Psychological Test and Assessment Modeling, Volume 65, 2023 (1), 145-178

Predicting Problem-Solving Proficiency with Multiclass Hierarchical Classification Using Process Data: A Machine Learning Approach

Qiwei He[1], *Qingzhou Shi*[2], *Elizabeth L. Tighe*[3]

[1] Educational Testing Service
[2] College of Education, University of Alabama
[3] Department of Psychology, Georgia State University

Abstract

Increased use of computer-based assessments has facilitated data collection processes that capture both response product data (i.e., correct and incorrect) and response process data (e.g., time-stamped action sequences). Evidence suggests a strong relationship between respondents' correct/incorrect responses and their problem-solving proficiency scores. However, few studies have reported the predictability of fine-grained process information on respondents' problem-solving proficiency levels and the degree of granularity needed for accurate prediction. This study uses process data from interactive problem-solving items in the Programme for the International Assessment of Adult Competencies (PIAAC) to predict proficiency levels with hierarchical classification methods. Specifically, we extracted aggregate-level process variables and item-specific sequences of problem-solving strategies. Two machine learning methods – random forest and support vector machine – affiliated with two multiclass hierarchical classification approaches (i.e., flat classification and hierarchical classification) were examined. Using seven problem-solving items from the U.S. PIAAC process data sample, we found that the hierarchical approach affiliated with any machine learning method performed moderately better than the flat approach in proficiency level prediction. This study demonstrates the feasibility of using process variables to classify respondents by problem-solving proficiency levels, and thus, supports the development of tailored instructions for adults at different levels.

Keywords: multiclass hierarchical classification, flat classification, machine learning, process data, problem-solving proficiency, PIAAC

Author Note

Correspondence concerning this article should be sent to Qiwei He, Educational Testing Service, 660 Rosedale Road, Princeton, NJ 08541, USA, email: qhe@ets.org

This research work was supported by the Institutes of Education Sciences, U.S. Department of Education, through Grant R305A210344.

1 Introduction

Increased use of computer-based assessments has facilitated data collection processes that record not only response data (i.e., correct and incorrect) but also granular process data (e.g., time-stamped action sequences) (Goldhammer et al., 2013; He et al., 2021). Interactive items in low-stakes, large-scale educational assessments are designed to provide scenario-based tasks and, as such, to better reflect what individuals know and can do in the 21st century (Ulitzsch et al., 2021). For example, the Programme for the International Assessment of Adult Competencies (PIAAC; OECD 2016) and the Programme for International Student Assessment (PISA; OECD 2014, 2017) both contain complex and interactive problem-solving items that reflect skills needed in daily life. Process data (e.g., keystroke inputs, mouse clicks) often capture information about the naturally occurring behaviors that respondents display during interactions with digital problem-solving tasks (e.g., Eichmann et al., 2020; Goldhammer et al., 2014; von Davier et al., 2019).

Data stored in log files, referred to as *process data* in the present study, are able to describe when and how respondents employ actions to solve interactive tasks (He et al., 2021). Recent evidence has revealed a strong relationship among respondents' behavioral patterns, problem-solving strategies, and proficiency scores (e.g., He et al., 2019; He & von Davier, 2016; Liao et al., 2019; Tang et al., 2020). However, few studies have reported the predictability of fine-grained process information on respondents' problem-solving proficiency levels and the degree of granularity required for accurate prediction. This quantified information would assist in developing more tailored instruction (e.g., specific instructions to guide low-skilled adults to use basic email functions such as reply, reply to all, and provide more advanced instruction for relatively high-skilled adults to identify key information when searching webpages) to help improve respondents' problem-solving proficiency levels.

In this paper, we use process data from seven PIAAC Problem Solving in Technology Rich Environment (PSTRE) items to demonstrate the predictability of process data on adults' problem-solving proficiency levels with supervised machine learning methods. Specifically, we used random forest (RF) and support vector machines (SVM) affiliated with two multiclass hierarchical classification approaches (i.e., flat classification and hierarchical classification). Both aggregate-level process variables (i.e., response time, number of actions, and time to the first action) and item-level process variables (i.e., sequence similarity and efficiency) that specify problem-solving strategies with action sequences were used in this study.

1.1 Problem-solving proficiency in PSTRE

PIAAC was administered in over 30 countries between 2012 and 2017 with a focus on adults aged 16 to 65 years old. PIAAC consists of an extensive background survey, which includes a wide range of socio-demographic information from respondents

(e.g., educational attainment, immigrant status, labor market status, information on familiarity with information, communication, and technology (ICT), and the use of digital technologies at work and in everyday life) as well as literacy, numeracy, and PSTRE assessments (Schleicher, 2008).

Problem-solving proficiency, captured in the PIAAC PSTRE domain, is defined as "using digital technology, communication tools, and networks to acquire and evaluate information, communicate with others, and perform practical tasks" (OECD, 2009, p. 9). In other words, this domain does not focus specifically on assessing computer literacy or basic ICT skills, but rather encompasses the active construction of strategies, goals, and planning that adults need to solve complex problems in personal, civic, and working environments that require the use of digital technologies (OECD, 2009).

There are three primary dimensions underlying the construct of PSTRE: (1) cognitive dimensions, which include planning, goal setting and progress monitoring, self-organizing, acquiring and evaluating information, and using information; (2) task-specific dimensions, which include items that require single or multi-steps, implicit or explicit problem prompts, and single or multiple constraints; and (3) technology dimensions, which includes email, web, and/or spreadsheet digital environments. These dimensions characterize the plethora and complexity of the skills that feed into digital problem-solving. For example, a PSTRE item may ask an adult to engage in online shopping by reviewing and clicking on webpages that follow specific criteria (e.g., identifying customer service for return policy; see an example item in Figure 1). This single item includes a simulated web environment and an email environment (technology dimension), multiple steps to complete (clicking through multiple webpages; task-specific dimension), and acquiring, evaluating, and making use of information from the various webpages to meet the specified criteria (cognitive dimension). In addition, this item assumes that the respondent can integrate basic ICT skills (e.g., clicking on different web pages, identifying a newly received email) and foundational literacy skills (e.g., reading the item prompt to understand the criteria and email content).

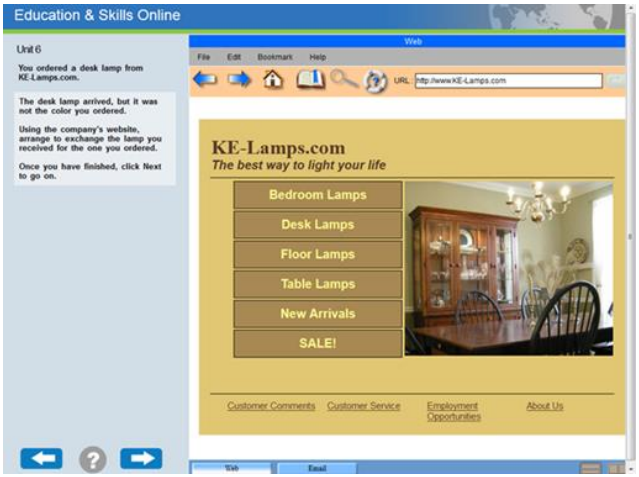
There are four proficiency levels specified in PSTRE: Below Level 1 (the lowest level), Level 1, Level 2, and Level 3 (the highest level)[1]. In accordance with the PSTRE task design framework (OECD, 2009, 2012), adults performing at PSTRE proficiency Below Level 1 (0 to 240 score points) are only able to complete tasks that involve the use of a single function within a generic interface to meet one explicit criterion without any categorical, inferential reasoning, or transforming of information. At this level, few steps are required and no subgoal has to be generated. Adults who perform at Level 1 (241 to 290 score points) can solve tasks that typically demand the basic use of widely available and familiar technology applications, such as email or a web browser, which involve few steps and a minimal number of operators. At this level, there is still little navigation required to access the information, and the

[1] The proficiency level corresponds to the item difficulty level in PIAAC, which was derived by a linear transformation from the item parameters into the performance score scale. See more details in the PIAAC technical report (OECD, 2016).

problem may be solved regardless of a respondent's awareness and use of specific tools and functions. Adults performing at PSTRE proficiency Level 2 (291 to 340 score points) can solve tasks that typically require the use of both generic and more specific technology applications (e.g., online form, spreadsheet), which involve multiple steps and operators (e.g., extract information from a spreadsheet and then input the useful information into an email environment). Some navigation across pages and applications is required to solve problems at this level. In addition, Level 2 adults need to demonstrate the ability to apply tools to resolve complex problems. For the highest level, PSTRE Level 3 (341 to 500 score points), adults can solve tasks with the use of both generic and more specific technology (e.g., reserve a meeting room online). At this level, tool usage (e.g., sorting and searching functions in the spreadsheet environment) is required to solve the items. Adult respondents must use clear self-defined subgoals to solve the task, especially because many of the tasks have unexpected outcomes and impasses (e.g., taking the time conflict and room capacity into consideration when booking a meeting room online). The hierarchical structure of the PSTRE proficiency levels allows us to characterize adults' response profiles and test-taking behaviors by two levels, namely, a binary classification, high proficiency (Level 2/ Level 3) and low proficiency (Below Level 1/Level 1) groups at the first level, and four PSTRE proficiency groups (i.e., Below Level 1, Level 1, Level 2, and Level 3) at the second level under the high and low categories correspondingly.

Figure 1.

An example PSTRE item



Note. An example item with an online shopping web environment from the Education and Skills Online Assessment, which shares the item interface structure with the PIAAC item design.

