks to visualize and analyze process data for educational assessment. *Journal of Educational Measurement, 53*(2), 190–211. <https://doi.org/10.1111/jedm.12107>

Appendix A.

Figure A1

Histograms of process indicators



Panel A: Histogram of the average similarity. Panel B: Histogram of the average efficiency. Panel C: Histogram of the consistency of similarity.

Appendix B.

The Main R Code to Implement Approaches Used in This Study

```
#####
#####
##### To extract features from response processes of an
item by MDS #####
#####
#####
# Item1.csv is the process data file of an item, including
two columns: "SEQID" indicating respondents' ID, and "se-
quence" that contains the recoded action sequences of all re-
spondents.
# So in the data file, each row indicates a recoded action
of a respondent.
library(ProcData)
seqs <- read.seqs(file = "Item1.csv", style = "multiple",
id_var = "SEQID", action_var = "sequence")
```

```
K_res <- chooseK_mds(seqs, K_cand = 3:30, return_dist
= T) # extracting 3 to 30 features from the process of this
item
```

```
K <- K_res$K # the number of process variables selected
based on the minimum loss
```

```
lv_mat <- seq2feature_mds(K_res$dist_mat,
K_res$K)$theta # the numeric matrix giving the K extracted
features
```

```
#####
```

```
#####
```

```
##### To fit the IRT model through the irtplay package
```

```
#####
```

```
#####
```

```
#####
```

```
library(irtplay)
```

```
item_para <- read.csv("item_para.csv") # read the file of
item parameters of GPCM, including six columns: category,
alpha, beta, t1, t2, and t3. The column "category" defines the
number of response categories in each item. The remaining
columns are the item parameters.
```

```
# Because the formula of GPCM used in the PIAAC tech-
nical report is a little different from that defined in irtplay
package, the step parameters given by PIAAC technical
report needs to be transformed before they can be used in irt-
play package.
```

```
# Items 1,3,6,7 are polytomously scored, and had step pa-
rameters.
```

```
d_1 <- as.numeric(item_para$beta[1] - item_para[1,
c("t1", "t2", "t3")])
```

```
d_3 <- as.numeric(item_para$beta[3] - item_para[3,
c("t1", "t2", "t3")])
```

```
d_6 <- as.numeric(item_para$beta[6] - item_para[6,
c("t1", "t2")])
```

```
d_7 <- as.numeric(item_para$beta[7] - item_para[7,
c("t1", "t2", "t3")])
```

```
x <- shape_df(par.dc = list(a =
item_para$alpha[c(2,4,5)], b = item_para$beta[c(2,4,5)], g
= NULL), par.py = list(a = item_para$alpha[c(1,3,6,7)],
d = list(d_1,d_3,d_6,d_7)), item.id =
item_para$item, cats = item_para$category, model =
c("GPCM", "2PLM", "GPCM", "2PLM", "2PLM", "GPCM", "GPCM"))
# create a dataframe which includes item meta data (e.g.,
item parameter, categories, models) to be used for the IRT
model-data fit analysis.
```

```
# Conduct theta estimation by EAP
```

```
lv <- est_score(x, data = resp, D = 1.7, method = "EAP",
range = c(-12,12), norm.prior = c(0,3), se = T) # resp is the
matrix storing item responses, each col is an item, each row
indicate a respondent
```

```
#####
```

```
#####
```

```
##### To fit the IRT model through the Rjags package
```

```
#####
```

```
#####
```

```
#####
```

```
library(rjags)
```

```
library(MCMCvis)
```

```
item_para <- read.csv("item_para.csv") # read the file
of item parameters of GPCM, including five columns: alpha,
beta, t1, t2, and t3
```

```
t0 <- rep(0,7)
```

```
d <- cbind(t0, item_para[,c("t1", "t2", "t3")])
```

```
d[is.na(d$t1), "t1"] <- 0
```

