Applying Psychometric Modeling to aid Feature Engineering in Predictive Log-Data Analytics: The NAEP EDM Competition

Fabian Zehner
DIPF*, ZIB†
fabian.zehner@dipf.de

Beate Eichmann
DIPF*, ZIB†
beate.eichmann@dipf.de

Tobias Deribo
DIPF*
deribo@dipf.de

Scott Harrison
DIPF*, ZIB†
harrison@dipf.de

Daniel Bengs
DIPF*
bengs@dipf.de

Nico Andersen
DIPF*
andersen.nico@dipf.de

Carolin Hahnel
DIPF*, ZIB†
hahnel@dipf.de

The NAEP EDM Competition required participants to predict efficient test-taking behavior based on log data. This paper describes our top-down approach for engineering features by means of psychometric modeling, aiming at machine learning for the predictive classification task. For feature engineering, we employed, among others, the Log-Normal Response Time Model for estimating latent person speed, and the Generalized Partial Credit Model for estimating latent person ability. Additionally, we adopted an n-gram feature approach for event sequences. Furthermore, instead of using the provided binary target label, we distinguished inefficient test takers who were going too fast and those who were going too slow for training a multi-label classifier. Our best-performing ensemble classifier comprised three sets of low-dimensional classifiers, dominated by test-taker speed. While our classifier reached moderate performance, relative to the competition leaderboard, our approach makes two important contributions. First, we show how classifiers that contain features engineered through literature-derived domain knowledge can provide meaningful predictions if results can be contextualized to test administrators who wish to intervene or take action. Second, our re-engineering of test scores enabled us to incorporate person ability into the models. However, ability was hardly predictive of efficient behavior, leading to the conclusion that the target label's validity needs to be questioned. Beyond competition-related findings, we furthermore report a state sequence analysis for demonstrating the viability of the employed tools. The latter yielded four different test-taking types that described distinctive differences between test takers, providing relevant implications for assessment practice.

Keywords: log files, psychometric models, domain knowledge–based feature engineering, process data, state sequence analysis, clustering, latent state, ensemble

*DIPF | Leibniz Institute for Research and Information in Education, Frankfurt am Main
†Centre for International Student Assessment (ZIB), Frankfurt am Main

1. INTRODUCTION

With the 2nd Annual EDM Data Mining Challenge from the Big Data for Education Spoke of the NSF Northeast Big Data Innovation Hub, also called the Nation's Report Card Data Mining Competition 2019,[1] its organizing consortium continued a young series of data competitions featured by the International Educational Data Mining Society. The data challenge consisted in predicting the efficiency of students' test-taking behavior in a second test part using the log data produced in the first test part. The organizer's goal was to identify students who had acted inefficiently by rushing through the second test half or by not reaching its end (Baker et al., 2019). Another central, and noticeably constraining, secondary goal was that accurate classification should be reached as early as possible during test administration (i.e., with as little log data as possible; Baker et al. 2019).

We regard log data captured during test administration as process data, which means it constitutes "empirical information about the cognitive (as well as meta-cognitive, motivational, and affective) states and related behavior that mediate the effect of the measured construct(s) on the task product" (Goldhammer and Zehner, 2017, p. 128). Thus, log data from assessment contexts is not just a nice-to-have by-product, but carries relevant information that can be drawn on for purposes such as the one promoted in the competition.

This paper reports details on our contribution to the competition, which was characterized by the integration of established psychometric models, and extends a proceedings paper by additional empirical analyses (Zehner et al., 2020).[2] Opposed to bottom-up, data-driven analyses, a literature-informed, top-down approach is characterized by identifying potential mechanisms at play and accordingly selecting methods, features, or both from a conceptual perspective. Our experience with log data emphasizes the importance of a focus on deductively informed feature engineering over presumably more predictive deep learning or other black-box methodology. We believe that the theoretical understanding of underlying behavioral and cognitive processes that drive characteristics of test-taking behavior, such as efficiency, is crucial for building robust predictive models in the educational domain. Otherwise, the risk of integrating spurious associations into productive classifiers is high; for example, in the form of shortcut learning (Geirhos et al., 2020). With respect to classification performance, our competition contribution ended up in the top quarter of leaderboard submissions and was ranked eighth among the teams that submitted their code in time on the final scoreboard (Baker et al., 2020).

In this paper, we furthermore present evidence that the validity of the data challenge's target label needs to be reassessed. Though the provided competition data did not include item scores, we were able to derive estimates for students' ability by re-constructing scores from log data. This way, we could show that students' ability was hardly associated with the target label. Efficiency can be defined as the characteristic of producing *desired results without waste* (Merriam-Webster, 2021). In the context of efficient test taking, this corresponds to successful test taking with minimum effort or time. Obviously, this involves two components, namely goal-reaching and resource-saving. As we elaborate on in detail throughout the following sections, the competition's operationalization of efficiency strongly emphasizes the latter component, *economy*, and largely neglects the former, *success*. This consideration is emblematic and shows the value of a top-down access to the matter.

[1] http://tiny.cc/CompAIED [2020-02-29]

[2] Our competition contributions have been submitted under the name *Team TBA* (Centre for Technology-Based Assessment — DIPF).

As a final contribution of this paper, we report on an in-depth analysis of test-taking types, based on test takers' states during item completion. The analysis continues the top-down approach from the competition-driven predictive modeling by identifying and operationalizing relevant states according to two response process theories. Test-taker states here are specified by context-aware log events during item completion and analyzed with respect to their temporal changes during test administration. The employed methods are capable of capturing information of individuals' interaction at the item level, but at the same time allow aggregating those at a large scale. This way, this addition shows how log data can be used for in-depth analyses that still scale up to mine large semi-structured data and, at the same time, broaden the scope of analysis beyond a single target label.

In the following, the paper first describes the setup provided by the competition organizers (Section 2), then describes the theoretical foundation of the state sequence analysis (Section 3) and focuses on our approach for feature engineering and classifier training (Section 4), reporting and discussing results on the classifier's performance level as well as single features' predictivity and findings on the state sequence analysis (Sections 5 and 6). The conclusion (Section 8) elaborates on the definition of efficient test taking and discusses the state of the art for corresponding operationalizations.

Please note that we use the term *item* here whenever we refer to single questions or tasks that constitute the smallest scored assessment unit in the NAEP test.

2. Competition Setup

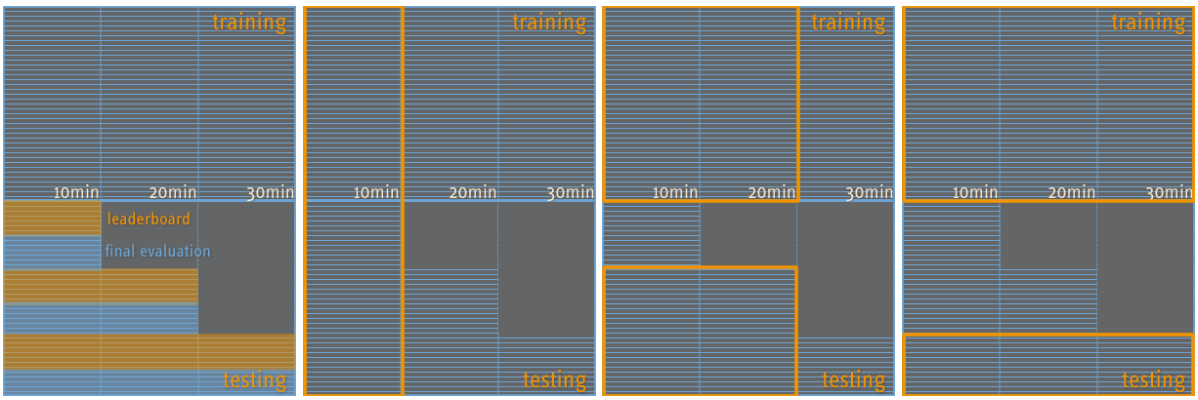
2.1. Data

The National Assessment of Educational Progress (NAEP), which is a biennial US national assessment conducted across 4th-, 8th-, and 12th-grade students and includes tests on a variety of subjects, provided the competition data set (Baker et al., 2019). Specifically, the data set stems from the 2017 test for 8th-grade students in mathematics. The test comprised two test blocks (Block A and B) that were time limited to 30 min per block.

NAEP 2017 mathematics assessment was digitally administered on tablet computers with keyboards (National Assessment Governing Board, 2017). Test items covered several domains, such as algebra or geometry, and were either presented as pure mathematics tasks or tasks applied in an everyday context. Items included stimulus material (text and/or figures) and either a list of responses to choose from (multiple choice), drag and drop response elements, or one or more text fields for constructed responses.

Block A consisted of a fixed test with 19 items, including 14 multiple-choice, 1 selected-response (drag and drop), and 4 constructed-response items. Two of the constructed-response items were scored polytomously with Full, Partial, or No Credit, the other seventeen dichotomously with only Full or No Credit. Several of the items are publicly released and can be found in the NAEP Question Tool (`https://nces.ed.gov/nationsreportcard/nqt/`).

Participating students could navigate liberally between items within the same test block. However, the large majority of students went linearly through the test. 84 percent attempted all items, 4 percent did not see one or two items, and the other 12 percent more than two items (with only the 10^{th} percentile at 4, and only the 5^{th} percentile at 6 non-visited items). Students who did not complete all items were substantially more likely to not reach items later in the test than skipping items earlier on (with 15%, 13%, 12%, and 11% of missing responses for the last



(a) Provided Data Setup (b) Tweaked Training Set 10min (c) Tweaked Training Set 20min (d) Tweaked Training Set 30min

Figure 1: Sketch of Provided Data and Subsets of Data Actually Used for Training. The horizontal aspect represents time into the test, the vertical aspect represents students.

4 items, respectively).