1.2 Exploring PSTRE proficiency with process data

As noted above, the complexity of defining adults' PSTRE proficiency brings new challenges in how we measure problem-solving skills, which calls for more data-driven evidence to describe the interactive problem-solving process that goes beyond response data alone (correct/incorrect on the task). As suggested by many recent studies (e.g., Gao et al., 2022; He et al., 2019b; Ulitzsch et al., 2022a), process data may be more appropriate to fully describe adults' behaviors and strategies during interactive tasks, such as PIAAC PSTRE items.

Much of the past process data research focuses on a single PSTRE item and compares respondents' strategies across different respondent groups by performance (i.e., success or failure) or demographics (e.g., gender, age, occupation, country). For example, Liao et al. (2019) focused on a particularly difficult PSTRE item (U02) and described the top action sequences (e.g., view folder, go to web environment, click cancel button) and how often each action was performed by respondents with different background characteristics (e.g., educational attainment, income). Results indicated differences by background characteristics, such that those with higher educational attainment and higher incomes made clearer goals to solve the item (based on which actions they undertook and applied) and were more likely to use pertinent actions to help solve the item (e.g., sorting and help actions). In another example, He and von Davier (2015, 2016) examined strategies on a single PSTRE item across countries. The higher performing group was characterized by more frequent use of the searching/sorting tool, whereas the lower performing group was more likely to engage in random clicks and use the help function frequently. In addition, the researchers reported that the lower performing group presented more hesitative behaviors, such as clicking the "cancel" button more frequently when approaching the next item. Xiao et al. (2021) applied hidden Markov models on time-stamped action sequence data to identify the latent states and transitions between states underlying the problem-solving process on two PIAAC PSTRE items. The groups with correct responses on both items were more engaged in the tasks (e.g., significantly longer action sequences) and used efficient tools more frequently to solve the tasks (e.g., using sorting and searching functions in a spreadsheet). In contrast, the group with incorrect responses was more likely to use shorter action sequences and exhibit more hesitative behaviors (e.g., clicking on the cancel button before heading to the next item or repeatedly selecting and canceling the sorting function). Most recently, Ulitzsch et al. (2022b) explored the early predictability of behavioral outcomes on interactive tasks with early-window clickstream data. These data can give insight into a respondent's progression through an item that ultimately leads to an incorrect response, for example, skipping an item immediately, pausing and then skipping an item, clicking and completing half of the item, or clicking through a series of screens and ultimately ending up selecting the incorrect answer. Based on derived features related to the occurrence, frequency, sequentiality, and timing of performed actions from early-window clickstreams, the authors used extreme gradient boosting to dynamically classify respondents who have a high probability of being out of track when solving a PSTRE task. These examples demonstrate the utility

of using process data to gain deeper insights into the strategies and processes of respondents on complex PSTRE items.

Some recent studies have also explored the possibility of identifying respondents' general behavioral features across several items. These studies have examined the consistency of respondents' behavioral patterns and strategies under various conditions. For example, Tang et al. (2020) explored process data from 14 PIAAC PSTRE items to extract latent variables through a multidimensional scaling framework and computed a dissimilarity measure to quantify the discrepancy between response process pairs. The authors found that a substantial amount of information was preserved in the process data and was predictive of demographic characteristics (age, gender) and adults' basic literacy and numeracy skills with a high accuracy rate. He et al. (2021) developed two process indicators (i.e., sequence similarity and efficiency) across different items that described the similarity and efficiency of respondents' action sequences against predefined sequences and reported high associations of these two features with problem-solving proficiency. Chen et al. (2019) proposed a model-based approach for the dynamic prediction of behavioral outcomes under different interactive environments. More specifically, the authors proposed to include features as time-varying covariates in an event history model, which at any given time during the solution process can be used to predict outcomes of the solution process (i.e., success or failure and time spent on the task). Finally, Xiao and Liu (2023) extended the method proposed in Chen et al. (2019) by defining the easiness parameter from the task level to the state level and adding the task's process characteristics into consideration.

All of these examples show the enormous potential of process data variables to predict PSTRE proficiency in addition to the information provided by the response data. Process data are highly informative in describing patterns of behaviors from different dimensions, including engagement, efficiency, response patterns, and strategies, which to some degree may be indicative of the extent to which adult respondents do or do not struggle with complex PSTRE tasks. For example, many low-skilled adults face challenges with basic technology skills, foundational literacy and numeracy skills, and/or problem-solving skills (Vanek, 2017). Therefore, process data may help identify breakdowns throughout the problem-solving process for these adults that would not be possible with response data (correct/incorrect) alone. This would enable researchers to develop better digital assessments tailored to low-skilled adults as well as education systems to tailor and integrate explicit digital problem-solving instructional practices to better equip these adults for the workforce and using technology in daily life (Cummins et al., 2019; Vanek, 2017).

1.3 The present study

The purpose of this paper is to demonstrate the predictivity of process data on PSTRE proficiency levels by highlighting a prediction process using supervised machine learning methods affiliated with hierarchical classification approaches. Specifically, we address two research questions:

(1) Can the process variables accurately predict adults' PSTRE proficiency levels? Which variable(s) are the most important in the prediction process?
(2) How do the four classification models perform in predicting PSTRE proficiency levels? From a technical perspective, are the model performances significantly different from each other?

The remainder of this article is structured as follows. In Section 2, we introduce the dataset, process variables, and machine learning methods (RF and SVM) with an affiliation of two classification approaches (i.e., flat and hierarchical classification). In Section 3, we present the performance results of the four models and report the most important process variables in the prediction process. Finally, in Section 4, we discuss the implications of our findings and provide an outlook for further development of prediction models using process data.

2 Method

2.1 Data

In this study, we used response and process data from seven items that were administered in the second cluster of PIAAC PSTRE (PS2) with a focus on the U.S. sample. A total of 1,338 adult respondents were available in the PS2 module. To evaluate classification performance and to avoid confusion from missing values, we only included adults who responded to all seven items in the PS2 module (N = 935). Based on PSTRE proficiency score thresholds[2] on the first PSTRE plausible value[3], the sample consisted of 49.3% in the low PSTRE proficiency group (i.e., Below Level 1 and

[2] Performance on PSTRE can be categorized into four levels: Below Level 1 (0–240), Level 1 (241–290), Level 2 (291–340), and Level 3 (341–500). For more details, refer to OECD (2016).

[3] The plausible value in PIAAC was derived from the population model that combined information from both the sample's survey and demographic background and response information. In order to highlight the classification performance of process data, in this study, we only used the first one out of ten plausible values derived from repetitive estimations in PIAAC. This approach helped place the respondents in a unique labeled category, thus enabling comparisons among machine learning methods affiliated with flat and hierarchical approaches. For more details of plausible values, refer to PIAAC technical report in OECD (2016).

Level 1) and 50.7% in the high PSTRE proficiency group (i.e., Level 2 and Level 3). It is noted that the sample sizes in Below Level 1 and Level 3 were imbalanced and relatively small, 8.8% and 6.4% of the whole sample, respectively.

Table 1 displays the sample demographic profiles split by the four PSTRE proficiency levels. The sample was relatively balanced in terms of gender, with 53% self-reporting as female. One interesting note was that females were over-represented in the low proficiency group (Below Level 1 and Level 1), nearly 10% higher than the male group. Approximately half of the sample attained a level of education higher than a high school diploma. Adults with lower educational attainment (i.e., high school and below) represented a larger proportion in the low proficiency group. The average age of the sample was 38 (SD = 13.9), with older adults over-represented in the Below Level 1 proficiency group (M = 45 years old) and slightly younger adults in the Level 2 proficiency group (M = 36 years old). Comparative data on the self-reported use of digital technologies at work and at home revealed positive associations with respondents' PSTRE proficiency levels. The lowest value of ICT at Home (1.80) and ICT at Work (1.59) was reported in the Below Level 1 group, whereas the highest value was reported in the Level 3 group.

Table 1

Sample Description by PSTRE Proficiency Levels (N=935)

	Below Level 1	Level 1	Level 2	Level 3	Total
N (%)	82 (8.8)	379 (40.5)	414 (44.3)	60 (6.4)	935 (100)
Gender					
Female (%)	40 (8.1)	227 (45.8)	201 (40.5)	28 (5.6)	496 (100)
Male (%)	42 (9.6)	152 (34.6)	213 (48.5)	32 (7.3)	439 (100)
Education					
Less than high school (%)	12 (16.7)	35 (48.6)	25 (34.7)	0 (0)	72 (100)
High school (%)	41 (11.7)	167 (47.9)	133 (38.1)	8 (2.3)	349 (100)
Above high school (%)	29 (5.7)	176 (34.3)	256 (49.9)	52 (10.1)	513 (100)
Age					
Mean	45.01	38.82	36.23	38.97	38.23
SD	13.48	14.77	12.95	12.2	13.91
ICT Home					
Mean	1.80	2.23	2.55	2.71	2.37
SD	0.88	0.88	0.98	0.58	0.94
ICT Work					
Mean	1.59	2.09	2.37	2.45	2.23
SD	0.82	1.10	1.14	1.02	1.11

Note. PSTRE proficiency score is categorized into four levels: Below Level 1 (0–240), Level 1 (241–290), Level 2 (291–340), and Level 3 (341–500).