Table A1
A Fragment of Raw and Recorded Process Data for Item U19a in PIAAC

ID	SEQID	Item_id	Event_name	Event_type	Timestamp	Event_description	Recorded Actions
1937834	4	1	taoPIAAC	START	0	TEST_TIME = 341	START
1937835	4	1	Stimulus	CELL_CHANGE	44987	id = content_spreadsheet_ColaD_row65	CELL_CHANGE
1937836	4	1	Stimulus	TOOLBAR	52187	id = spreadApp	
1937837	4	1	Stimulus	ENVIRONMENT	52187	environment = SS	SS
1937838	4	1	Stimulus	DOACTION	52187	action = as://customFeedTrace(ENVIRONMENT,environment,SS)	
1937839	4	1	Stimulus	DOACTION	52188	action = as://resetScroll(mailerzone,spreadsheetzone)	
1937840	4	1	Stimulus	DOACTION	52188	action = as://hide(mailerzone)	
1937841	4	1	Stimulus	DOACTION	52188	action = as://show(spreadsheetzone)	
1937842	4	1	Stimulus	TOOLBAR	53019	id = spreadApp	
1937843	4	1	Stimulus	ENVIRONMENT	53019	environment = SS	SS
1937844	4	1	Stimulus	DOACTION	53033	action = as://customFeedTrace(ENVIRONMENT,environment,SS)	
1937845	4	1	Stimulus	DOACTION	53033	action =	
1937846	4	1	Stimulus	DOACTION	53034	as://resetScroll(mailerzone,spreadsheetzone)	
1937847	4	1	Stimulus	DOACTION	53034	action = as://hide(mailerzone)	
1937848	4	1	Stimulus	TOOLBAR	54197	action = as://show(spreadsheetzone)	
1937849	4	1	Stimulus	ENVIRONMENT	54198	id = spreadApp	
1937850	4	1	Stimulus	DOACTION	54198	environment = SS	SS
1937851	4	1	Stimulus	DOACTION	54198	action = as://customFeedTrace(ENVIRONMENT,environment,SS)	
1937852	4	1	Stimulus	DOACTION	54198	action =	
1937853	4	1	Stimulus	DOACTION	54199	as://resetScroll(mailerzone,spreadsheetzone)	
					54199	action = as://hide(mailerzone)	
						action = as://show(spreadsheetzone)	

Note. This is a fragment of the process data for Item U19a, in which the records from the third to the eighth rows are associated with a single interaction from the respondent (i.e., the test-taker clicked a toolbar button to switch to the spreadsheet environment). "ID" indicates the order of the current action; "SEQ_ID" and "Item_id" represent test takers and items, respectively; "Event_name," "Event_type," and "Event_description" indicate the type and details of the current action; "timestamp" indicates the time point of the current action. In this study, we retained only the records on the respondents' interaction events, and recorded them based on the "Event_type" and "Event_description," as well as the item requirement. The recorded actions are shown in the last column. Therefore, the recorded action sequence for this fragment of process data is (START, CELL_CHANGE, SS, SS, SS). This sequence fragment indicates that after the start, the respondent changed the marking state of a row in the spreadsheet environment, but then repeatedly clicked the button that switched to the spreadsheet environment. Data source: The first seven columns in this table were extracted from the log data exported by the LogDataAnalyzer tool, based on the PIAAC Log File Data, both of which can be downloaded at <http://www.oecd.org/skills/piaac/data/piaaclogfiles/>.

Table A2**Sample Sizes Used to Extract Information from Process Data for Each Item**

Item	Original Sample Size ^a	Number of Respondents with NR Sequence Pattern ^b	Sample Size for Extracting Process Variables
U19a	1355	84	1271
U19b	1355	189	1166
U07	1355	119	1236
U02	1355	196	1159
U11b	1353	226	1127
U16	1354	157	1194
U23	1353	125	1228

^aThe original sample size for some items is less than 1355, due to missing records.

^bThe NR sequence pattern represents the nonresponse pattern defined in the current study: "Start, Next, Next_OK."