For some items, a virtual calculator was available. The test environment also included a virtual drawing tool that allowed students to draw freely and make handwritten annotations, a text-to-speech function that allowed students to listen to all written task materials, and a help button that provided information on how to use the test environment. The students could navigate from the review screen back to the items and onward to the next test block. Students' actions in the virtual environment (i.e., clicking on elements or keystrokes when typing a response) were recorded as log data events. The data provided 42 such events (e.g., `Enter Item`, `Click Choice`, `Back`). From this log data, it is possible to reconstruct the rough course of each student's test completion behavior. At the same time, this description of the action space also points out that students' interactions with the assessment system are mainly test-flow centered (on a meta-cognition level) rather than item-induced due to tasks involving complex interaction objects (e.g., such as in computer-based scientific simulations that require interactions with multiple sliders and buttons). Please note that the NAEP test does contain scenario-based tasks with more complex within-item interactions, which, however, were not included in the provided data set.

For the competition, the organizers arranged the log data into two sets (see Figure 1a). The first one was a training set (upper half in Figure 1a), taken from Block A with log data for $n_{train} = 1{,}232$ students across the whole 30 min of Block A. This data subset was accompanied by the target label indicating whether students behaved efficiently in Block B. The underlying data from Block B for classifying students as efficient or inefficient has neither been provided for the competition nor afterwards. The second data set was an evaluation (testing) set (lower half in Figure 1a) for which the efficiency labels had to be predicted by the contestants. This evaluation set comprised log data stratified into three conditions: i) $n_{10min} = 411$ students with *10 min of log data* from the start of Block A (first column in the lower half of Figure 1a); ii) $n_{20min} = 411$ students with *20 min of log data* from the start of Block A (first two columns in the lower half of Figure 1a); iii) $n_{30min} = 410$ students with *complete 30 min* of Block A (all three columns in the lower half of Figure 1a). The competition organizers, in turn, halved the evaluation set so that the leaderboard displayed teams' prediction accuracy on one half of the evaluation set (orange

areas in Figure 1a), and final evaluation for the final scoreboard was carried out on the remaining half (blue areas in Figure 1a). The training and evaluation sets consisted of 438,291 and 301,924 event logs, respectively. Figures 1b–1d sketch how we rearranged the provided data sets for capitalizing on the maximum information for time condition–specific feature engineering and classifier training (see Section 4.1).

2.2. Target Label: Efficiency

The competition organizers categorized the students into two groups: Students who completed Block B efficiently or inefficiently. Students were labeled efficient when they met two criteria: "1) being able to complete all problems in Block B, and 2) being able to allocate a reasonable amount of time to solve each problem [in Block B]" (Baker et al., 2019).

The definition of efficiency captures two key test-taking behaviors: when students *go too slow*, and as such fail to complete all the items in a block, and when students *go too fast* through the test, therefore not spending enough time on each item. Students who are inefficient through being too slow can easily be identified due to their failure to complete all items. However, for students going too fast, "a reasonable amount of time" can be difficult to operationalize. As such, the organizers chose to impose an arbitrary threshold for which students were evaluated on the total time taken on an item, with "the 5th percentile as the cut-off for the 'reasonable amount of time'" (Baker et al., 2019). With this operationalization, 39.6 percent of students in the training data were labeled as inefficient.

2.3. Evaluation Metrics

The prediction was evaluated against two key measures: adjusted AUC and adjusted kappa, with $\mathrm{AUC}_{adj} = (\mathrm{AUC} - 0.5) * 2$ and κ_{adj} transforming Cohen's Kappa (Cohen, 1960) in that its lower limit was set to 0, $\{\kappa_{adj} \in \mathbb{R} | 0 \leq \kappa_{adj} \leq 1\}$.

For the evaluation of the models within the competition, AUC_{adj} and κ_{adj} formed an aggregated score.

3. Test-Taking States

Going beyond the competition task, in-depth analyses on test-taking behavior provide valuable information both for assessment designers as well as assessment and educational practitioners. For this, we identified states that test takers run through during test—and more specifically item—completion.

When test takers work on a test, they perform a complex series of cognitive and metacognitive operations. Leading to the goal specified by a test's instructions (e.g., *Try to answer as many questions correctly as possible*), these steps can be summarized into more or less fine-grained phases. With the requirement to generalize to various item types of the NAEP mathematics test, we chose a coarse abstraction of three relevant on-task states: *Processing*, *Responding*, and *Reviewing*. In the following, we depict the rough contours of the theoretical foundation for defining these test-taking states, combining two established response process models.

In their survey response process model, Tourangeau et al. (2009) describe the components playing a part when survey respondents react to questions. The model distinguishes four such components: comprehension, retrieval, judgment, and response. That is, respondents first have to make sense of the task or question at hand. Second, based on their comprehension of the

question's content and articulated goal, respondents aim at retrieving relevant information; for example, by actively searching for it or retrieving it from memory. In turn, the resulting information serves as the basis for selecting information for the intended response, with the respondent drawing inferences on top where necessary. Finally, the respondent translates their mental intention of responding into an observable reaction such as selecting or producing a response.

While this response process model explicates (reciprocal) phases that can inform our definition of test-taking states, Graesser and Black (2017) describe a response process model that is rather motivated from a cognitive psychological background and republishes their seminal publication in the 1980s. Their model distinguishes five components that contribute to a respondent's actions of answering a question: interpret question, identify question category, apply to knowledge structures, articulate answer, and evaluate goals. In a nutshell, this model proposes that respondents start with interpreting the question, parsing it, and extracting its semantic focus. Next, they identify the question category by determining the type of required response; for example, *why*-questions aim for causal relationships and *who*-questions for concept completions. The identified semantic focus and question category are then applied to knowledge structures which the respondent retrieves from memory or builds up by retrieving further external information. Most crucially, the semantic focus and question category govern the filtering of information in order to narrow information processing down to seemingly relevant knowledge structures and statement nodes. The filtered and selected statement nodes are then articulated in a response (for details see the QUEST model; Graesser and Franklin 1990) or, more generically, translated into a reaction by interacting with the (assessment) environment. Finally, and importantly, respondents will evaluate their goal reaching, potentially re-iterating previous processes for adjustment or refinement.

Quite obviously, the two response process models align to and complement each other. Based on the structure of the NAEP items and the available log data, and with assessment as the field of application here, our analysis discriminates three phases according to the two described models. First, when respondents are confronted with a task in an educational test, they find themselves in the state of *Processing*. The two response models describe the manifold and complex cognitive processes that take place in this state in great detail, for example by referring to reading the stimulus materials and question, understanding them, and building a mental model for an intended reaction (i.e., response). While the log data of single items might grant the possibility to further decompose this conglomerate of actions, the goal of applicability to different item types across the NAEP math test requires a coarse-grained perspective and generic taxonomy. Second, test takers translate their mental intention into action by giving a response, here called *Responding*. In the two models mentioned above, this state relates to the components of *response* and *articulate answer*, respectively. Third and last, after test takers have provided their final response, they might reassess whether they have reached their goal with their response, here called *Review*. This state is only implied in Tourangeau et al. (2009)'s model during the response phase, but more explicitly reflected and distinguished in Graesser and Black (2017)'s *evaluate goals* component. Section 4.5 further specifies how these identified states are operationalized in the context of the competition's log data.

4. METHODS

In this section, we first describe a data transformation step of splitting the three temporal conditions for feature extraction and training. This turned out to be essential for achieving appropriate

classifier generalizability to the test set. Next, we describe our feature engineering as well as restrictive feature selection and outline how all those strings were pulled together for building an ensemble classifier for prediction. We close this section by describing the operationalizations for extracting and analyzing test-taking states.

All competition-related statistical analyses have been carried out using *R 3.6.1* (R Core Team, 2020), with the package *mlr 2.17.0* (Bischl et al., 2016) for machine learning, *TAM 3.3-10* (Robitzsch et al., 2019) for item difficulty and person ability estimation, and *LNIRT 0.4.0* (Fox et al., 2019) for item time intensity and person speed estimation. For the test-taking state sequence analysis, *R 4.0.3* (R Core Team, 2020) has been used, together with *TraMineR 2.2-1* for optimal matching and state sequence visualizations (Gabadinho et al., 2011) as well as *WeightedCluster 1.4-1* for clustering (Studer, 2013).

4.1. Improving Generalizability by Separating Conditions

Our early submissions of predictions to the leaderboard revealed that the classifiers' performance—though evaluated by stratified, repeated cross-fold validation—would always decrease substantially when being evaluated on the test set. That is, the generalizability of these classifiers to the test set was low, even when cross-validations testified to stable out-of-sample classification.

The primary reason that we identified was that the training set contained 30 min of log data, whereas the test set was split into three conditions with only the first 10 min, 20 min, or the full 30 min of log data available (see Section 2.1). Obviously, it is reasonable that feature realizations and their indication for one class vary over (testing) time. As an example, the time students take to work on single items does not only vary by item characteristics but is also influenced by the item's position within the test. Another example is indicated by the log event of the timeout screen that limits students' time to 30 min. Naturally, this event is reasonably predictive, but while it is available in the 30 min condition, it is not in the 10 min or 20 min condition. Therefore, training sets tailored to each condition were necessary for the classifiers to generalize properly to the test set.

For this purpose, we created three data sets (again, see Figures 1b–1d): (i) the first 10 min of log data from the 10, 20, and 30 min conditions for predicting test set cases with 10 min of log data (highlighted area in Figure 1b), (ii) the first 20 min of log data from the 20 and 30 min conditions for test-set cases with 20 min of log data (highlighted area in Figure 1c), and (iii) the full 30 min of log data for test-set cases with 30 min of log data (highlighted area in Figure 1d). For feature extraction, we combined the respective training and test (sub)sets. This way, we maximized the available information for norm-referenced features and parameter estimation procedures. Since we employed supervised learning methods for the competition's prediction task, the test sets were excluded from classifier training.

The result of splitting the conditions was that we constructed three classifiers for each learning method and set of features. Each case in the test set, however, was classified by only one model, determined by the condition the test case belonged to.