2.2 Instrument

As mentioned above, this study only included the seven items from the PS2. The rationale for this was to include the items in which the predefined action sequences had already been fully validated by item developers and content experts at the time that we developed this study. The predefined action sequences were critical in developing new process variables across items in this study, which is described in detail in section 2.3. These items were administered to the respondents in a fixed order, and there were no limits on time or solutions imposed. Of the seven items, three are dichotomous and four are polytomous. Table 2 summarizes the content, difficulty, and environments of

each item. The third and fourth columns illustrate the item difficulty score and its corresponding PIAAC PSTRE proficiency level. The most difficult items were classified at Level 3, and the easiest were classified at Level 1. It is important to note that the difficulty parameter of each item was calibrated by the response data only. In other words, the high-difficulty item was not necessarily the one with the most complicated interactive interface or which required a longer time or more action sequences to solve. For example, U02 and U11b are both high-difficulty items at Level 3. However, item U02 involves more complicated procedures and requires respondents to switch between web and email environments, whereas item U11b only involves a single environment with a straightforward email interface design. To complete these two items, on average, adults used 4.7 minutes and 66.2 actions on U02 but used only half the amount of time and much fewer actions, that is, 2 minutes and 36 actions, to complete U11b. The last three columns in Table 2 present the environments (i.e., email, web, and spreadsheet) of each item. In the PS2 module, three PSTRE items involve multiple environments, whereas four items involve only one environment.

Table 2

Item Content, Difficulty Level, and Environments of the Seven PIAAC PSTRE Items in PS2

Item ID	Item Content	Score	Level	Number of RS	Environments		
					Email	Web	Spreadsheet
U19a	Club Membership	268	1	4	X		X
U19b	Club Membership	296	2	4			X
U07	Book Order	305	2	2		X	
U02	Meeting Room	346	3	5	X	X	
U16	Reply All	286	1	16	X		
U11b	Locate Email	355	3	18	X		
U23	Lamp Return	321	2	1	X	X	

Note. The score thresholds to difficulty levels follow the rule: Below Level 1 (0–240), Level 1 (241–290), Level 2 (291–340), and Level 3 (341–500). For more details, refer to OECD (2016). RS indicates the predefined action sequences (i.e., reference sequences) for each item.

2.3 Process variables

We extracted three aggregate-level process variables and two item-level process variables for prediction models in this study. The three aggregate-level process variables included *total response time (T)*, *number of actions (including keystrokes) (A)*, and *time to the first action (F)*, which have been widely used in recent studies to describe respondents' test-taking behaviors in reading, math, complex problem solving, and collaborative problem solving (e.g., de Boeck & Scalise, 2019; Engelhardt et al., 2019; Goldhammer et al., 2014; Han et al., 2019; He et al., 2022; Liao et al., 2019; Stadler et al., 2019) and to enhance latent ability estimation in psychometric joint modeling (e.g., Lu et al., 2020; Qiao et al., 2022; Zhang et al., 2022).

The two item-level process variables were derived from fine-grained action sequences, specifically by computing the sequence distance between individual action sequences and the predefined (reference) ones established by item developers and content experts. These distance measures facilitate the development of new indicators that characterize the behavior of respondents across items (He et al., 2021). In this study, we extracted two indicators, *similarity* and *efficiency*, which have been previously proposed by He et al. (2019, 2021). We extracted these indicators by computing the distance measures between the observed and the reference sequence with the longest common subsequence (LCS) method. The LCS of a set of sequences is a subsequence whose length equals the maximum number of actions that are shared, in sequential order, with the reference sequences. (See algorithms for computing LCS in Appendix A. For more details about the LCS method see He et al., 2021; Sukkarieh et al., 2012).

The indicator *similarity* captures how much, on average, a respondent's sequence deviates from a reference sequence (or the closest reference sequence in the case of items designed to have multiple reference sequences) predefined by item developers and content experts. For each item, similarity is defined as the ratio between the length of LCS (i.e., $len(LCS)$) and the length of the reference sequence (i.e., $len(RS)$). The higher the ratio, the more similar the observed sequence is to the reference sequence.

The indicator *efficiency* measures a respondent's ability to solve items using the minimum possible number of actions and is operationalized by the number of actions undertaken by a respondent over the number of actions contained in the reference sequence. High efficiency indicates that there are no or few excess (redundant) actions. Efficiency is defined as the ratio between the length of LCS (i.e., $len(LCS)$) and the length of the observed sequence (i.e., $len(OS)$) and measures the degree to which the LCS and the actual observed sequence overlap. A ratio close to 1 implies that a large proportion of the LCS can be matched with the OS, namely, the respondent solving the problem in an efficient way without performing too many actions that do not belong to the reference sequence.

As shown in the fifth column in Table 2, it is noted that the predefined action sequences do not have to be unique in order to successfully solve an item. In the PS2,

only one item (U23) has a single unique predefined action sequence to solve the item, all of the others have multiple solutions. Item U11b represents the maximum number of available predefined action sequences, with 18 available solutions to successfully solve the item. It is possible to adapt the LCS method to fit situations in which multiple solutions for a task exist: in these contexts, the reference sequence that generates the longest LCS when paired with the observed sequence would be retained as the solution path that the respondent was most likely to follow (He et al., 2021). For example, when identifying key information in a spreadsheet item, respondents are allowed to use either the searching or sorting function. Therefore, there could be at least two predefined reference sequences: one which uses the search function (RS_Search), and the other which uses the sorting function (RS_Sort). The LCS would be calculated by matching the individual action sequence with these two predefined reference sequences, respectively, thus, derived as two LCSs (e.g., LCS1_search and LCS2_sort). The longer LCS within the two, for instance, LCS2_sort, would indicate that the observed action sequence shares higher similarity with RS_Sort. Therefore, we would assume that this respondent was more likely to use the sorting function strategy, and hence, we would use the LCS2_sort against RS_Sort to calculate the similarity and efficiency for this respondent on this specific item.

Positive correlations have been reported between similarity and task completion in previous studies (e.g., Hao et al., 2015; He et al., 2021), whereas efficiency has been found to be negatively correlated with PSTRE proficiency scores (e.g., He et al., 2021; Ulitzsch et al., 2022a). This suggests that it is challenging to achieve high proficiency scores in an efficient way, that is, minimizing the number of redundant or useless actions to solve digital tasks successfully.

2.4 Multiclass hierarchical classification

Multiclass classification is the single-label problem of categorizing instances into precisely one of several (more than two) classes. Hierarchical classification is a system of grouping things according to a hierarchy. In the field of machine learning, hierarchical classification is sometimes referred to as instance space decomposition, which splits a complete multiclass problem into a set of smaller classification problems. It is different from the multi-label classification, in which the labels are nonexclusive and there is no constraint on how many classes the instance can be assigned to. In the current study, each respondent was labeled as only one class (i.e., Below Level 1, Level 1, Level 2, and Level 3) based on response data on PIAAC PSTRE items. These labels were set as a "gold standard" in the classification evaluation. We explored two multiclass hierarchical approaches to examine the prediction performance of process data affiliated with RF and SVM techniques.

2.4.1 Flat approach

The flat classification approach, which is the simplest one in handling hierarchical classification problems, completely ignores the class hierarchy, typically predicting only classes at the leaf nodes. This approach behaves like a traditional classification algorithm during training and testing. However, it provides an indirect solution to the problem of hierarchical classification, because, when a leaf class is assigned to an instance, one can consider that all of its ancestor classes are also implicitly assigned to that instance. However, this very simple approach has the disadvantage of having to build a classifier to discriminate among a large number of classes (all leaf classes), without exploring information about parent-child class relationships present in the class hierarchy. Panel (a) in Figure 2 illustrates this approach used in the current study. Starting from the "root" level, the flat classification approach trains the data by four classes (i.e., Below Level 1, Level 1, Level 2, and Level 3) with equal weights at one time without taking the higher hierarchy level (high and low) into consideration.

2.4.2 Hierarchical approach

The hierarchical approach (also known as a top-down approach) considers the tree-based structure in the dataset and exploits the local information on relationships among different levels. In the current study, we used the local classifier per node approach, which consists of training one binary classifier for each node of the class hierarchy. Panel (b) in Figure 2 shows the tree structure following a hierarchical approach. The training phase consisted of two levels: in the first level, we trained the classifier to distinguish the high and low proficiency groups, and then within the high and low group (parent node), respectively, we trained the classifier for the second level, that is Below Level 1 and Level 1 (binary child nodes) under the low proficiency group, and Level 2 and Level 3 (binary child nodes) under the high proficiency group. In the testing phase, the system first predicts its first level class, and then it uses that predicted class to narrow the choices of classes to be predicted at the second level, and so on, recursively, until the most specific prediction is made (for more details about hierarchical classification see Koller & Sahami, 1997; Silla & Freitas, 2011).

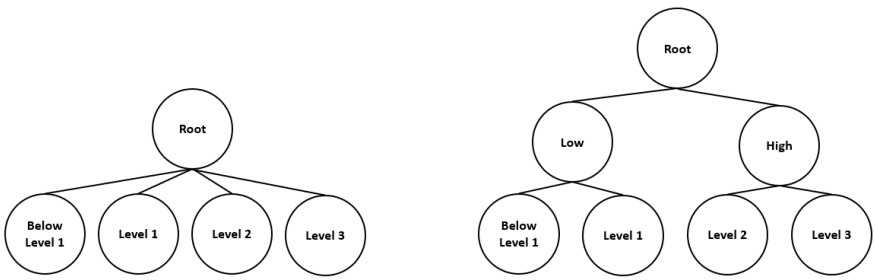
However, the hierarchical approach has two issues that need to be further considered: error propagation and overfitting. The first problem arises from the fact that we are chaining decisions and thus, propagating the error to each subsequent step. The simplest procedure for avoiding this problem consists of blocking, in which entering the next step of prediction can only happen if we achieve sufficient confidence in the prediction of the previous node. This will lead to cases where our predictions are not as informative as possible (i.e., predicting only a higher node of the tree), but they will not be as error prone. In this study, we set the confidence threshold as 0.8 in the

prediction of the first level[4] (high/low); that is, both positive predictive value (PPV) and negative predictive value (NPV) were required to be at least 0.8 to continue the prediction to the second level. Otherwise, the prediction in this instance will be blocked at the first level and only output the class as high or low and will be labeled as non-classification in the second level.