Table A3**Descriptive Statistics of 21 Process Features Extracted from Process Data of Seven Items**

Features	Min	Max	M	SD	Skewness	Kurtosis
u19a						
u19a_lv1	-0.237	0.366	0.018	0.125	0.276	-0.287
u19a_lv2	-0.280	0.283	-0.007	0.107	-0.320	0.091
u19a_lv3	-0.219	0.281	-0.002	0.092	0.650	0.357
u19b						
u19b_lv1	-0.601	0.263	-0.015	0.168	-1.385	2.247
u19b_lv2	-0.285	0.391	-0.018	0.131	0.688	-0.072
u19b_lv3	-0.435	0.197	0.002	0.113	-0.456	-0.580
u07						
u07_lv1	-5.946	0.239	0.026	0.257	-13.604 (-0.872) ^a	309.840 (-0.439) ^a
u07_lv2	-0.266	0.302	0.009	0.108	0.043	1.336
u07_lv3	-0.167	0.363	-0.001	0.098	1.989	4.521
u02						
u02_lv1	-0.252	0.445	0.032	0.137	-0.237	-0.589
u02_lv2	-0.314	0.219	-0.002	0.069	-1.094	2.361
u02_lv3	-0.209	0.284	0.001	0.058	0.111	3.880
u11b						
u11b_lv1	-0.325	0.384	0.023	0.200	0.101	-1.462
u11b_lv2	-0.300	0.414	0.016	0.166	0.136	-0.524
u11b_lv3	-0.295	0.121	-0.002	0.074	-0.829	0.526
u16						
u16_lv1	-0.337	0.302	-0.013	0.147	0.261	-0.933
u16_lv2	-0.287	0.254	-0.022	0.117	-0.178	-0.545
u16_lv3	-0.321	0.268	-0.005	0.112	-0.385	-0.013
u23						
u23_lv1	-0.462	0.333	0.041	0.167	-0.418	-0.173
u23_lv2	-0.371	0.283	0.004	0.138	0.736	-0.324
u23_lv3	-6.085	0.328	-0.009	0.214	-24.449(0.063) ^b	694.547(2.341) ^b

Note. The results were summarized based on process features of $N = 938$ respondents. The items are listed in order of administration.

^aThe numbers in parentheses are the skewness and kurtosis of the first process variable of U07 calculated after removing the extreme value -5.946.

^bOnly one respondent (ID = 2037) had the extreme value (-5.946) in that variable.

The numbers in parentheses are the skewness and kurtosis of the third process variable of U23 calculated after removing the extreme value -6.085. Only one respondent (ID = 2743) had the extreme value (-6.085) in that variable.

$Y \leftarrow \text{resp} + 1$ # resp is the matrix storing item responses, each col is an item, each row indicate a respondent. The response category in the raw data starts from 0, and is transformed to start from 1 here for the MCMC estimation.

```
##### 1. define the model #####
GPCM <- '
var pai[J,I,4],eta[J,I,4,4],eta_sum[J,I,4],nume[J,I,4],
nume_eta[J,I,4,4];
model{
# likelihood for Y
for (j in 1:J){ # person j
for (i in 1:I){ # item i
# the denominator of GPCM
```

```
for (u in 1:m[i]){
for (r in 1:u){ eta[j, i, u, r] <- 1.7*alpha[i]*(theta[j]-
beta[i]+d[i,r])}
eta_sum[j, i, u] <- exp(sum(eta[j, i, u, 1:u]))
deno[j,i] <- sum(eta_sum[j, i, 1:m[i]])
for (k in 1:m[i]){
# the numerator of GPCM
for (r in 1:k){ nume_eta[j, i, k, r] <-
1.7*alpha[i]*(theta[j]-beta[i]+d[i,r])}
nume[j,i,k] <- exp(sum(nume_eta[j, i, k, 1:k]))
# probability of the kth category
pai[j,i,k] <- nume[j,i,k] / deno[j,i]}
```