4.2. Feature Engineering

In this section, we describe the selection of engineered features of which some ended up in at least one of the base classifiers that contributed to the final ensemble. We start with the two crucial psychometric models used for estimating students' speed and ability. Then, we describe our approach of extracting features from log data and deriving simple indicators that we assumed

would indicate efficient or inefficient test behavior, using the software package *LogFSM*. Finally, we describe the concept and operationalization of rapid guessing as well as an adopted technique for representing log event sequences.

4.2.1. Latent Test-Taker Speed

Efficient test taking as operationalized in the competition (see Section 2.2) is mainly characterized by test takers' time handling. If a student went relatively quickly through the test (in Block B), they were labeled as inefficient. If a student spent too much time on some item (in Block B), they would not be able to complete all items and thus be labeled as inefficient, too. Therefore, the most evident feature is test-taker speed.

Test-taker speed can be inferred from the time spent on items in a test. However, the time spent on an item is determined by the characteristics of the item and the test-taker. On the one hand, item characteristics, such as complexity, require and evoke a shorter or longer time on task due to the item's inherent *time intensity*. On the other hand, some test takers will have the tendency or skill to move faster through a test than others; this characteristic is called *test-taker speed*. Both time intensity and test-taker speed are not directly observable and can only be estimated as latent variables.

A model that allows the separation of time on task into item and person parameters is the *Lognormal Response Time Model* (van der Linden, 2006):

$$f(t_{ip};\tau_p,\alpha_i,\beta_i) = \frac{\alpha_i}{t_{ip}\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left[\alpha_i(\ln t_{ip} - (\beta_i - \tau_p))\right]^2\right\} \tag{1}$$

Response time distributions take values in the positive reals and typically have long tails. The log-transformation hence is a sensible way to approximate normality and is expected to lead to better fit than a normal model on raw response times (van der Linden, 2006). The lognormal model takes three parameters into account and is based on the log-transformed time t_{ip} that person p spent at item i. Item time intensity β_i captures item i's tendency to evoke more or less time spent for completing it. Test-taker speed τ_p is a person's tendency and ability to spend more or less time on item completion. Because some items will show more homogeneous time distributions than others, the dispersion parameter α_i estimates an item's discriminatory power with respect to test-taker speed.

The parameters of interest are estimated in a Bayesian framework using a Markov Chain Monte Carlo method with a Gibbs sampler (van der Linden, 2006; Fox et al., 2019). We used Expected A Posteriori (EAP) estimators of test-taker speed τ_p as features for predictive modeling. While more complex models have been proposed which achieve a joint estimation of persons' latent speed and ability (and items' time intensity and difficulty, respectively; van der Linden 2007) and allow to include covariates (Klein Entink et al., 2008), we decided to stick with the simpler model, as incorporation of ability (van der Linden, 2006) showed no gains for predictions.[3]

4.2.2. Latent Test-Taker Ability

The provided log data did not include item scores. Thus, we had to re-engineer scores based on the log data and information from released items available through the NAEP Questions Tool

[3]Please note that the two polytomously scored items were collapsed to dichotomous scoring as the employed `LNIRT`-package only allows dichotomous scores at this point.

(National Center for Educational Statistics, 2020).

LOG DATA–BASED SCORE RECONSTRUCTION. To reconstruct the item scores, a detailed analysis of the log files was undertaken. Using unique item identifiers, we were able to map the competition data's items to example items provided in the NAEP Questions Tool, a public query tool used to showcase NAEP items. It was identified that 14 released items could be found in the log data, with another 5 items being unreleased in the Questions Tool. The mapping was verified and confirmed by text-to-speech contents in the log data. Thus, with the information from the public tool, we could derive key responses for those 14 items.

To reconstruct the scoring, it was necessary to identify when a test taker provided a response and to score it relative to the item information obtained from the Questions Tool. Given that the test allowed free navigation between items, navigation actions were used to define blocks of log data that related to each test taker's actions for a single item over an uninterrupted time span. This means that it was possible to have multiple blocks of log data for a single item, as a student may initially answer a question and then navigate back at a later time; for example, when reviewing their responses.

Within each block of data, the information was analyzed for actions relevant to that item type. Five different item types were identified by unique codes and the data structures that comprised a response. Each type of item required certain actions to indicate a response. For example, multiple-choice items where there are four possible answers contained observable actions such as *Click Choice* to indicate the test takers response, or *Eliminate Choice* where they had eliminated an option they thought was incorrect. For some items, test takers were required to type in their responses as free text. This meant that the log data typically showed multiple keystrokes being entered. In the log data, each keystroke was represented in JSON format. Responses to these items required reconstruction of the keystroke pattern to identify the final text input for reconstructing the response. At the end of each block, the response, or absence of a response, was recorded for the block of data.

From the response information for each block, it was possible to identify the last response that would form the final scored response. Given that users could navigate back and edit past responses, the scoring process identified all blocks associated with each item in the test. Where multiple visits had occurred, the last response was scored. Dichotomous scoring was used for seventeen and partial credit scoring for two items.[4] In some cases though, there was no response at all, but rather the student just viewed the item and navigated past. This was scored as *No Credit* in accordance with the NAEP technical report for scaling. Items that have never been viewed by a student were regarded as *Not Reached* and did not affect the later ability estimation.

Using the available 14 scored items, we estimated an intermediate ability score for test takers. By identifying the top 100 test takers across the 14 items, we then used their responses to correlate the remaining 5 unreleased items to identify the most likely correct answer based on the distribution of responses for the unreleased items. With this procedure, despite the limited information, we were able to infer the correct scoring for the data.

PARAMETER ESTIMATION. With this complete set of scores, we applied a Generalized Partial Credit Model (Muraki, 1992) for estimating person ability. Theoretically, such ability estimates

[4]The scoring of item *M3553E1/VH134366* was simplified by collapsing its three partial credit categories into one.

together with the speed estimates should be reasonably predictive of efficiency, as efficiency is defined by a trade-off between performance and effort (see Section 1). The model is represented by the following equation (Muraki, 1992):

$$\mathrm{P}_{jk}(\theta_p) = \frac{\exp\left[\sum_{\nu=1}^{k} a_j\left(\theta_p - b_{j\nu}\right)\right]}{\sum_{c=1}^{m_j}\exp\left[\sum_{\nu=1}^{c} a_j\left(\theta_p - b_{j\nu}\right)\right]} \tag{2}$$

The equation models the probability of a person p with the latent ability θ_p to respond to an item j by attaining the kth score level (e.g., partial credit). Here, m_j denotes the number of response categories of the item. In this model, subsequent score levels are ordered by their difficulty. The parameter b_{jk} represents the difficulty of reaching a certain score level and a_j constitutes the item discrimination (i.e., the degree to which the item is capable of distinguishing between more or less able test takers). The parameter b_{j1} has a special role as it cancels out of the equation and hence, its value can be arbitrarily defined and bears no meaning. If $m_j = 2$, that is, item j is dichotomous, the GPCM reduces to the 2PL IRT model. We used Marginal Maximum Likelihood for estimating model parameters. For person ability, Weighted Likelihood Estimators (Warm, 1989) were used. This way, test-taker ability θ_p can be directly used as a feature for predictive modeling.

4.2.3. Simple Indicators of Students' Work Process

The analysis of process indicators is based on the assumption that latent characteristics of a test taker can be inferred from attributes of their work process (Goldhammer and Zehner, 2017). However, the creation of indicators is often retrospective, depends on the specific assessment system employed, and is based on plausibility and expert opinion about which indicators might be of potential interest for a particular research question (e.g., time on task, number of page visits, or switching between environments). With the intent to provide a tool to facilitate the creation of process indicators from log data, the software package LogFSM (Kroehne, 2019) has been developed that can be used in R. Instead of providing a list of generic indicators, LogFSM requires the formulation of one or multiple theoretical models that a test developer or researcher has about the work process in a task. Afterwards, LogFSM reconstructs a given set of log data according to the predefined theoretical model(s). Attributes of the reconstructed work process then serve as process indicators.

The procedure of LogFSM utilizes the concept of finite state machines (Kroehne and Goldhammer, 2018). The work process is decomposed into a finite number of states which represent sections of the theoretically defined response process. For example, a researcher who wishes to distinguish process components in a math assignment might define the states *Task Reading*, *Task Processing*, *Responding*, and *Reviewing* that could alternatively be collapsed into states of lower granularity like *Stimulus Processing* and *Task Answering*. Practically, states are identified by events that represent test-taker interactions with the assessment platform (i.e., log events). The occurrence of such events can serve as the conditions that must be met in order to change from one state to another one, which is called transition. The interpretation of an event might differ from state to state, which may result in differences as to whether or not a transition is triggered. Depending on the previous state of a test taker, for example, a radio button click event might be interpreted as a first-time response (*Responding*) or an edited response (*Reviewing*). In summary, the interpretation of states and state sequences is constituted by the interplay of visible components of the assessment system (e.g., texts, images), the possibilities for interactions (e.g.,

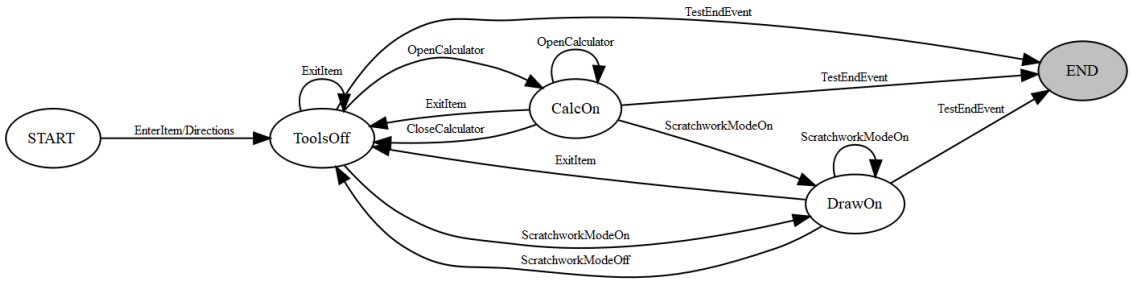


Figure 2: Exemplary Finite State Machine for Reconstructing Information from the NAEP Log Data

buttons, text fields), the contexts in which events take place (e.g., accessing a calculator before or after a response was given), and—most importantly—the predefined assumptions about test-taking behavior and cognitive operations (e.g., reading instructions, reconsidering an answer) (Kroehne and Goldhammer, 2018).