The second problem of overfitting is common to all types of classification problems, but it is particularly pressing in the hierarchical case. When we navigate our hierarchical structure, we are reducing the amount of data present in each step (because we are only focusing on a subset of potential outcomes). The problem can arise whenever we start training classifiers based on a small dataset (i.e., with a small number of observations). The probability increases that our statistical modeling of that category becomes very strict since each observation is a major contributor, which can lead to poor generalization performance. We monitored the possible overfitting issue in model training by dynamically checking the training predictive accuracy in both flat and hierarchical approaches and by employing the nested cross-validation method to reduce potential overfit in training, which is explained further in section 2.6.

Figure 2.

An illustration of flat classification and hierarchical classification approaches



[4] The confidence threshold is usually set at 0.8 in psychology and psychiatry prediction tests. The PPV means the percentage of a case predicted as a positive test and actually is positive. The NPV is just the opposite. The PPV and NPV are considered acceptable in the range of 0.8 to 1.0. We set 0.8 as the threshold in the first level to maximize (but within an acceptable range) case prediction reaching the second level where a more accurate prediction of case labels can be achieved. This threshold could be adjusted as needed in real practice. It is recommended to set a range of thresholds and to provide complete information for diagnosis or screening (for more details, refer to Dietterich, 1998; He et al., 2012, 2017).

2.5 Supervised machine learning

Supervised machine learning methods aim to train algorithms that map feature vectors to labels based on input-output pairs and infer this function to classify new data or predict class labels for unseen instances. For the present study, the goals of supervised machine learning were specified as training classifiers based on respondents' behavioral patterns (e.g., process variables) to match the class labels (i.e., PSTRE proficiency levels), thus, accurately predicting class labels (e.g., proficiency levels) for new process data collected under similar settings. We employed two commonly used and well-functioned machine learning methods, RF and SVM, affiliated with different hierarchical classification settings to predict respondents' proficiency levels with their process data.

2.5.1 Random Forest

The RF algorithm (Breiman, 2001), an extension of the classification and regression tree (CART), is a random ensemble of multiple trees. This algorithm increasingly adjusts itself by randomly combining a predetermined number of single tree algorithms. By aggregating the prediction results obtained from individual trees, the forest reduces prediction variance and improves overall prediction accuracy (Dietterich, 2000).

The complexity of the random forest algorithm is characterized by combinations of two hyperparameters, number of trees ($ntree$) and number of predictor variables used to grow a tree ($mtry$). Empirical studies (Breiman, 2001; Janitza & Hornung, 2018; Mitchell, 2011) reveal that $mtry$ and $ntree$ are more influential than other factors in controlling the complexity of the random forest algorithm. In this study, the size of a tree (i.e., the number of generations or the total number of nodes) was not restricted, and the number of branches used at each split was fixed at 2. We focused on exploring the combinations of $mtry$ and $ntree$, where $ntree = 100, 300, 500$, and $mtry = 4, 6, 8, 10, 12$. The tuning results showed that $ntree = 100$ and $mtry = 4$ produced the highest and most stable predictive accuracy rate in the flat approach. The hyperparameters were marginally increased to $ntree = 300$ and $mtry = 8$ in the hierarchical approach to achieve the most optimal results. Therefore, we set these two sets of hyperparameters in RF to report the prediction rate.

2.5.2 Support Vector Machine

The SVM (Vapnik & Lerner, 1963) uses a kernel function to create an optimal boundary (maximal margin hyperplane) that classifies the dataset in different regions. The maximal margin hyperplane is generated by maximizing the margins, or the distance of the vectors from the hyperplane, and the data points closest to the hyperplane (i.e., the support vector points that determine the hyperplane's position and orientation).

The function of kernel is to take data as input and transform it into the required form. In this study, we tried both linear and nonlinear kernel functions in the SVM classifier, including linear kernel, polynomial kernel, Gaussian radial basis function (RBF), and sigmoid kernel, to explore the underlying structure of process information. Table 3 presents the kernel functions and their corresponding equations and parameters to be estimated.

Table 3

Kernel Functions (Linear, Polynomial, RBF, and Sigmoid)

Kernel Function	Equation	Parameters
Linear	$k(x_i, x_j) = x_i x_j$	N/A
Polynomial	$k(x_i, x_j) = [coef + \gamma(x_i x_j)]^d$	$\gamma, d, coef$
RBF	$k(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$	γ
Sigmoid	$k(x_i, x_j) = \tanh(\gamma(x_i x_j) + coef)$	$\gamma, coef$

Note. x_i, x_j are observations in the dataset, $coef$ indicates coefficient, d indicates degree of polynomial.

Besides choosing the best-fit kernel function, two hyperparameters, regularization C and γ, also need to be carefully tuned to optimize the performance of the SVM classifier. The C regularization parameter represents how much misclassification of the training data is allowed in the model. By changing the regularization parameter, we can increase or decrease the error in classifying training data by changing the width of the margin. The γ parameter decides how much influence the data points at a certain distance from the hyperplane will have. If gamma is high, then nearby points will be considered. If gamma is low, far away points will have an influence too.

We used grid search cross-validation to tune the SVM hyperparameters, that is, to test all possible combinations of the values, C and γ under each kernel model, and get accuracies for each combination of hyperparameters and choose the one that performs the best. In the current study, we set $C = \{0.1, 1, 10, 100\}$, gamma = {1, 0.1, 0.01, 0.001} and the kernel function as {linear, polytomous, RBF, Sigmoid}. The hyperparameter tuning results showed the combination of $C = 0.1$, $\gamma = 0.1$ with $kernel = RBF$ produced the highest predictive accuracy rate. Therefore, we will fix this optimal setting for the SVM prediction report. This hyperparameter setting was retained in both flat and hierarchical approaches with the SVM model.

2.6 Analytic strategy

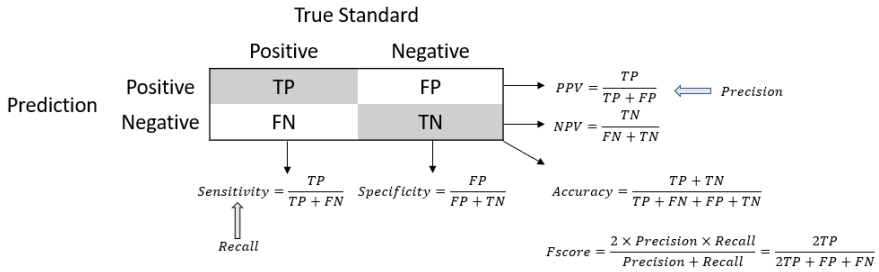
We first trained RF and SVM models affiliated with the flat approach by directly mapping the 35 process variables (five process variables by seven items in total) to four labels (Below Level 1, Level 1, Level 2, and Level 3). We then trained the two machine learning methods affiliated with the hierarchical classification by two levels. In the testing process, we input the test instances from the root and used the labels at the final leaf node as the final predicted label in both flat and hierarchical classification. However, the test data only needed to be predicted once at one level in the flat approach, whereas the data had to be predicted twice at two levels in the hierarchical approach.

Given concerns about the small sample sizes in the two extreme groups, Below Level 1 and Level 3, we employed a stratified nested cross-validation approach. Nested cross-validation has an outer loop with k folds for model evaluation and an inner loop that splits each of the k outer folds into l inner folds used for hyperparameter tuning. As suggested by Raschka (2018), for very small datasets, it is recommended to use a larger k in k-fold cross-validation for evaluating the generalization performance. We increased the parameter setting to $l = 20$ inner folds and $k = 10$ outer folds. It resulted in approximately 85% of data in the training set, 5% in the validation set for parameter tuning, and 10% in the test data to check the generalizability of the trained models.

The model evaluation was conducted on four classification settings: RF-flat, RF-hierarchical, SVM-flat, and SVM-hierarchical. We monitored six performance metrics, overall classification accuracy, sensitivity, specificity, PPV, NPV, and F-score derived from the confusion matrix (see Figure 3) alongside the area under the receiver operating characteristic curve (AUC ROC) values. All analyses were conducted in R version 4.1.1 (R Core Team, 2021) with R package `caret` (Kuhn, 2008) 6.0-93 version.

Figure 3.

Confusion matrix and derived evaluation metrics



3 Results

3.1 Statistical distributions of the process variables

We extracted 35 process variables as input features in prediction models. Table 4 reports the statistical distributions of process variables by each item. A further comparison between the high and low proficiency groups is plotted in Figure 4. On average, respondents spent at least two minutes completing one PSTRE item. The high proficiency group spent more time on difficult items (e.g., U02 and U11b) than the low proficiency group. On average, the low proficiency group spent 30 seconds longer solving the first two items than the high proficiency group, which may suggest that the low proficiency group requires more time to familiarize themselves with the testing environment. We also found that respondents on average engaged in at least 20 actions to solve one PSTRE item and used significantly more actions for challenging items. The high proficiency group usually used longer action sequences to solve each item relative to the low proficiency group, especially when the item involved multiple environments (e.g., U02) or required free text inputs (e.g., U16). We found that on average respondents spent more time initiating their first interaction at the beginning of the testlet and gradually reduced this time spent for the remaining items. Adults in the low proficiency group spent approximately 20 seconds longer than the high proficiency group on the first two items (e.g., U19a and U19b) to get acquainted with the task environment.

We also obtained two item-level sequence-based indicators, sequence similarity and efficiency, by computing the similarity between individual observed sequences and predefined action sequence(s). The high proficiency group exhibited higher similarity scores across all the items than the low proficiency group, suggesting that the action sequences used by the high proficiency group were usually closer to the predefined sequences. Interestingly, the low proficiency group exhibited higher efficiency scores than the high proficiency group on items with more complex designs and multiple environments (e.g., U02 and U23). This may indicate that adults with low proficiency intended to use limited actions without engaging in further explorations to arrive at a final solution compared to the high proficiency group.