Table A4

Spearman's Correlations of 21 Process Features Extracted from Process Data of Seven Items

	u19a_lv1	u19a_lv2	u19a_lv3	u19b_lv1	u19b_lv2	u19b_lv3	u07_lv1	u07_lv2	u07_lv3	u02_lv1	u02_lv2	u02_lv3
u19a_lv1	1.000											
u19a_lv2	0.053	1.000										
u19a_lv3	-0.112**	-0.004	1.000									
u19b_lv1	-0.381**	0.011	-0.055	1.000								
u19b_lv2	-0.232**	0.114**	-0.075*	0.148**	1.000							
u19b_lv3	-0.400**	-0.090**	-0.168**	0.295**	0.034	1.000						
u07_lv1	0.144**	-0.134**	0.073*	-0.128**	-0.306**	-0.025	1.000					
u07_lv2	0.044	-0.014	-0.031	-0.033	-0.099**	-0.026	0.016	1.000				
u07_lv3	-0.054	0.018	0.005	0.040	0.093**	-0.025	-0.008	-0.251**	1.000			
u02_lv1	0.247**	-0.222**	0.046	-0.176**	-0.423**	-0.067*	0.296**	0.110**	-0.074*	1.000		
u02_lv2	0.119**	-0.043	-0.005	-0.020	-0.011	-0.102**	0.028	0.011	-0.006	0.079*	1.000	
u02_lv3	-0.110**	0.073*	-0.075*	0.082*	0.177**	0.075*	-0.141**	-0.009	-0.009	-0.308**	-0.040	1.000
u11b_lv1	0.179**	-0.127**	0.007	-0.167**	-0.226**	-0.090**	0.148**	0.052	-0.043	0.264**	0.028	0.028
u11b_lv2	0.039	-0.051	-0.016	-0.056	-0.119**	0.005	0.123**	-0.036	-0.006	0.146**	-0.043	-0.043
u11b_lv3	-0.028	0.023	0.026	-0.008	0.056	0.013	-0.019	-0.027	0.051	-0.067*	0.089**	0.089**
u16_lv1	-0.092**	0.049	0.045	0.036	-0.016	0.005	-0.001	-0.045	0.035	-0.046	-0.035	-0.035
u16_lv2	-0.250**	0.271**	-0.007	0.192**	0.303**	0.091**	-0.251**	-0.071*	0.058	-0.355**	-0.040	-0.040
u16_lv3	-0.022	0.022	0.059	0.023	0.013	-0.071*	-0.039	0.058	0.001	-0.022	-0.046	-0.046
u23_lv1	0.177**	-0.193**	0.045	-0.189**	-0.367**	-0.072*	0.284**	0.122**	-0.075*	0.368**	0.033	0.033
u23_lv2	-0.026	0.042	0.018	0.020	-0.103**	0.082*	0.065*	0.079*	-0.013	0.027	0.019	0.019
u23_lv3	.071*	-0.005	-0.001	-0.092**	-0.055	-0.057	.074*	-0.040	0.023	-0.017	-0.071*	-0.071*
u02_lv3	1.000											
u11b_lv1	-0.112**	1.000										
u11b_lv2	-0.065*	-0.042	1.000									
u11b_lv3	0.045	-0.039	0.115**	1.000								
u16_lv1	0.028	-0.030	-0.012	0.023	1.000							
u16_lv2	0.142**	-0.177**	-0.125**	0.043	-0.048	1.000						
u16_lv3	-0.062	-0.028	-0.061	0.011	0.024	-0.027	1.000					
u23_lv1	-0.200**	0.207**	0.122**	-0.041	-0.095**	-0.381**	-0.014	1.000				
u23_lv2	0.044	0.012	-0.025	0.045	0.093**	0.004	0.028	-0.123**	1.000			
u23_lv3	-0.022	0.047	-0.014	0.024	-0.057	-0.014	0.002	0.105**	-0.068*	1.000		

Note. According to Table A3, some process variables were not normally distributed. Therefore, the Spearman's correlations were computed, instead of the Pearson's correlations. The process variables of all items are listed in order of item administration.