Finally, process indicators can be derived as attributes of the reconstructed states (or the reconstructed sequence of states) from log data that contextualize test-taking behavior according to the theoretically assumed test-taking process. The integration of the characteristics of a task, the available log events, and the theoretical expectations about the test-taking behavior assign a substantive meaning to an indicator (Kroehne and Goldhammer, 2018). For example, an indicator that reflects how long a student actually spends reviewing and checking a particular response again can be defined as the total time in a state *Reviewing* aggregated over multiple revisits of the task and cleaned for the time in other states such as *Responding*.

For the competition's data analysis, we specified five finite state machines to represent different attributes of students' work processes. The states of theses finite state machines represented students' on-screen page (26 states); attempting, processing, or reviewing of one of the 14 multiple-choice items (46 states) and items with other response formats (19 states); students' use of the text-to-speech tool (4 states); and their use of the calculator and the drawing tool (5 states). Figure 2 shows the last mentioned model as an example. We distinguished between having the calculator active (state *CalcOn*), having the drawing tool active (state *DrawOn*), and both tools being inactive (state *ToolsOff*). Transitions between states were triggered by the log events described in Section 2.1. For example, the state *CalcOn* was transferred to the state *ToolsOff* when the calculator was closed. That is, when the student pressed the calculator button (*CloseCalculator*), the drawing tool was activated (*ScratchworkModeOn*), or the item was left (*ExitItem*). Vice versa, when the drawing tool was activated, students' could not open the calculator, allowing for the modeling of distinct states. Self-transitions were specified to deal with double-clicks.

Several simple indicators were then derived as aggregated attributes of the reconstructed states or sequence of states. For example, the number of occurrences of the state *CalcOn* across items reflects how often a student opened the calculator during the assessment. A summary of the derived simple indicators and their descriptions is provided in Table 1. With only timing, count, and binary variables involved, no missing values were generated.

Table 1: Simple Indicators Serving as Features or for Derived Feature Modeling

Indicator	**Unit**	**Description**
Time on Screen	msec	Total time a student spent on each item within the test. This included the directions, review, and help screens.
Time on Directions	msec	Total amount of time spent on the directions screen.
Views w/o Attempts	count	Number of times a student viewed an item without interacting with the item's objects necessary for responding.
Items Attempted	count	Number of items at which a student showed behavior indicating they were attempting to complete the item.
Items Incomplete	count	Number of items which a student attempted, yet left the response area with incomplete information; e.g., only placing 3 out of 4 drag-and-drop boxes into the response area.
Items Completed	count	Number of items a student completed such that they interacted—as intended by the item—with all elements necessary for complete responding.
(Too) Fast	count	Number of times a student was in the fastest 5% of test respondents for a given item. This operationalization was chosen for being in line with the competition's definition.
Reviews	count	Number of times a student visited the review screen. This screen outlined which items they have worked on and which they have not worked on yet.
Timeout	binary	Indicator whether a student received the time-out screen, typically indicating that they failed to complete all items within the time limit of 30 min.
Text to Speech	count	Number of times a student used the text-to-speech feature.
Help	count	Number of times a student opened the help dialogue.
Calculator	count	Number of times a student opened the calculator.
Drawing Tool	count	Number of times a student opened the drawing tool.

4.2.4. Rapid Guessing

Compromised effort and persistence have been shown to be identifiable by investigating rapid guessing behavior (Wise, 2017). The concept of *rapid guessing behavior* is based on the assumption that the amount of time that a test taker spends on an item before responding is not sufficient to perceive the item and develop a serious solution (Schnipke and Scrams, 1997). A rapid guess can therefore be defined as a response to an item with a response time below a certain threshold.

For the definition of the thresholds, multiple approaches are possible (Wise, 2019). Following the competition's operationalization of inefficient test-taking behavior (Baker et al., 2019), the present work identified item-specific response time thresholds for rapid guesses based on a 5th percentile cut-off value. This implies the assumption that the fastest 5 percent of test takers on each item showed rapid guessing behavior. This was in line with the competition's definition of inefficient test-taking behavior and, thus, necessary for predicting the accordingly constructed target label. However, this is not state of the art, and the Discussion section reviews alternative approaches.

On the basis of the identified rapid guesses, a response matrix X_{pj} was constructed, indicating whether a response to item j by person p was observed and identified as a rapid guess. The entries in this matrix are specified as follows:

$$x_{pj} = \begin{cases} \text{NA} & \text{if no response is observed} \\ 1 & \text{if a response is observed and flagged as a rapid guess} \\ 0 & \text{if a response is observed and not flagged as a rapid guess} \end{cases} \tag{3}$$

X_{pj} was then used to extract several rapid guessing indicators. The indicators encompass a dichotomous grouping variable (whether a person showed at least one rapid guess), the sum of rapid guesses, and an estimation of a unidimensional latent rapid guessing propensity. The operationalization of latent rapid guessing propensity through dichotomous indicators is similar to previous studies, for example, those by Goldhammer et al. (2017) or Liu et al. (2019). However, contrary to Liu et al. (2019), the present study applied no joint estimation of rapid guessing propensity and ability. This way, the estimation of rapid guessing propensity does not account for the latent covariance of the two variables. For the estimation of the latent rapid guessing propensity, a Rasch model (Rasch, 1980) was selected:

$$\mathrm{P}(X_{pj} = 1) = \frac{\exp\left(\xi_p - \sigma_j\right)}{1 + \exp\left(\xi_p - \sigma_j\right)} \tag{4}$$

The Rasch model is similar to the GPCM presented in Section 4.2.2, just reduced to the dichotomous case and keeping the discrimination parameter constant. While the notation of symbols and indices is generally continued here (with the exception of changing θ to ξ for ability to distinguish and rapid guessing propensity in our notation), σ_j represents an item's propensity to evoke rapid guessing and X_{pj} denotes the observed rapid guessing behavior, with $x \in \{0, 1\}$. When a student did not respond to an item, their observed rapid guessing x_{pj} for this item was set to *missing*; that is, this item did not affect the student's estimation of rapid guessing propensity. The rapid guessing propensity parameter ξ, therefore, does not contain information on test-takers' tendency to (rapidly) omit or to not reach items towards the end of the test. For person parameter estimation, EAP estimates were used.

4.2.5. n-Grams of Log Events

The occurrence of certain log events can indicate behaviors or unobservable meta-cognitive, cognitive, or affective states of interest. This is also true for combinations of such. In the context of the competition, disengaged behavior might be a precursor or indicator for (later) inefficient test taking. For example, (a) a student's use of the drawing tool in an item that does not require its usage could be indicative of inefficient test taking as could be (b) the playing-around with the text-to-speech feature. For incorporating such predictive features, we adopted an approach by He and von Davier (2016) that borrows techniques from natural language processing and information retrieval.

At the core of the procedure (He and von Davier, 2016), a student's log events are considered as n-grams of a sequence. An n-gram is defined as a tuple of n subsequent log events within a student's complete sequence of log events. For computational as well as sample size reasons, it is common to limit analyses to uni-, bi-, and trigrams, which the present analysis did too. Hence, a sequence such as `ACAD` (representing four log events) would be decomposed into four unigrams ($2 \times \langle$`A`$\rangle$, $\langle$`C`$\rangle$, $\langle$`D`$\rangle$), three bigrams ($\langle$`AC`$\rangle$, $\langle$`CA`$\rangle$, $\langle$`AD`$\rangle$), and two trigrams ($\langle$`ACA`$\rangle$, $\langle$`CAD`$\rangle$). We decided to make each event item-specific; that is, the event `Draw` was captured together with the item ID, for example, `DrawTask4`. This way, events were contextualized. Varying by the 10, 20, and 30 min conditions, we obtained 7,448, 13,482, and 17,553 n-grams, excluding sequences that occurred in less than 15 students' log event streams to prevent overfitting.

Next, the frequency sf_{ij} of each n-gram i is computed for each student j (i.e., sequence frequency). These frequencies are then weighted by inverse sequence frequency (borrowing from the term *inverse document frequency*), $\text{ISF}_i = \log(N/\text{sf}_i)$, with N representing the total number of sequences, and log-normalized; that is $(1 + \log(\text{sf}_{ij})) * \text{ISF}_i$. This way, sequences occurring across many test administrations are scaled down in their importance and vice versa. Also, higher frequencies are dampened by the log-transformation.

The weighted n-gram frequencies can then be checked for their predictiveness of, for example, efficiency, using a χ^2-distributed statistic (details at He and von Davier 2016; Manning et al. 1999). This revealed 841, 1,259, and 1,190 significantly predictive n-grams ($\alpha = .05$) for the respective condition.

In the last step, we compressed the selected features in a principal component analysis. Due to the need for a low-dimensional feature space (see Section 4.3), we extracted only a few components, retaining only 5 percent of the original information. This resulted in 6, 9, and 14 components, respectively, for the three conditions.

4.3. Feature Selection

We applied several different feature selection strategies. First, we used random forests to obtain features' importance for predicting students' efficiency in Block B. Second, we evaluated the accuracy of predictions using different combinations of features. Both strategies showed speed to be the most predictive feature in all conditions. However, the importance of the other features differed depending on the data set and combination of features.