Table 4

Statistical Distributions of Process Variables by PIAAC PSTRE Items in PS2

Item ID	Response Time (minutes)	Number of Actions	Time to the First Action (seconds)	Similarity	Efficiency
U19a	2.2 (1.4)	28.7 (19.3)	60.1 (55.8)	0.7 (0.2)	0.4 (0.1)
U19b	3.8 (2.3)	25.2 (24.7)	41.6 (27.8)	0.7 (0.2)	0.5 (0.2)
U07	2.2 (1.2)	20.6 (11.5)	40.8 (28.9)	0.7 (0.3)	0.6 (0.1)
U02	4.7 (3.4)	66.2 (86.9)	36.8 (82.6)	0.5 (0.2)	0.3 (0.2)
U16	2.8 (1.8)	116.4 (102.5)	29.5 (21.6)	0.6 (0.2)	0.2 (0.2)
U11b	2.0 (1.5)	36.0 (34.2)	20.9 (19.6)	0.7 (0.2)	0.4 (0.3)
U23	2.0 (1.9)	29.7 (42.2)	25.2 (18.5)	0.6 (0.3)	0.6 (0.2)

Note. Displayed values are means and standard deviations of process variables by each PSTRE item. The variable, number of actions, includes frequency of keystrokes.

3.2 Predictive accuracy

To address the first research question, we compared the accuracy prediction rate among the four models. Table 5 presents the predictive accuracy and AUC ROC of the two machine learning methods affiliated with two hierarchical classification settings based on the average of the ten-fold outer loop cross-validation. The prediction rate of process variables with the flat approach was satisfactory at 70%, and the AUC ROC was in the good range at around 82%. The predictive accuracy and AUC ROC were enhanced to 91% when the multiclass hierarchical approach was used in the classification. This result is relatively comparable to the findings in Tang et al. (2020), where information from process data demonstrated a predictive accuracy of over 88% on the success of solving a PSTRE item, and the out-of-sample correlations with literacy and numeracy proficiency scores were over 70%. No significant differences were found between the RF and SVM methods either affiliated with the flat or hierarchal approach. This suggests that these five process variables exhibited high predictability on respondents' PSTRE proficiency level. The predictive accuracy was found to be even higher (by 13%) when the prediction followed the hierarchical approach. To monitor the degree of overfitting, we compared the prediction results on both training and test sets. The prediction results were slightly higher in the training set, but within an acceptable difference range between the training and test set accuracy rate (<5%) (Tan et al., 2019). This suggests that although these four models showed marginally overfitting on the training set, the models were still promising in producing valid and accurate predictions.

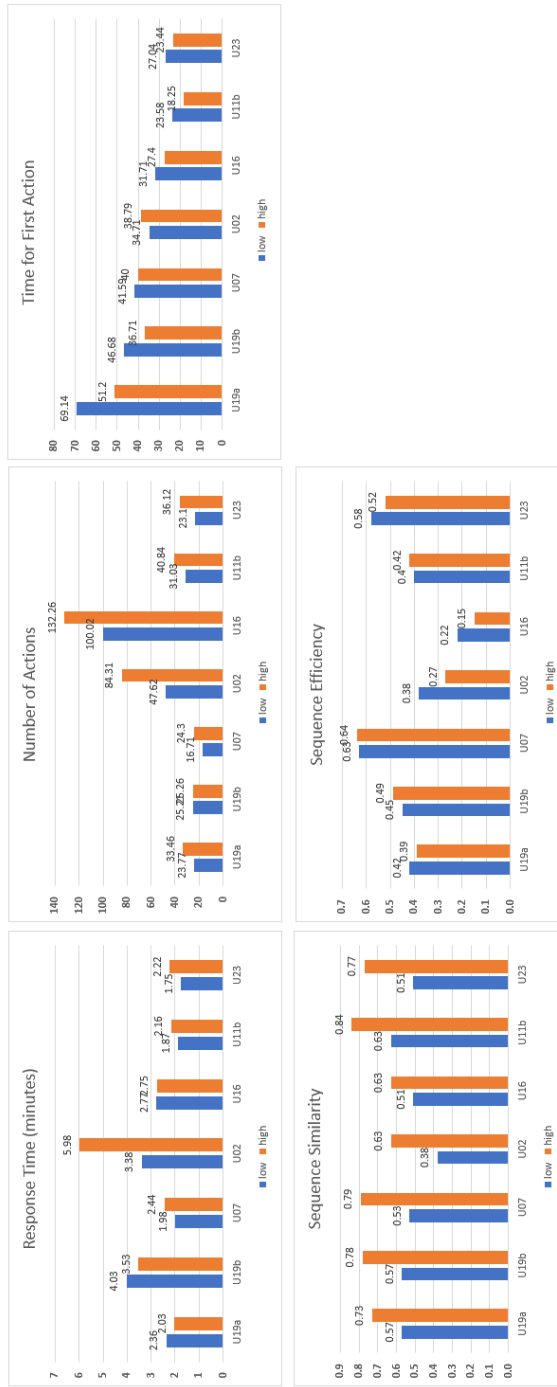


Figure 4.

Average values across five process variables by low and high PSTRE proficiency groups

Table 5

Predictive Accuracy of Training and Test Sets in Two Machine Learning Methods Affiliated with Two Hierarchical Settings

	RF-Flat	SVM-Flat	RF-Hierarchical	SVM-Hierarchical
Training Sets				
Accuracy	0.75 (0.02)	0.72 (0.01)	0.88 (0.03)	0.87 (0.02)
AUC ROC	0.84 (0.02)	0.82 (0.02)	0.93 (0.02)	0.93 (0.01)
Testing Sets				
Accuracy	0.70 (0.02)	0.70 (0.01)	0.84 (0.03)	0.83 (0.02)
AUC ROC	0.83 (0.02)	0.82 (0.02)	0.91 (0.02)	0.91 (0.01)

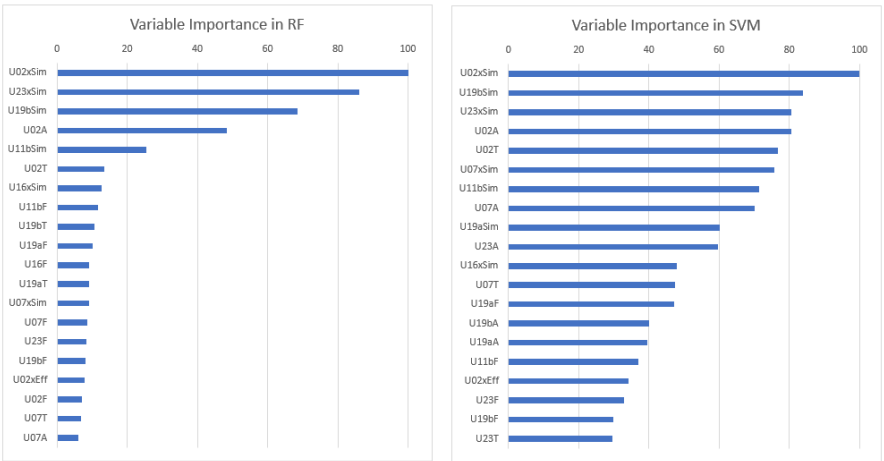
Note. AUC ROC indicates the area under the curve of the receiver operating characteristic curve. RF indicates random forest, and SVM indicates support vector machine. Displayed values are means and standard deviations across all ten outer folds cross-validation.

Figure 5 displays the top 20 variables that contributed to the classification in the RF and SVM methods, respectively. The measure of variable importance is based on the weighted sums of the absolute regression coefficients. The weights are a function of the reduction of the sums of squares across the number of partial least squares components and are computed separately for each outcome. Therefore, the contribution of the coefficients is weighted proportionally to the reduction in the sums of squares. All measures of importance are scaled to have a maximum value of 100. When using the RF, we found that the top three process variables were all related to sequence similarity, which suggests that compared with other variables, the indicator *similarity* was most informative in predicting respondents' proficiency level. This result also implies that three items (U02, U23, and U19b) were highly discriminative in distinguishing respondents' behavioral patterns and/or strategies during the problem-solving process. We also noticed that 7 out of 20 important variables were related to the *time to the first action* variable, suggesting that this variable was very important in every item in the classification. It also indicates that pause time (possibly for reading instructions) before conducting the first action could provide critical information to differentiate adults at various proficiency levels. The top 20 important variables extracted from the SVM classifier were mostly similar to the RF result; however, there were some differences in the ranking order. For example, the top three important variables extracted

in RF and SVM classifiers were the same, but the sequence similarity in U23 showed relatively higher importance in RF compared to SVM, in which sequence similarity in U19b ranked second. In addition, process variables related to U07 were ranked more important using the SVM classifier. However, these variables were found to be highly correlated and therefore, we had to remove the variables with overlapping information to avoid overfitting. Further, we found that the variables in general show higher importance weights using the SVM classifier than the RF classifier. The potential reason could be that for RF, the number of random variables for the hyperparameter tunings were set as small, fixed numbers (e.g., $ntry = 4\ or\ 8$) whereas there were no restrictions for SVM for the model training. Therefore, the SVM could benefit from using a range of variables to input for parameter tuning.

Figure 5.

Top 20 most important process variables using RF and SVM approaches



Note. The abbreviations at the end of each variable indicate: A for actions, T for response time, F for time to the first action, Sim for sequence similarity, and Eff for sequence efficiency.