* $p < 0.05$.
 $p < 0.01$.

```

Y[j,i] ~ dcat(pai[j,i,(1:m[i])]) # because the category
start from 1}}
# prior models
for (j in 1:J){
# define the prior of theta: mean and precision (the inverse
of variance)
theta[j] ~ dnorm(0, 1/9)}'
#### 2. compile the model ####
irt_jags <- jags.model(textConnection(GPCM),
data = list(Y = Y, # response data matrix
I = 7, # number of items
J = nrow(data), # number of respondents
m = item_para$category, # each item has m[i] categories
alpha = item_para$alpha,
beta = item_para$beta,
d = d),
inits = list(list(.RNG.name = "base::Wichmann-Hill",
.RNG.seed = 10),
list(.RNG.name = "base::Marsaglia-Multicarry",
.RNG.seed = 20)),
n.chains = 2)
#### 3. update the model, constituting the burn-in phase
####
update(irt_jags, n.iter = 10000)
#### 4. simulate posterior ####

```

```

params <- c("theta")
irt_sim <- coda.samples(model = irt_jags, vari-
able.names = params, n.iter = 10000)
plot(irt_sim) # trace plot of the posterior samples
# convergence diagnostics
gelman.diag(irt_sim) # the Gelman-Rubin diagnostic:
scale reduction factors for each theta
gelman.plot(irt_sim) # the development of the scale-
reduction over the chain iterations
# get posterior summary of the 10000 draws for all thetas
para.sum <- MCMCsummary(irt_sim)
#####
##### To fit the combination model through the Rjags
package #####
#####
library(rjags)
library(MCMCvis)
data <- read.csv("combined_score&lv.csv") # the file con-
taining both response data and process variables, in which
each row represents a respondent, including 29 columns: the
first column is respondents' IDs, the 2nd to 8th columns are
scores of seven items, the 9th to 29th columns contains the
21 process variables.

```


Table A5*Priors of Regression Parameters of θ in the Combination Model for the Second Half of the Sample*

Number of Items Used	Item Used	Priors
1	U19a	$b_0 \sim N(-0.496, 0.014)$ $b_{1_1} \sim N(4.622, 0.581)$ $b_{1_2} \sim N(-3.913, 0.655)$ $b_{1_3} \sim N(1.615, 0.688)$
2	U19a, U19b	$\sigma^2 \sim IG(241.500, 105.278)$ $b_0 \sim N(-0.447, 0.002)$ $b_{1_1} \sim N(1.453, 0.159)$ $b_{1_2} \sim N(-1.440, 0.167)$ $b_{1_3} \sim N(0.591, 0.221)$ $b_{2_1} \sim N(-2.654, 0.086)$ $b_{2_2} \sim N(-4.839, 0.167)$ $b_{2_3} \sim N(1.161, 0.190)$
3	U19a, U19b, U07	$\sigma^2 \sim IG(241.500, 24.764)$ $b_0 \sim N(-0.400, 0.001)$ $b_{1_1} \sim N(1.520, 0.041)$ $b_{1_2} \sim N(0.305, 0.085)$ $b_{1_3} \sim N(-0.369, 0.107)$ $b_{2_1} \sim N(1.347, 0.117)$ $b_{2_2} \sim N(-1.103, 0.119)$ $b_{2_3} \sim N(0.982, 0.169)$ $b_{3_1} \sim N(-1.907, 0.053)$ $b_{3_2} \sim N(-3.689, 0.110)$ $b_{3_3} \sim N(1.145, 0.129)$
4	U19a, U19b, U07, U02	$\sigma^2 \sim IG(241.500, 23.365)$ $b_0 \sim N(-0.242, 0.001)$ $b_{1_1} \sim N(1.697, 0.063)$ $b_{1_2} \sim N(1.238, 0.194)$ $b_{1_3} \sim N(-2.919, 0.271)$ $b_{2_1} \sim N(1.379, 0.037)$ $b_{2_2} \sim N(0.298, 0.084)$ $b_{2_3} \sim N(-0.349, 0.100)$ $b_{3_1} \sim N(1.055, 0.089)$ $b_{3_2} \sim N(-0.742, 0.079)$ $b_{3_3} \sim N(0.479, 0.107)$ $b_{4_1} \sim N(-1.610, 0.040)$ $b_{4_2} \sim N(-3.152, 0.089)$ $b_{4_3} \sim N(0.692, 0.084)$
5	U19a, U19b, U07, U02, U11b	$\sigma^2 \sim IG(241.500, 34.901)$ $b_0 \sim N(-0.279, 0.001)$ $b_{1_1} \sim N(1.357, 0.053)$ $b_{1_2} \sim N(1.145, 0.165)$ $b_{1_3} \sim N(-2.656, 0.226)$ $b_{2_1} \sim N(1.215, 0.032)$ $b_{2_2} \sim N(0.223, 0.068)$ $b_{2_3} \sim N(-0.232, 0.085)$ $b_{3_1} \sim N(-0.734, 0.036)$ $b_{3_2} \sim N(-1.656, 0.073)$ $b_{3_3} \sim N(0.074, 0.054)$ $b_{4_1} \sim N(0.891, 0.076)$ $b_{4_2} \sim N(-0.436, 0.071)$ $b_{4_3} \sim N(0.432, 0.090)$ $b_{5_1} \sim N(-1.402, 0.035)$ $b_{5_2} \sim N(-2.730, 0.077)$ $b_{5_3} \sim N(0.622, 0.073)$
6	U19a, U19b, U07, U02, U11b, U16	$\sigma^2 \sim IG(241.500, 29.584)$ $b_0 \sim N(-0.215, 0.001)$ $b_{1_1} \sim N(1.199, 0.048)$ $b_{1_2} \sim N(1.044, 0.146)$ $b_{1_3} \sim N(-2.355, 0.202)$ $b_{2_1} \sim N(1.136, 0.026)$ $b_{2_2} \sim N(0.228, 0.057)$ $b_{2_3} \sim N(-0.252, 0.070)$ $b_{3_1} \sim N(0.911, 0.018)$ $b_{3_2} \sim N(0.087, 0.024)$ $b_{3_3} \sim N(0.055, 0.105)$