Moreover, we frequently observed that if the addition of a feature improved the classification performance on the training data substantially (evaluated by stratified, repeated ten-fold cross-validation), it reduced the performance on the test data significantly. Thus, low-dimensional models were always to be favored over high-dimensional ones. For our final classifier ensemble, the 10 and 20 min classifiers indeed turned out—with one exception—to work best with only

one single feature: latent person speed. In the 30 min condition, more features were selected for the final prediction. For a list of the resulting features for all conditions, see the following Section 4.4.2.

4.4. PREDICTION

4.4.1. Harvesting More Information: Multi-Label Classification

The binary target label split students into efficient and inefficient test takers. However, the competition's definition of *inefficient* behavior mixed two types of test takers: those who are going too fast and those who are going too slow. Since the two types have different feature realizations, the learning algorithms have to optimize towards at least two different conditions for the same class. Most algorithms' optimization works better if they have fewer conditions to optimize for within each class.

Therefore, we used the latent test-taker speed feature for further splitting the inefficient category into *Going too Slow* and *Going too Fast*. This new target label with now three instead of two classes was used for one set of classifiers (see Section 4.4.2). For doing so, the latent speed estimated by the Lognormal Response Time Model (see Section 4.2.1) distinguished between students going too fast and going too slow. An analysis showed that substantial rapid guessing behavior started at a threshold of about $\tau_{thr} = 0$ and, thus, optimally divided the two inefficient subgroups. Also, the number of timeouts decreased to a minimum at this point. The threshold was solely used to further distinguish the inefficient label and was only involved in one set of classifiers as described above, but not in any of the additional analyses or feature-engineering steps.

The resulting target label identified about 23 percent of the test takers as going too fast and about 17 percent as going too slow, keeping the original share of 60 percent of efficient test takers. On average, too-fast students showed about 2.2 rapid guesses, whereas too-slow students showed only about 0.4 rapid guesses, and efficient students showed about 0.5 rapid guesses. Similarly, 56 percent of too-slow students received a timeout, whereas only 33 percent of efficient and 11 percent of too-fast students received one. Note that the efficiency label–related variables refer to students' behavior in Block B, whereas the estimated latent speed and other reported variables refer to Block A.

4.4.2. Three Sets of Base Classifiers

For the final prediction, we created three sets of base classifiers that were to be merged in an ensemble. Each set followed a different idea, incorporated different features, and was trained by a different learning algorithm. In turn, each set contained three classifiers, with one of them tailored to the 10, 20, and 30 min conditions, respectively. We experimented with different feature sets, learning algorithms (common ones such as support vector machines, AdaBoost, J48, neural nets, and others), and hyperparameters for each base classifier. Table 2 shows which features and learning algorithms were used in which classifier. The issue of which features to include and which learning algorithm to employ was determined by our resulting performance with respect to the leaderboard. Due to an unstable performance in the test set, no systematic hyperparameter tuning was carried out.

Our first set of classifiers used support-vector machines with a radial kernel and C-classification for all three conditions (with $C = 1$, $\gamma = 1/n$, $\epsilon = 0.001$, shrinking). In the 10 min

Table 2: Three Sets of Base Classifiers

Condition	Learner	Speed	#Complete	#Incomplete	#TooFast	n-grams
Classifier Set (1): Speed & Test Completion						
10 min	Support Vector Machine	+				
20 min	Support Vector Machine	+	+			
30 min	Support Vector Machine	+	+	+	+	
Classifier Set (2): Multiclass Speed & Test Completion						
10 min	Multi-Class Support Vector Machine	+				
20 min	Multi-Class Support Vector Machine	+				
30 min	Multi-Class Support Vector Machine	+	+	+	+	
Classifier Set (3): Speed, Test Completion, & n-Grams						
10 min	JRip (Rule Learner)	+				+
20 min	Support Vector Machine	+				+
30 min	Support Vector Machine	+	+	+	+	+

condition, only speed was used for prediction. In the 20 min condition, the number of completed items was added. In the 30 min condition, all features that got through feature selection (except n-grams, on purpose) were incorporated: speed, number of completed items, number of incomplete items, and items completed too fast.

Our second set of classifiers was designed similarly to the first one, but with a multiclass support-vector machine and the multiclass label distinguishing going-too-slow and going-too-fast students (see Section 4.4.1). In the 10 and 20 min conditions, speed was the only predictor of importance according to the feature selection procedure. In the 30 min condition, again, all features (except n-grams) were incorporated.

Our third set of classifiers differed from the other two sets in that it incorporated one principal component of the n-grams of event sequences (see Section 4.2.5). Apart from that, the same set of features were used as in the second classifier set. The 10 min condition made use of a propositional rule learner instead of the otherwise employed support-vector machine. The rule learner's parameters were set to $F = 3$ folds, $N = 2$ as the minimal weight, maximum error rate of included rules $\geq .5$, and pruning was used.

4.4.3. Ensemble Learning

The three described sets of classifiers were combined in a final ensemble classifier. For this, we averaged probabilities of a condition's three base classifiers, but favored inefficient classifications. We chose to favor inefficient classification since our base classifiers produced not enough inefficient classifications. Therefore, we ended up with an ensemble of classifiers for the 10, 20, and 30 min conditions each.

4.5. Test-Taking Types in the NAEP Math Test

We used a generalized version of the finite state machine described in Section 4.2.3 to identify different test-taking types for the math test in NAEP. This finite state machine distinguished four different states during test completion.

4.5.1. Four States: Processing, Responding, Reviewing, Non-Task Processing

To distinguish different types of test-taking behavior, we compared test-taking behavior with respect to temporal changes of four different states. (I) A test taker is in the *Processing* state when beginning to work on an item. They stay in this state until making their first attempt to give a response to that item. As soon as a response is entered, their state changes from *Processing* to *Responding* or *Reviewing*. (II) The test taker is in the *Responding* state as soon as they enter their first response. They stay in *Responding* until they give their final response. If the test taker's first responding interaction equals their final response (i.e., they do not change their initial response and there is only one responding-relevant object to interact with), the *Responding* state is skipped. In such a case, the test-taker state changes directly from *Processing* to *Reviewing*. (III) The test taker is in the *Reviewing* state as soon as they have entered their final response. They then stay in the *Reviewing* state for the remaining time spent on the item (i.e., including revisits without responding-relevant actions). (IV) Since test takers in NAEP also receive generic instructions which are not directly related to certain items (e.g., introductory directions) and they can also navigate to overview pages while working on the items, such screen views were specified to lead to changing to a fourth state (*Non-Task Processing*), indicating that a test taker did not directly interact with and work on a certain item. Note that, with students being assigned to one state at all times, no missing values appear with respect to students' state at a certain point in time during their test.

4.5.2. Optimal Matching and Clustering

We used the sequences and duration of these test-taker states to perform an optimal matching algorithm that provides the dissimilarity between sequences (and therefore between test takers' behavior). To do so, we created one state per second for each test taker, resulting in a sequence for each test taker that was at maximum 30 minutes (1800 seconds) long. Test takers who took the test in less than 30 minutes received missing values (i.e., missing states) towards the end of their temporal state sequence. These sequences then went into the optimal matching algorithm that determines pairwise edit distances for transforming one sequence into another, which can serve as a measure of dissimilarity. We set the costs of adding or deleting one sequence element to $c(a^{Ins}) = c(a^{Del}) = 1$ and the costs for exchanging one element for another to $c(a^{Sub}) = 2$. Since not all sequences were equally long, we normalized the dissimilarity between the sequences by dividing them by the maximum length of the two sequences. This avoids larger dissimilarity values due to skewed sequence lengths. The result of optimal matching is a matrix that contains pairwise dissimilarities of all sequences.

Based on this dissimilarity matrix, we conducted a hierarchical cluster analysis in order to identify groups of test takers that showed similar behavior over the course of Block A, using Ward as the fusion algorithm (Ward, 1963). Finally, the normalized point-biserial correlation (PBC), average silhouette width (ASW), and Hubert's C index (HC) served as quality criteria for determining the optimal number of clusters.

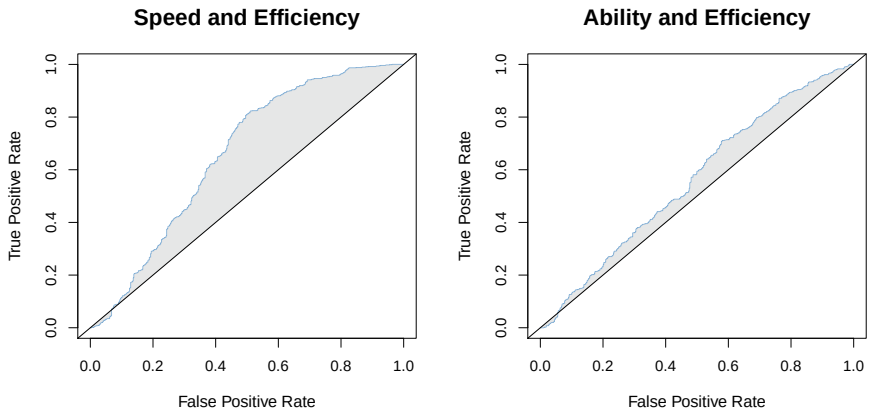


Figure 3: ROC Curves of Two Features: Speed (left) and Ability (right)

5. Results

5.1. Competition Results

The final evaluation of our prediction resulted in $\mathrm{AUC}_{adj} = 0.27$ and $\kappa_{adj} = .19$. In the leaderboard with all 82 competitors, this corresponded to rank 25, with several teams having submitted multiple results. In the final table, which only included 13 teams that submitted their code in time, our contribution was ranked eighth. The winner achieved $\mathrm{AUC}_{adj} = 0.34$ and $\kappa_{adj} = .22$. The rather low performance values, even for the winners, were accompanied by corresponding differences between the test and evaluation set, resulting in substantial changes in the ranking and indicating rather unstable models being prone to changes in the evaluation data. This is in line with the wavering performance during testing we observed.