We further examined variable dependency by computing bivariate correlations among the top 20 important variables. Among the 400 pairs, five pairs (1.5%) had a correlation higher than 0.5 (see Table 6). Interestingly, these high correlations were mostly found within one item (e.g., U07, U19) rather than across items. For example, the two variables U19aT (total response time in U19a) and U19aF (time to the first action in U19a) were highly correlated ($r = .725$). This makes sense because these two variables both contributed to the predictive model but may have overlapping information from the timing dimension (e.g., pause time before the first interaction made a large contribution to the total response time in U19a, which was the first item in PS2).

Table 6

Correlations Among the Top Important 20 Variables Selected in RF and SVM

	U19aT	U19aF	U19bT	U19bF	U19bSim	U07xA	U07xT	U07xF	U07xSim	U02xA	U02xT	U02xF	U02xSim	U02xEff	U16xF	U16xSim	U11bF	U11bSim	U23xF	U23xSim
U19aT	1																			
U19aF	**.725****	1																		
U19bT	.281**	.219**	1																	
U19bF	.245**	.196**	.451**	1																
U19bSim	-.096**	-.189**	-0.023	-.152**	1															
U07xA	0.001	-0.046	0.037	-0.034	.304**	1														
U07xT	.212**	.104**	.236**	.251**	.194**	**.736****	1													
U07xF	.238**	.106**	.247**	.262**	-0.029	-0.007	**.502****	1												
U07xSim	-0.03	-.081*	-0.023	-.065*	.384**	**.812****	**.583****	-0.032	1											
U02xA	-0.063	-.106**	0.023	-0.045	.201**	.131**	.077*	-0.021	.162**	1										
U02xT	.065*	-0.013	.166**	.083*	.299**	.293**	.380**	.125**	.332**	.409**	1									
U02xF	.068*	.070*	.068*	.075*	-0.013	0.001	.085**	.077*	-0.018	-0.019	.451**	1								
U02xSim	-.147**	-.180**	-0.058	-.163**	.490**	.350**	.214**	-0.061	.432**	.391**	.702**	0.037	1							
U02xEff	0.047	.091**	-.114**	0.046	-.308**	-.251**	-.167**	0.011	-.286**	-.470**	-.490**	0.047	-.493**	1						
U16xF	.271**	.159**	.246**	.237**	-.088**	-0.004	.200**	.194**	-0.015	-0.045	.104**	.065*	-.083*	0.052	1					
U16xSim	-.077*	-.130**	0.008	-.092**	.210**	.131**	.074*	-0.011	.188**	.157**	.194**	-0.01	.279**	-.263**	-0.007	1				
U11bF	.143**	.124**	.156**	.184**	-.112**	-0.056	.124**	.168**	-.078*	-.073*	0.029	0.049	-.128**	.095**	.184**	-.094**	1			
U11bSim	-.178**	-.167**	-.150**	-.190**	.366**	.254**	.098**	-.073*	.306**	.210**	.241**	-0.044	.419**	-.259**	-.114**	.251**	-.187**	1		
U23xF	.176**	.173**	.224**	.250**	-.124**	-0.025	.157**	.193**	-0.059	-0.04	.113**	0.05	-.071*	0.013	.219**	-0.052	.152**	-.120**	1	
U23xSim	-.154**	-.169**	-0.056	-.166**	.423**	.333**	.193**	-0.024	.409**	.175**	.326**	0.014	.488**	-.322**	-.080*	.272**	-.130**	.342**	-.111**	1

** Correlation is significant at the 0.01 level (2-tailed).

* Correlation is significant at the 0.05 level (2-tailed).

Note. The shaded cells indicate correlation $r \geq 0.5$.

To control for the dependency of input variables, we excluded four variables (U19aT, U07A, U07T, and U23T) from the highly correlated pairs and re-ran the prediction analysis. The predictive accuracy had no significant change using RF but there was a marginal increase of 0.01 using SVM. This result makes sense because the variables removed from the analysis had low importance ranks in RF, and therefore, may not contribute much to the prediction. However, the removed variables had higher importance ranks in SVM (e.g., U07A, namely, the number of actions in U07, ranked 20th in the variable importance in RF but ranked 8th in SVM) and thus, removing these may have had a higher impact on the predictive accuracy in SVM.

3.3 Prediction performance in multiclass hierarchical classification

To address the second research question, we further evaluated the prediction performance of the four models (RF-flat, RF-hierarchical, SVM-flat, and SVM-hierarchical) by each proficiency level. As shown in Figure 6, the RF and SVM performances are very similar and consistent across all proficiency levels. The major differences in performance between the flat and hierarchical approaches can be seen in the Below Level 1 group, in which the hierarchical approach resulted in two times higher sensitivity (0.62 in both RF and SVM) and F1 scores (0.67 in RF and 0.66 in SVM) compared to the flat approach. The PPV was also slightly enhanced from 0.67 to 0.73 when the hierarchical approach was employed. On the contrary, the specificity and NPV marginally dropped from 0.99 to 0.95 and 0.96 to 0.93, respectively, when the hierarchical classification approach was employed. These results indicate that the hierarchical approach performed significantly better than the flat approach in distinguishing the Below Level 1 group from the Level 1 group within the low proficiency group.

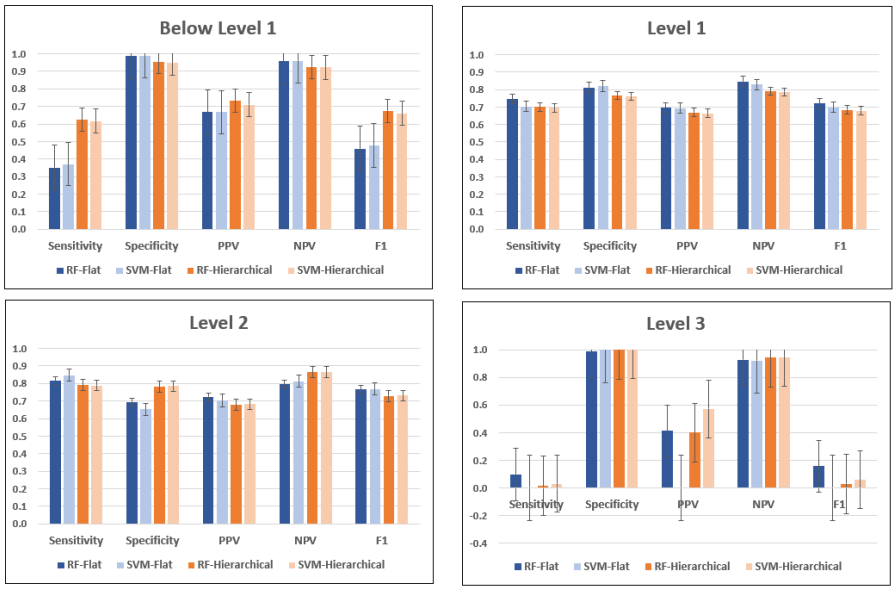
Comparatively, a marginal decrease was found across all performance metrics in Level 1 prediction when the hierarchical approach was employed. This implies that the better predictive accuracy in the hierarchical approach may have a larger contribution from the extremely low group with a bit of a trade-off from the Level 1 group. This also suggests that the behavioral patterns of the respondents in the Below Level 1 group are more distinguishable from those at Level 1 within the low proficiency group when using local classifiers, but the patterns might not be sufficiently robust from Level 2 and Level 3. This result echoes previous findings in which similar behavioral patterns were found in extremely high and extremely low PSTRE proficiency groups. However, these similar patterns were interpreted in completely different ways (He et al., 2021). For example, adults in the extremely low proficiency group may use longer action sequences for aimlessly clicking around, whereas those in the extremely high proficiency group may use longer action sequences to make more meaningful explorations to solve complex tasks.

We also found that the specificity and NPV resulted in marginally higher values in the hierarchical approach than in the flat approach for Level 2. This suggests a greater

capacity of excluding instances that were not in Level 2 using the hierarchical classification.

Figure 6.

Performance metrics by proficiency levels



Note. RF indicates random forest, and SVM indicates support vector machine. PPV indicates positive predictive value. NPV indicates negative predictive value.

Level 3 predictions showed the poorest results regardless of using the flat or hierarchical approach. The specificity and NPV were close to perfect values, and the sensitivity and F1 scores were extremely low. This implies that once an instance was labeled as Level 3, the confidence in the decision was extremely high to make a correct diagnosis. Nevertheless, it was very challenging to identify the adults in Level 3 based on the available process variables. This suggests that the behavioral patterns in Level 3 from the five process variables might not be sufficiently robust to distinguish this group.

In addition, Table 7 presents the prediction performance of the hierarchical approach at the first level (i.e., a binary classification of high and low proficiency groups). The accuracy was high, around 0.83 in both RF and SVM models. The AUC ROC was also excellent, at over 0.90. This result suggests that respondents' test-taking behaviors are sufficiently informative to screen respondents into high or low proficiency groups in addition to their final responses. The PPV and NPV were both higher than

0.8 at the first level, indicating that the confidence value was higher than the specified threshold for blocking at the first level, and there were no non-classified instances at the second level.

Table 7

Prediction Performance at the First Level (High/Low) in Hierarchical Classification

	RF	SVM
Accuracy	0.826 (0.02)	0.825 (0.01)
Sensitivity	0.780 (0.01)	0.770 (0.02)
Specificity	0.859 (0.01)	0.867 (0.01)
PPV	0.805 (0.02)	0.813 (0.03)
NPV	0.841 (0.02)	0.836 (0.02)
F1	0.792 (0.01)	0.790 (0.02)
AUC ROC	0.904 (0.02)	0.906 (0.01)

Note. Displayed values are means and standard deviations across all ten outer folds cross-validation. RF indicates random forest, and SVM indicates support vector machine. PPV indicates positive predictive value. NPV indicates negative predictive value. AUC ROC indicates the area under the curve of the receiver operating characteristic curve.