(Continued)

Table A5
(Continued)

Number of Items Used	Item Used	Priors
6	U19a, U19b, U07, U02, U11b, U16	$b4_1 \sim N(-0.670, 0.029)$ $b4_2 \sim N(-1.453, 0.062)$ $b4_3 \sim N(0.018, 0.045)$ $b5_1 \sim N(0.744, 0.066)$ $b5_2 \sim N(-0.394, 0.058)$ $b5_3 \sim N(0.467, 0.077)$ $b6_1 \sim N(-1.266, 0.030)$ $b6_2 \sim N(-2.424, 0.066)$ $b6_3 \sim N(0.594, 0.063)$ $\sigma^2 \sim IG(241.500, 23.241)$
7	U19a, U19b, U07, U02, U11b, U16, U23	$b0 \sim N(-0.246, 0.001)$ $b1_1 \sim N(0.974, 0.039)$ $b1_2 \sim N(0.989, 0.114)$ $b1_3 \sim N(-1.988, 0.162)$ $b2_1 \sim N(0.958, 0.021)$ $b2_2 \sim N(0.233, 0.049)$ $b2_3 \sim N(-0.294, 0.057)$ $b3_1 \sim N(0.751, 0.014)$ $b3_2 \sim N(-0.012, 0.019)$ $b3_3 \sim N(0.202, 0.084)$ $b4_1 \sim N(-0.474, 0.024)$ $b4_2 \sim N(-1.163, 0.053)$ $b4_3 \sim N(0.011, 0.037)$ $b5_1 \sim N(0.669, 0.052)$ $b5_2 \sim N(-0.314, 0.048)$ $b5_3 \sim N(0.386, 0.060)$ $b6_1 \sim N(-1.073, 0.024)$ $b6_2 \sim N(-2.005, 0.053)$ $b6_3 \sim N(0.575, 0.052)$ $b7_1 \sim N(1.162, 0.030)$ $b7_2 \sim N(-0.657, 0.032)$ $b7_3 \sim N(0.935, 0.082)$ $\sigma^2 \sim IG(241.500, 16.848)$