5.2. Predictiveness of Single Features

With respect to single features, two of them were of particular interest: test-taker speed and ability. Figure 3 shows their ROC curves. Obviously, the latent speed feature taken alone predicts efficient test taking noticeably well ($\mathrm{AUC}_{adj} = 0.36$, $\kappa = .30$ in a single-feature support-vector machine[5]). In contrast, students' ability does not capture a lot of relevant information for predicting efficient test taking ($\mathrm{AUC}_{adj} = 0.16$, $\kappa = .07$ in a single-feature support-vector machine[5]). Please note that, after the competition, we corrected the scoring of items that students navigated to but did not attempt. Such cases were formerly considered missing observations (without impact on the ability estimate), but need to be regarded as incorrect responses for person parameter estimation in the given context. This change did not affect ability's overall predictiveness.

The large overlap of distributions between efficient and inefficient test takers for the ability feature further shows that the efficiency label does not contain much information about test takers' ability (right part of Figure 4). There is a small difference in that inefficient students have lower ability values on average ($\Delta = -0.22$, Cohen's $d = -0.19$). While the overlap of distributions appears somewhat similar for the speed feature (left part of Figure 4), the long right tail and prominence of faster inefficient test takers make the feature space more easily separable.

[5]based on the 30min training data and a stratified 10-times tenfold cross-validation

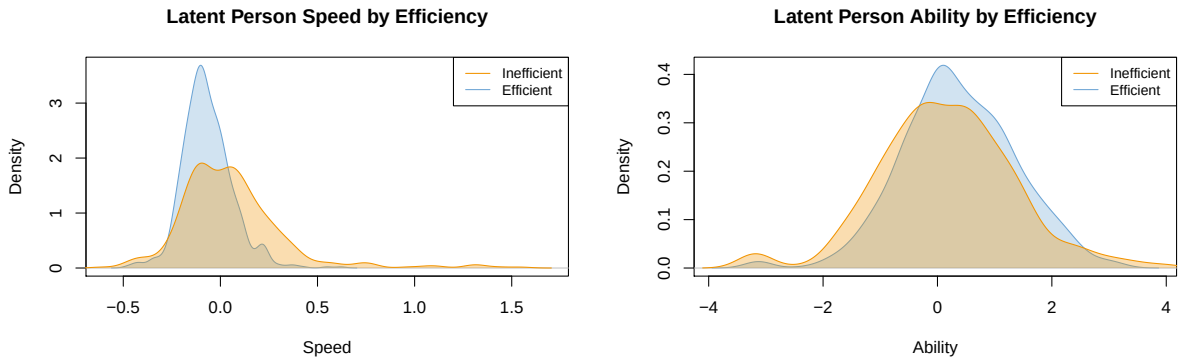


Figure 4: Distributions of Speed (left) and Ability (right), Separated by Efficiency

The effect size of the subgroups' difference is remarkably higher ($\Delta = 0.12$, Cohen's $d = 0.57$).

Finally, it has to be mentioned that a number of test takers who were relatively fast but answered correctly (and were thus estimated as relatively able) were classified as *in*efficient in Block B. It is possible that these students changed their behavior in the second test block. The other possibility is that the efficiency label classifies these instances erroneously as inefficient.

5.3. NAEP-Math Test-Taking Types

As a result of our cluster analysis (PBC: $r = .51$; HC: $C = .10$, ASW: $\tilde{s}(k) = .20$), we extracted four clusters of test takers, which are depicted in Figure 5. The plots' x-axes represent testing time (0–30 minutes), and the y-axes represent the relative distribution of the four states (*Processing*, *Responding*, *Reviewing*, *Non-Task Processing*; see Section 4.5). Students with shorter testing time than 30 minutes are not in any state towards the end of the testing time (represented as *missing*).

It becomes evident that all test takers, irrespective of their cluster assignment, spent most of their time in the *Processing* state. In Clusters #1 and #4, the majority of test takers did not finish their item completion within the 30 minute time limit, while test takers in Clusters #2 and #3 all finished their work within the given time limit. Regarding state sequences, Clusters #2 and #3 mainly differed with respect to the temporal aspect. Test takers in Cluster #2 showed the fastest item completion, while test takers in Cluster #3 needed slightly more time to finish the items. Clusters #1 and #4 differed mainly with respect to the distribution of states the test takers were in. In Cluster #4, test takers spent more time in *Responding* and in *Reviewing* states than test takers in Cluster #1, who showed mostly *Processing* behavior. However, in Cluster #4, test takers apparently invested relatively more time after entering their first response, which indicates that they reviewed and changed their responses more often.

For more profound insights into what distinguishes students from different clusters, Table 3 provides an overview of each clusters' characteristics. The first cluster—we call *Excessively Processing*—includes most students ($n = 653$). These students tend to be the slowest ($\bar{\tau}_{\#1} = -0.11$), with 58 percent receiving a timeout. While their mean ability estimates are second lowest and close to the average of the observed student groups ($\bar{\theta}_{\#1} = 0.21$, with an overall mean of $\bar{\theta} = 0.30$), they spend about 80 percent of their time before interacting with items in

Table 3: Outstanding Cluster Characteristics (Means)

Cluster	**#1** *Excess. Processing*	**#2** *Rushing*	**#3** *To the Point*	**#4** *Hesitant*	**Unit**
n	653	323	412	252	count
Efficiency (in B)	68%	35%	67%	63%	%
Items Completed	16.2	18.6	18.6	16.4	count
Total Time	29:18	18:12	24:30	29:30	min
Overall Speed[1]	-0.11	0.25	0.02	-0.07	logits
Timeout	58%	0%	0%	61%	%
Rapid Guessing[2]	-0.18	0.65	-0.20	-0.06	logits
Ability[3]	0.21	0.40	0.58	-0.06	logits
Processing	80%	73%	77%	63%	%
Responding	13%	16%	14%	24%	%
Reviewing	6%	8%	7%	11%	%
Non-Task Processing	1%	3%	2%	2%	%
Processing	23:27	13:27	18:44	18:45	min
Responding	3:46	2:53	3:29	6:58	min
Reviewing	1:47	1:25	1:46	3:11	min
Non-Task Processing	0:22	0:27	0:31	0:37	min

· All variables except *Efficiency* are based on log data from Block A.
[1] as of the Lognormal Response Time Model (cf. Section 4.2.1)
[2] students' rapid guessing EAPs (cf. Section 4.2.4)
[3] students' math ability WLEs (cf. Section 4.2.2)

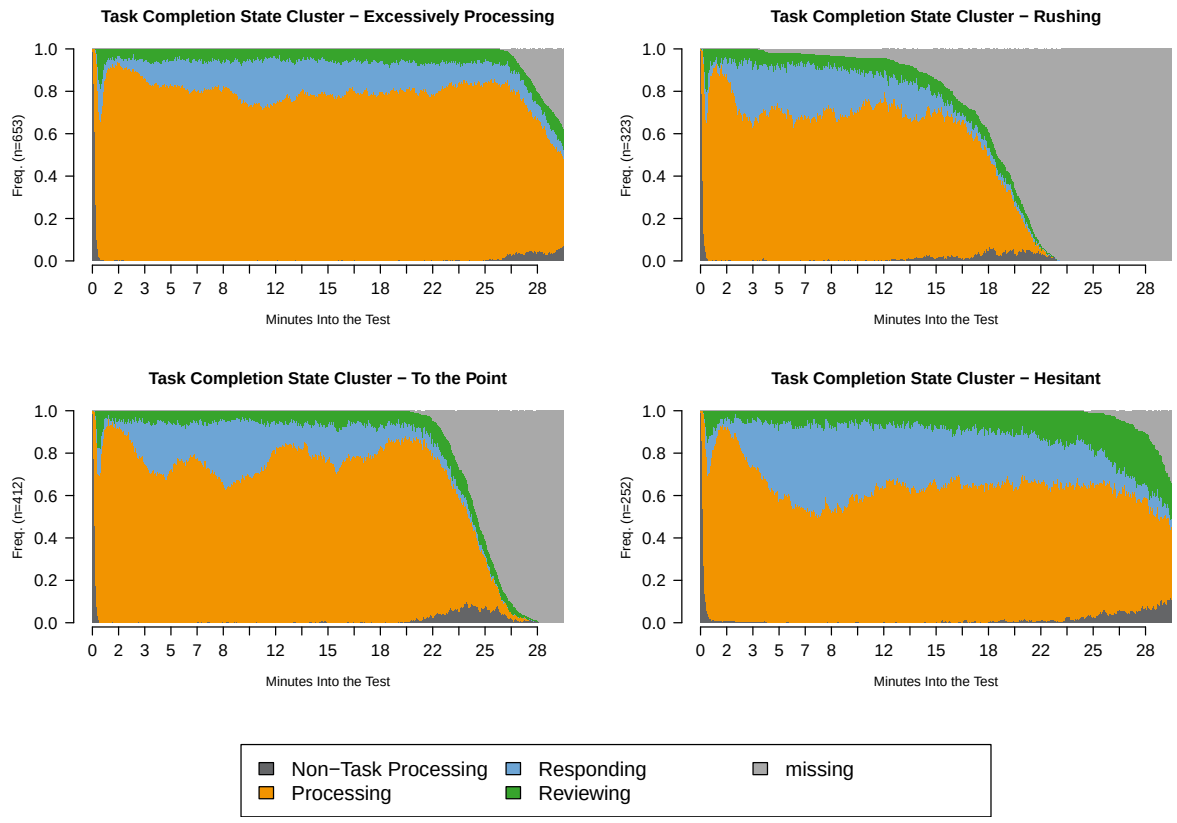


Figure 5: Sequence Distributions of Test Taker States Over the Course of Test Administration

the *Processing* state. That is, with the least relative shares on the other states, students in this cluster take their time until entering their response, and they less often tend to change their initial response.