4 Discussion

In a technology-rich world, digital problem-solving skills are crucial to succeed in educational contexts and to meet the demands of 21st-century workplace environments. Process data provide deeper insights into respondents' test-taking behaviors, problem-solving strategies, and cognitive processes in learning and applying knowledge. In this study, we provide new evidence of the high predictability of process data to PIAAC PSTRE proficiency levels. We also illustrate how to extract meaningful process variables and conduct multiclass hierarchical classifications with machine learning methods to predict PSTRE proficiency levels using process data. Compared with the flat approach, the hierarchical classification approach was preferred to identify adults with extremely low PSTRE proficiency levels, which helped enhance the general predictive accuracy. Under the hierarchical structure, the problem-solving behavioral patterns were informative to distinguish the Below Level 1 and Level 1

groups, but did not perform well in identifying respondents with an extremely high proficiency level (Level 3). This implies greater variability in adults with low PSTRE proficiency levels. Low-skilled adults are a heterogenous population demographically that often struggle with basic foundational literacy and numeracy skills, basic computer literacy skills, and more complex problem-solving skills. These digital problem-solving skills are essential for active participation in today's workforce as well as daily living (e.g., responding to emails and using a spreadsheet for budgeting; Cummins et al., 2019; Vanek, 2017). Our results indicate that process data may provide more nuanced information on the strategies low-skilled adults use during the problem-solving process as well as potential breakdowns (e.g., hesitative behaviors, inefficient or repetitive behaviors) during the process. These results could inform instructional strategies that could be tailored to help low-skilled adults improve their problem-solving skills and test-taking strategies.

It is noted that the hierarchical classification approach follows a chaining decision procedure. Therefore, the error in the upper level may be propagated to the subsequent step. In this study, we set the confidence threshold as 0.8, marginally less than both the PPV and NPV at the first level. If the threshold is raised to 0.9 or even higher, we would expect an increased number of non-classifications in the second level. The instances with lower PPV or NPV at the first level (binary classification between high and low groups) would be retained in the first level and not passed to the second level. As a result, when excluding the non-classification cases in the second level, the purity of correct diagnosis is increased. Thus, sensitivity and specificity in the second level would also be enhanced. The set degree of confidence threshold is more dependent on the research purposes of a study and tolerance of the non-classification rate by different hierarchical levels.

Flat and hierarchical approaches have different pros and cons. Flat is more straightforward and would be preferable when the sample size is small and does not have a clear hierarchical structure. The obvious advantage of this approach is its simplicity. However, the cons are also apparent. For example, important information from the natural hierarchy of the data could have highly valuable classification; however, ignoring those parent-child class relationships could reduce prediction performance. The hierarchical approach, though a bit more complicated, may provide better prediction results, especially when there is a hierarchical relationship and a possibility for the variables to mix up. The local classifiers approach is highly intuitive and uses the hierarchy information in the data while retaining simplicity and generality. However, depending on the taxonomy and the method chosen, it may produce a rather bulky final model. There is also the problem of error propagation, which occurs when an error at one level could influence all the following ones. The results obtained in the current study also reiterate the findings from Silla and Freitas (2011), "it seems any hierarchical classification approach is overall better than the flat classification approach, when solving a hierarchical classification problem" (p. 2). Therefore, we recommend employing a hierarchical classification approach when the data structure does show a hierarchical relationship and choosing the simple flat approach when there is no clear hierarchical structure found in the dataset.

To avoid confusion in prediction performance, the nonresponse data were not included in the current study. In the current sample, 87% of nonresponses came from the Below Level 1 group. Therefore, the occurrence of nonresponses could be a robust classifier to distinguish the Below Level 1 group from the other three groups if we were to add this indicator to the prediction model. Further analysis revealed that the general predictive accuracy could be improved by five percentage points if the nonresponse behavior patterns (e.g., action sequence as "Start, Next, next_OK") were taken into consideration. It would be interesting to include the time interval between actions in the nonresponse behavioral patterns in future studies to better understand the potential reasons for nonresponse. For example, the respondents might quickly skip an item because of low engagement or spending more time reading and/or thinking but ultimately give up (Ulitzsch et al, 2022a). Analyzing potential nonresponses for the low-skilled group would also have important implications for understanding types of items that may be too challenging or demotivating, which has implications for test development.

There were two process variables that exhibited robust importance for prediction and may warrant further discussion. First, the sequence similarity variable ranked at the top of importance and had a high correlation with PSTRE proficiency in He et al. (2019a, 2021). The average sequence similarity value across all respondents may indicate how "easy" respondents found optimal solutions to solve an interactive item. A higher average value suggests that respondents more easily found optimal solutions, indicating that, on average, the action sequences are closer to the predefined sequences. Similar to the item parameters in item response modeling (Lord, 1980), the sequence similarity measure could be useful in estimating the degree of item complexity and discrimination based on test-taking behaviors and problem-solving strategies during the item solving process. This would provide supplementary information to support interactive item and test development. Thus, we recommend including the optimal action sequences predefined by item developers and content experts as a standard requirement for test development. This information would be beneficial for checking the item quality of interactive items, tracking respondents' problem-solving strategies, and providing meaningful prior information for latent ability estimates. Second, an aggregate-level process variable, time to the first action, emerged as a robust, important variable (in the top 20) in this study. This provides convincing evidence of its importance to proficiency level prediction. Compared with other aggregate-level variables, such as response time and number of actions, which are commonly used in process data analysis, we think that time to the first action warrants further attention in future studies. The time spent before the first action may indicate respondents' time spent on reading instructions and/or time spent on figuring out how to approach the complex task and thus, seems to further differentiate respondents based on proficiency levels.

There are some limitations that merit discussion. First, the sample size is relatively small and not balanced across proficiency levels. In particular, the sample size is much smaller in the extremely low (Below Level 1) and extremely high proficiency (Level 3) groups, which may impact classification accuracy in general. We recommend larger

sample sizes and including data from multiple countries in future studies. In this study, we took the imbalanced sample distribution into consideration by using the stratified sampling method when splitting our dataset in nested cross-validation. The imbalanced classification could also be handled by adopting alternatives that might result in better prediction. For example, the synthetic minority oversampling technique (SMOTE, Chawla et al., 2002) could augment data for the smaller class. In this approach, new examples can be synthesized from existing examples. SMOTE works by selecting examples that are close in the feature space, drawing a line between the examples in the feature space, and drawing a new sample at a point along that line. In addition, we only considered one country (the U.S.) in this study. For generalizability, future studies should use data from multiple countries and examine potential country effects using process data to predict PSTRE proficiency levels.

Second, the current study only considered five process variables per item in the analyses. Although the process variables showed robustness in the classification, more item-specific, fine-grained features, such as n-grams in action occurrences by different proficiency groups (He & von Davier, 2015, 2016), could be considered in future studies. In addition, the incorporation of cognitive variables (e.g., the time interval between actions and pause stages) may also be interesting to examine in future studies to improve the prediction model for adults' PSTRE proficiency levels, especially for the extremely high proficiency group.

Third, we only applied two commonly used machine learning methods with a focus on comparing the performance between flat and hierarchical approaches in different data structure settings. More robust machine learning methods, including XGBoost (Ulitzsch et al., 2022b) and neural networks (Zhu et al., 2016), could be explored in future studies.

In summary, the emergence of interactive item types and growing new analytic techniques are gradually shifting us away from traditional testing formats, both in terms of item development and the way items are scored. Process data are critical to understanding respondents' behaviors and strategies on interactive items and are considered the new forefront for future large-scale assessments. Further, recent new artificial intelligence applications, such as Chat GPT, bring new challenges to test validity, which may hinder our ability to know whether items are solved by machines or humans. This pressing concern reiterates the importance and urgency of incorporating process data into the test design process and evaluating respondents' latent ability by tracking their problem-solving process rather than using the final response (correct/incorrect) only. This study provides new evidence about the high predictability of process data and recommends hierarchical classification methods to predict problem-solving proficiency using process data. Future research should consider an intensive exploration of machine learning methods to incorporate process data into adaptive testing, challenging other researchers to improve the prediction methods for screening adults by different proficiency levels and integrating the findings from this study into an operational intake procedure.

Appendix A.
Algorithms to compute longest common subsequences.

The algorithm for identifying the Longest Common Sequence is defined as follows. Let $X = (x_1, x_2, \dots, x_i)$ and $Y = (y_1, y_2, \dots, y_j)$ be two sequences. x_i and y_j are actions contained in X and Y, respectively. X and Y are indexed as $X_{1,}\, X_{2,}, \dots, X_i$ and $Y_{1,}\, Y_{2,}, \dots, Y_j$, respectively. Let $LCS\,(X_i, Y_j)$ represents the set of longest common subsequence of prefixes X_i and Y_j. The set of sequences is given as:

$$LCS(X_i, Y_j) = \begin{cases} \emptyset,\ if\ i = 0\ or\ j = 0 \\ LCS(X_{i-1}, Y_{j-1})\hat{}x_i,\ if x_i = y_j \\ \max\left(LCS(X_i, Y_{j-1}), LCS(X_{i-1}, Y_j)\right),\ if\ x_i \neq y_j \end{cases} \quad (1)$$

To find the LCS of X_i and Y_j, compare x_i and y_j. If they are equal, then the sequence $LCS\,(X_{i-1}, Y_{j-1})$ is extended by that element, x_i. If they are not equal, then the longer of the two sequences, $LCS\,(X_i, Y_{j-1})$ and $LCS\,(X_{i-1}, Y_j)$ is retained. The length of LCS is defined as:

$$\text{length}\left(LCS(X_i, Y_j)\right) = \begin{cases} 0,\ if\ i = 0\ or\ j = 0 \\ \text{length}(i-1, j-1) + 1,\ if x_i = y_j \\ \max\left(\text{length}(i, j-1), \text{length}(i-1, j)\right), if\ x_i \neq y_j \end{cases} \quad (2)$$

References

Breiman, L. (2001). Random forests. *Machine learning, 45*, 5–32.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321–357.