Note. The priors listed in this table are the posteriors for the same parameters obtained from the first half of the sample. N represents the normal distribution and the numbers in the parentheses are the mean and variance respectively. IG denotes the inverse-gamma distribution.

```

item_para ← read.csv("item_para.csv")
i0 ← rep(0,7)
d ← cbind(i0,item_para[,c("t1","t2","t3")])
d[is.na(d$t1),"t1"] ← 0
Y ← data[,2:8] + 1 # the response data and the response
category starts from 1
##### 1. define the model #####
GPCM ← '
var pai[J,I,4],eta[J,I,4,4],eta_sum[J,I,4],nume[J,I,4],
nume_eta[J,I,4,4];
model{
# likelihood for Y
for (j in 1:J){ # person j
for (i in 1:I){ # item i
# the denominator of GPCM
for (u in 1:m[i]){
for (r in 1:u){
eta[j, i, u, r] ← 1.7*alpha[i]*(theta[j]-beta[i]+d[i,r])
eta_sum[j, i, u] ← exp(sum(eta[j, i, u, 1:u]))
deno[j,i] ← sum(eta_sum[j, i, 1:m[i]])
for (k in 1:m[i]){
# the numerator of GPCM

```

```

for (r in 1:k){
nume_eta[j, i, k, r] ← 1.7*alpha[i]*(theta[j]-
beta[i]+d[i,r])
nume[j,i,k] ← exp(sum(nume_eta[j, i, k, 1:k]))
# probability of the kth category
pai[j,i,k] ← nume[j,i,k] / deno[j,i]
Y[j,i] ~ dcat(pai[j,i,(1:m[i])]) # because the category
start from 1}
# define the prior of theta
theta[j] ~ dnorm(miu[j], tau.sq) # tau.sq is 1/er-
ror_variance
for (cv_i in 1:21){ # 21 is the number of covariates.
cv_product[j,cv_i] ← b1[cv_i]*X[j,cv_i] # X denote the
matrix of covariates}
miu[j] ← b0 + sum(cv_product[j,1:21]) # 21 is the num-
ber of covariates!!!}
# prior models
b0 ~ dnorm(0, 0.01) # give b0 a prior N(0, 10^2)
for (b1_i in 1:21){ # 21 is the number of process variables
b1[b1_i] ~ dnorm(0, 1/4) # give each slope parameter a
prior N(0,2^2), all slopes are stored in b1}
tau.sq ~ dgamma(1, 1) # give variance an inverse-gamma
prior by giving the inverse of variance a gamma prior}'
##### 2. compile the model #####
irt_jags ← jags.model(textConnection(GPCM),

```

```

data = list(X = data[,9:(9+21-1)], # the data of 21 process
variables
Y = Y, # response data
I = 7, # number of items
J = nrow(data), # number of respondents
m = item_para$category, # each item has m[i] categories
alpha = item_para$alpha,
beta = item_para$beta,
d = d),
inits = list(list(.RNG.name = "base:Wichmann-Hill",
.RNG.seed = 10),
list(.RNG.name = "base:Marsaglia-Multicarry",
.RNG.seed = 20)),
n.chains = 2)
#### 3. update the model, constituting the burn-in phase
####

```

```

update(irt_jags, n.iter = 10000)
#### 4. simulate posterior ####
params <- c("b0", "b1", "tau.sq", "theta")
irt_sim <- coda.samples(model = irt_jags, variable.names = params, n.iter = 50000)
plot(irt_sim) # trace plot of the posterior samples
#### convergence diagnostics
gelman.diag(irt_sim) # the Gelman-Rubin diagnostic:scale reduction factors for each parameter
gelman.plot(irt_sim) # the development of the scale-reduction over the chain iterations (also useful for determining a burn-in)
#### get posterior summary of the 10000 draws for all theta
para.sum <- MCMCsummary(irt_sim)

```