Similarly slow, but with only $n = 252$ students assigned to it (of which 61% received a timeout), we call the fourth cluster *Hesitant* as they spend the relatively largest share of their time, compared to students in other clusters, in the *Responding* and *Reviewing* states. This means, while they are still spending a large amount of time in the *Processing* state, they interact longer with the item before entering their final response. They also remain somewhat longer in the item after responding before proceeding to the next one than students in other clusters. Although they tend to be relatively slower compared to the other students in the sample ($\bar{\tau}_{\#4} = -0.07$), they also show some, but not excessive rapid guessing behavior at single items ($\bar{\xi}_{\#4} = -0.06$). Most apparently, students in this Cluster #4 are on average relatively less able than students in other clusters ($\bar{\theta}_{\#4} = -0.06$).

In contrast to these two clusters with students showing relatively slow test taking, we call the second cluster *Rushing* ($n = 323$). Students in this cluster finish Block A early on with relatively high speed ($\bar{\tau}_{\#2} = 0.25$), not receiving any timeouts, and the highest tendency for rapid guessing ($\bar{\xi}_{\#2} = 0.65$). However, and importantly, these students still show the second-highest average ability level ($\bar{\theta}_{\#2} = 0.40$), meaning they are also providing a sufficient number of correct responses. Most remarkably, this second cluster is the only one significantly predicting inefficient test-taking behavior in Block B, with only 35 percent of efficient test takers in Block

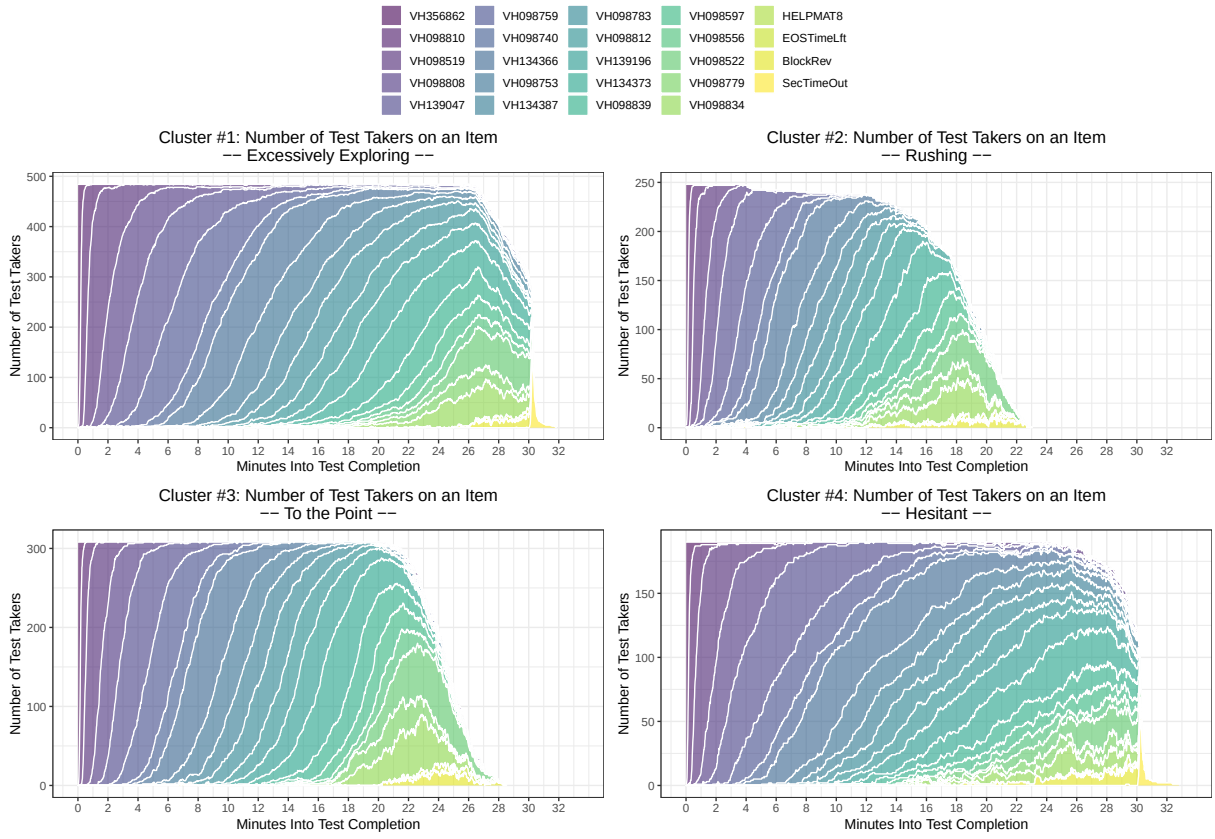


Figure 6: Distribution of Item Completions Over the Course of Test Administration

B. This seems to indicate that their rushed test-taking continues in the second test block.

Finally, students in the third cluster—we call *To the Point* ($n = 412$)—do not spend as much time in the test as students in Cluster #1 and #4, receiving no timeouts at all, showing the least rapid guessing behavior ($\bar{\xi}_{\#3} = -0.20$), and have the relatively highest ability on average ($\bar{\theta}_{\#3} = 0.58$). Compared to the other quicker students from Cluster #2, they spend more time Processing before switching to Responding.

Figure 6 depicts the relative share of students for each cluster and point in time during test administration on a certain item. While these plots naturally visualize the different paces of students in the different clusters, they also show how much smaller the *Rushing* Cluster's (#2) deviation is with regard to which item they worked on at a given point during test administration. Cluster #4's plot also shows how students in this cluster take relatively more time for the first items and then have to rush through the last items in face of the upcoming timeout. This is not the case for Cluster #1's students who still spend a similar amount of time on the block's last items compared to the quick students in Clusters #2 and #3.

6. Discussion

In this paper, we present a top-down approach for engineering features by means of psychometric modeling to predicting efficient test-taking behavior in the context of the NAEP Data Mining Competition for 2019. Furthermore, we present mining techniques that allow contextu-

alized in-depth analyses at the large scale for distinguishing test-taking behavior based on log data. The paper makes four important contributions: first, to the understanding of the provided competition data; second, presenting viable log data sequence mining techniques for educational assessment practice and research; third, to the setting of the competition; and fourth, to the community showing how to mine log data with a context-sensitive approach for identifying test-taking types.

One major contribution is that we demonstrate the value of domain knowledge–driven, top-down feature engineering that was informed by established psychometric models. Referring back to Merriam Webster's bipartite definition of efficiency as the characteristic of producing desired results without waste (Merriam-Webster, 2021), it is interesting how task success is not incorporated into the competition's conceptual specification of efficient test taking. The data patterns mirror the lack of the desired results in the competition's operationalization of the target label, demonstrating a prominence of speed as the sole determinant for the classification as efficient test taking. Remarkably, the outstanding speed feature serves as the only feature in some classifiers of our final ensemble that only falls short of the winning contribution by $\Delta\mathrm{AUC}_{adj} = .07$ and $\Delta\kappa = .03$. Empirically, ability did not provide any incremental increase in kappa or AUC beyond the speed feature. As a result, the ability feature was not included in any of the base classifiers after feature selection. It has to be noted that, at the theoretical level, the definition of efficiency only incorporates ability indirectly. That is the case because students who do not reach the end of the test cannot solve the corresponding items. Students who are going too fast are likely to fail as well. The resulting ability estimates, which are based on item success, hence, are indirectly incorporated in the efficiency label that is actually based on speed criteria exclusively. Nevertheless, this indirect impact was not large enough for granting substantial predictivity to students' ability for inferring their test-taking efficiency as specified by the competition.

It is apparent that the presented predictive modeling's performance does not exceed a moderate level, if at all. This is similarly true for the competition winners. While behavioral predictions with temporal delay can always be expected to be weak, there seem to be multiple additional reasons inherent to the provided data set and challenge behind the moderate predictive classification performance. From our point of view, there are three major points that are worth following-up on in discussions around the competition. The most prominent one is the data reduction to twenty and ten minutes of log data for two thirds of the test data. The resulting leaderboard and final scoreboard data evaluation were dominated by the secondary goal of predictions with less data. Also, since the different conditions shape the data and derived features quite differently, the training of classifiers had to be tailored to those.

The paper's second important contribution is that it provides evidence questioning the target label's validity. Using additional data sources from outside the competition, we were able to re-engineer scores for estimating test taker ability. Importantly, feature selection led to excluding the ability feature, as it failed to be predictive of the efficiency label. This was a strong indicator for suboptimal operationalization of efficiency.

This especially relates to the labeling of students as going too fast. To identify test takers spending a reasonable amount of time on an item, the competition organizers chose the 5th percentile of response times within an item as the threshold. Such a norm-oriented classification leads to labeling a fixed number of test takers as inefficient at each item, even when there are none or substantially less than 5 percent. Instead, criterion-based classification would be worthwhile. However, if corresponding criteria are not available, norm-oriented approaches would

need to be combined with a dynamic threshold to be determined for each item, as the response time distributions of items typically differ considerably. The high ratio of 40 percent of students labeled as inefficient, which seems unreasonably high, is probably the result of this purely norm-based decision.

One option for identifying an appropriate threshold constitutes the visual inspection of distributions if little information about items is available. Often, response time distributions are bimodal. The first, very early peak is then typically associated with rapid guessing, while the second peak corresponds to the actual response time mean of those test takers who did not exhibit rapid guessing behavior. The threshold would be set after the obvious extinction of the first peak (Wise and Kong, 2005). For setting an actually accurate threshold, methods that combine response time, item information, and response accuracy are considered state of the art. For an overview see for example (Wise, 2019). Further, the identification of thresholds should be guided by contextual considerations, judging for example whether false positives or false negatives are more acceptable in the context of the test.