Chen, Y., Li, X., Liu, J., & Ying, Z. (2019). Statistical analysis of complex problem-solving process data: An event history analysis approach. *Frontiers in Psychology, 10*, 486

Cummins, P., Yamashita, T., Millar, R., & Sahoo, S. (2019). Problem-solving skills of the U.S. workforce and preparedness for job automation. *Adult Learning, 30*(3), 111–120.

de Boeck, P., & Scalise, K. (2019). Collaborative problem solving: Processing actions, time, and performance. *Frontiers in Psychology, 10*, 1280.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation, 10*(7), 1895-1923.

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning, 40*, 139–157.

Eichmann, B., Greiff, S., Naumann, J., Brandhuber, L., & Goldhammer, F. (2020). Exploring behavioural patterns during complex problem-solving. *Journal of Computer Assisted Learning, 36*(6), 933–956.

Engelhardt, L., Naumann, J., Goldhammer, F., Frey, A., Wenzel, S. F. C., Hartig, K., & Horz, H. (2019). Convergent evidence for the validity of a performance-based ICT skills test. *European Journal of Psychological Assessment, 36*, 269–279.

Gao, Y., Cui, Y., Bulut, O., Zhai, X., & Chen, F. (2022). Examining adults' web navigation patterns in multi-layered hypertext environments. *Computers in Human Behavior, 129*, 107142.

Goldhammer, F., Naumann, J., & Keβel, Y. (2013). Assessing individual differences in basic computer skills: Psychometric characteristics of an interactive performance measure. *European Journal of Psychological Assessment, 29*(4), 263–275.

Goldhammer, F., Naumann, J., Stelter, A., Tóth, K., Rölke, H., & Klieme, E. (2014). The time on task effect in reading and problem solving is moderated by task difficulty and skill: Insights from a computer-based large-scale assessment. *Journal of Educational Psychology, 106*(3), 608–626.

Han, Z., He, Q., & von Davier, M. (2019). Predictive feature generation and selection using process data from PISA interactive problem-solving items: An application of random forests. *Frontiers in Psychology, 10*, 1421.

Hao, J., Shu, Z., & von Davier, A. (2015). Analyzing process data from game/scenario-based tasks: An edit distance approach. *Journal of Educational Data Mining, 7*(1), 33–50.

He, Q., Borgonovi, F., & Paccagnella, M. (2019a). 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, Paris, https://doi.org/10.1787/650918f2-en.

He, Q., Borgonovi, F., & Paccagnella, M. (2021). Leveraging process data to assess adults' problem-solving skills: Identifying generalized behavioral patterns with sequence mining. *Computers & Education*, *166*, 104170.

He, Q., Borgonovi, F., & Suárez-Álvarez, J. (2022). Clustering sequential navigation patterns in multiple-source reading tasks with dynamic time warping method. *Journal of Computer-Assisted Learning*. https://doi.org/10.1111/jcal.12748

He, Q., Liao, D., & Jiao, H. (2019b). Clustering behavioral patterns using process data in PIAAC problem-solving items. In B. P. Veldkamp and C. Sluijter (eds.), *Theoretical and Practical Advances in Computer-based Educational Measurement* (pp. 189–212), Springer.

He, Q., Veldkamp, B. P., Glas, C. A. W., & de Vries, T. (2017). Automated Assessment of Patients' Self-Narratives for Posttraumatic Stress Disorder Screening Using Natural Language Processing and Text Mining. *Assessment, 24*(2), 157–172.

He, Q., Veldkamp, B. P., & Vries T. de. (2012). Screening for posttraumatic stress disorder using verbal features in self-narratives: A text mining approach. *Psychiatry Research, 198*(3), 441–447.

He, Q., & von Davier, M. (2015). Identifying feature sequences from process data in problem-solving items with n-grams. In A. van der Ark, D. Bolt, S. Chow, J. Douglas & W. Wang (Eds.), *Quantitative Psychology Research: Proceedings of the 79th Annual Meeting of the Psychometric Society* (pp.173–190). New York: Springer.

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. 749–776). Hershey, PA: Information Science Reference.

Janitza, S., & Hornung, R. (2018). On the overestimation of random forest's out-of-bag error. *PloS One, 13*(8), e0201904.

Koller, D., & Sahami, M. (1997). Hierarchically classifying documents using very few words. *International Conference on Machine Learning, 97*, 170–178.

Kuhn, M. (2008). Building predictive models in R using the caret package. *Journal of Statistical Software, 28*, 1–26.

Liao, D., He, Q., & Jiao, H. (2019). Mapping background variables with sequential patterns in problem-solving environments: An investigation of U.S. adults' employment status in PIAAC. *Frontiers in Psychology, 10*, 646.

Lord, F. M. (1980). *Applications of Item Response Theory to Practical Testing Problems*. Erlbaum: Hillsdale.

Lu, J., Wang, C., Zhang, J., & Tao, J. (2020). A mixture model for responses and response times with a higher-order ability structure to detect rapid guessing behaviour. *British Journal of Mathematical and Statistical Psychology, 73*(2), 261–288.

Mitchell, M. (2011). Bias of the random forest out-of-bag (OOB) error for certain input parameters. *Open Journal of Statistics, 1,* 205–211.

OECD. (2009). PIAAC problem solving in technology-rich environments: A conceptual framework. *OECD Education Working Paper No. 36.* OECD Publishing.

OECD. (2012). *Literacy, Numeracy and Problem Solving in Technology-Rich Environments: Framework for the OECD Survey of Adult Skills.* OECD Publishing. https://doi:10.1787/9789264128859-en.

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

OECD. (2016). *Technical Report of the Survey of Adult Skills (PIAAC)* (2nd ed.). Paris: OECD Publishing

OECD. (2017). *PISA 2015 technical report*. OECD Publishing. Paris, France. https://www.oecd.org/pisa/sitedocument/PISA-2015-technical-report-final.pdf

Qiao, X., Jiao, H. & He, Q. (2022), Multiple-group joint modeling of item responses, response times, and action counts with the Conway-Maxwell-Poisson distribution. *Journal of Educational Measurement*. DOI:10.1111/jedm.12349

Raschka, S. (2018). Model evaluation, model selection, and algorithm selection in machine learning. In arXiv [cs.LG]. https://doi.org/10.48550/arXiv.1811.12808

R Core Team (2021). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.

Schleicher, A. (2008). PIAAC: A new strategy for assessing adult competencies. *International Review of Education, 54*, 627–650.

Silla, C. N., & Freitas, A. A. (2011). A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, *22*, 31–72.

Stadler, M., Fischer, F., & Greiff, S. (2019). Taking a closer look: An exploratory analysis of successful and unsuccessful strategy use in complex problems. *Frontiers in Psychology, 10*, 777.

Sukkarieh, J. Z., von Davier, M., & Yamamoto, K. (2012). From biology to education: Scoring and clustering multilingual text sequences and other sequential tasks. *Research report No. RR-12-25*. Educational Testing Service.

Tan, P. N., Steinbach, M., & Kumar, V. (2019). *Introduction to data mining*. Pearson Education India.

Tang, X., Wang, Z., He, Q., Liu, J., & Ying, Z. (2020). Latent feature extraction for process data via multidimensional scaling. *Psychometrika, 85*(2), 378–397.

Ulitzsch, E., He, Q., & Pohl, S. (2022a). Using sequence mining techniques for understanding incorrect behavioral patterns on interactive tasks. *Journal of Educational and Behavioral Statistics, 47*(1), 3–35.

Ulitzsch, E., He, Q., Ulitzsch, V., Nichterlein, A., Molter, H., Niedermeier, R., & Pohl, S. (2021). Combining clickstream analyses and graph-modeled data clustering for identifying common response processes. *Psychometrika*, *86*, 190-214..

Ulitzsch, E., Ulitzsch, V., He, Q., & Lüdtke, O. (2022b). A machine learning-based procedure for leveraging clickstream data to investigate early predictability of failure on interactive tasks. *Behavior Research Methods*. https://doi.org/10.3758/s13428-022-01844-1

Vanek, J. (2017). *Using the PIAAC Framework for Problem Solving in Technology-Rich Environments to Guide Instruction: An Introduction for Adult Educators. Retrieved from:* [https://edtech.worlded.org/wpcontent/uploads/2017/09/PSTRE_Guide_Vanek_2017.pdf]Washington, DC.

Vapnik, V. & Lerner, A. (1963). Pattern recognition using generalized portrait method. *Automation and Remote Control, 24*, 774—780.

von Davier, M., Khorramdel, L., He, Q., Shin, H., & Chen, H. (2019). Developments in psychometric population models for data from innovative items. *Journal of Educational and Behavioral Statistics, 44*(6), 671–705.

Xiao, Y., He, Q., Veldkamp, B. P., & Liu, H. (2021). Exploring latent states of problem-solving competence using hidden Markov modeling on process data. *Journal of Computer-Assisted Learning, 37*(5), 1232–1247.

Xiao, Y., & Liu, H. (2023). A state response measurement model for problem-solving process data. *Behavior Research Methods*, https://doi.org/10.3758/s13428-022-02042-9

Zhang, S., Wang, Z., Qi, J., Liu, J., & Ying, Z. (2022). Accurate assessment via process data. *Psychometrika*, *88*(1), 76–97.

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.