An additional area of interest was that the binary label for efficiency mixes two types of students within its inefficient value: going too fast and going too slow. This has implications for the learning algorithms that have to optimize their parameters towards two different conditions within one class. Moreover, from a substantive perspective, this mixes at least two types of students: those who are disengaged—thus, either rushing or meandering pointlessly through the test—and those who are too thoroughly working, poorly monitoring their progress, or who are just less able.

As a final major contribution, our test-taker state sequence analysis demonstrates how events in log data can be contextualized for disambiguating their substantive meaning. While we do not claim to report a generalizable theory of test-taking with the subset of the NAEP mathematics test data, the adopted methods provide valuable information on test-takers for a particular test that could be used by assessment practitioners, researchers, and educational practitioners. Most importantly, the analysis shows that there is not one single test-taking behavior associated with high ability in the NAEP math test, whereas some are associated with more or less successful test-taking. Rather, the relatively lowest ability was associated with a test-taking behavior that we labeled as *Hesitant*. These test-takers generally took rather long for test completion, resulting in a high number of timeouts and a relative rushing through the last items in the block. They also took more time for responding and reviewing their responses. Most interestingly, for the NAEP test designers, this does not mean that there was an inherent problem with time management evoked by the test, but that students, especially in the largest Cluster #1, who were even slower but received a similar amount of timeouts, distributed their time more evenly across items and turned out to be in the normal range of ability on average. This set of findings demonstrates the importance of in-depth analyses that still scale up to large data sets and complex, semi-structured data (such as log data).

Moreover, with Cluster #2, which we refer to as *Rushing*, we identified students with a high probability of being classified as inefficient later in the test. Thus, on the one hand, the reported analysis is able to provide further information for predictive classification, and on the other hand, this shows that the going-too-fast behavior of these students seems temporally rather stable. This includes quite some rapid guessing, but with a sufficient number of correct responses. Cluster #3, called *To the Point*, shows how students allocate their testing time efficiently. These students were, on average, more successful than students assigned to other clusters. Even though they went relatively quickly through the test with not a single timeout limiting their test completion,

they exhibited higher ability. Compared to the *Rushing* students, they took quite some time for processing item information before switching to the responding state, in which they spend substantially less time than the *Hesitant* cluster, but a similar amount of time compared to the other clusters. Overall, it seems likely, but not testifiable with the present data, that the different test-taking behaviors also capture test engagement to some degree. It has to be noted that the differences in ability between clusters are only of a moderate magnitude.

7. Limitations

The paper already highlighted the presented study's limitations over the course of the different sections. Put briefly, this mainly concerns the target construct's operationalization, with ability not being accounted for and rushing through the test being identified with a normative threshold, as well as the split into three different time conditions and the efficiency label merging going-too-fast and going-too-slow students into one category. On top of the challenges inherent to the data competition, this study's main limitation constitutes the employment of baseline machine learning. Further, it possibly could have been of interest to take multiple measures of different response tendencies, aside from rapid guessing propensity, into account. Those tendencies could encompass for example (rapid) omitting or not-reached propensity (e.g., Sahin and Colvin 2020; Pohl et al. 2014). Especially as omitted responses, at least partially, seem to appear due to a lack of test-taking motivation (e.g., Jakwerth and Stancavage 2003; Wise and DeMars 2005). These tendencies therefore could be of interest when trying to get closer to efficient test-taking behavior in future research. The selection of feature sets and learning algorithms was optimized towards the test set which turned out to provide rather unstable evaluations. Interpretations of the clustering approach of test-taking state sequences are hampered by the fact of suboptimal clustering of data vectors in the hyperdimensional space.

8. Conclusion

Overall, the NAEP Data Mining Competition for 2019 provided an important opportunity to further develop conversations about how educational data mining and psychometric modeling can support data quality of assessments by identifying disengaged test-taking behavior.

One of the central messages of the competition is that predictions of test-taking efficiency are highly dependent on the definition, measurement, and evaluation of efficiency itself. That is true for the presented approach, as well as for other competition entrants, as could be seen through the scoreboard test set evaluation phase that signified varying performance of classifiers across data sets. In such a case, and if classifiers are meant to be put into productive usage, it is even more important from our point of view to have comprehensible models. Imagine a hypothetical situation when a teacher sees a student being flagged on a dashboard after 20 min of testing. The flag indicates the risk for inefficient test taking later on, but we know that the flag's accuracy is fairly low. It is vital that the teacher is informed about the basis of the flag's decision criteria. As we have shown, the competition's target label classified some of the most able students as inefficient who by ability are reasonably quick in completing the items. The consequences of a teacher going to a successful, engaged student and telling them they should aim at being more engaged, focused, or efficient in their test-taking, would be reasonably disruptive. Such an intervention could even be considered unethical, as it might change individuals' test outcomes for the worse. It can be assumed that such an invasive and intrusive test administrator behavior

would be counterproductive and decrease rather than improve data quality. However, if the included features for predictions are transparent, known, and understandable, the teacher could communicate those and contextualize the flag accordingly. A risk of more powerful black-box deep-learning classifiers is that a small to medium share of more accurately classified cases does not necessarily outweigh the resulting obscurity of classification mechanisms. More generally, the effects of the invasive disruption of a test administrator proactively trying to motivate test-takers on the standardization of the assessment setting need to be studied. Moreover, before using such a measure, classifiers would need to be checked for biases towards certain subgroups in order to still adhere to standards of standardized assessments (AERA/APA/NCME, 2014). Overall, we would recommend refraining from using such predictions with low to moderate accuracy in productive assessments as long as the effects of changes in the test administration are unknown.

Instead, the Discussion section gives some insights into possible improvements for the setup of a more proper training data set for predictions. Mainly, a representative definition of efficiency seems necessary, in particular one that reflects the current scientific state of the art which factors in students' ability. Furthermore, the described top-down feature-engineering with psychometric operationalizations, together with the referenced tools, can be helpful for mining log data from assessments at a large scale while retaining the individual perspective. With the illustrated software package LogFSM, for example, we were able to identify test-takers who clearly showed consistently inefficient behavior but were labeled as efficient, and vice versa. These observations are constrained by the fact that the log data of Block B was not available, yet served as the basis for the evaluation of the efficiency label. However, we think that the number of these cases is too large for being an effect of temporal instability only. We believe that these analyses combined with more innovative machine learning designs that the educational data mining community can provide are promising for further improvement of predictions of test-taking efficiency.

References

AERA/APA/NCME. 2014. *Standards for Educational and Psychological Testing*. American Educational Research Association, Washington, DC.

Baker, R., Woolf, B., Katz, I., Forsyth, C., and Ocumpaugh, J. 2019. Nation's Report Card Data Mining Competition 2019. https://sites.google.com/view/dataminingcompetition2019/home.

Baker, R., Woolf, B., Katz, I., Forsyth, C., and Ocumpaugh, J. 2020. Press release: 2019 NAEP Educational Data Mining Competition Results Announced. https://sites.google.com/view/dataminingcompetition2019/winners.

Bischl, B., Lang, M., Kotthoff, L., Schiffner, J., Richter, J., Studerus, E., Casalicchio, G., and Jones, Z. M. 2016. mlr: Machine learning in R. *Journal of Machine Learning Research 17,* 170, 1–5.

Cohen, J. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement 20,* 1, 37–46.

Fox, J.-P., Klotzke, K., and Entink, R. K. 2019. *LNIRT: LogNormal Response Time Item Response Theory Models*. R package version 0.4.0.

Gabadinho, A., Ritschard, G., Müller, N. S., and Studer, M. 2011. Analyzing and Visualizing State Sequences in R with TraMineR. *Journal of Statistical Software 40,* 4, 1–37.

GEIRHOS, R., JACOBSEN, J.-H., MICHAELIS, C., ZEMEL, R., BRENDEL, W., BETHGE, M., AND WICHMANN, F. A. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence 2,* 11 (Nov.), 665–673.

GOLDHAMMER, F., MARTENS, T., AND LÜDTKE, O. 2017. Conditioning factors of test-taking engagement in PIAAC: An exploratory IRT modeling approach considering person and item characteristics. *Large-Scale Assessments in Education 5,* 1, 1–25.

GOLDHAMMER, F. AND ZEHNER, F. 2017. What to make of and how to interpret process data. *Measurement: Interdisciplinary Research and Perspectives 15,* 3-4, 128–132.

GRAESSER, A. C. AND BLACK, J. B. 2017. *The Psychology of Questions*. Psychology Revivals. Routledge.

GRAESSER, A. C. AND FRANKLIN, S. P. 1990. QUEST: A cognitive model of question answering. *Discourse Processes 13,* 3, 279–303.

HE, Q. AND VON DAVIER, M. 2016. Analyzing process data from problem-solving items with n-grams: Insights from a computer-based large-scale assessment. In *Handbook of Research on Technology Tools for Real-World Skill Development*, Y. Rosen, S. Ferrara, and M. Mosharraf, Eds. IGI Global, Hershey, PA, 750–777.

JAKWERTH, P. M. AND STANCAVAGE, F. B. 2003. An Investigation of Why Students Do Not Respond to Questions. NAEP Validity Studies. Working Paper Series. Tech. Rep. NCES-WP-2003-12, National Center for Education Statistics, Washington, D.C. Apr.

KLEIN ENTINK, R. H., FOX, J.-P., AND VAN DER LINDEN, W. J. 2008. A multivariate multilevel approach to the modeling of accuracy and speed of test takers. *Psychometrika 74,* 1, 21–48.

KROEHNE, U. 2019. LogFSM: Analyzing log data from educational assessments using finite state machines. http://logfsm.com/index.html.

KROEHNE, U. AND GOLDHAMMER, F. 2018. How to conceptualize, represent, and analyze log data from technology-based assessments? A generic framework and an application to questionnaire items. *Behaviormetrika 45,* 2 (Aug.), 527–563.

LIU, Y., LI, Z., LIU, H., AND LUO, F. 2019. Modeling test-taking non-effort in MIRT models. *Frontiers in Psychology 10*, 145.

MANNING, C. D., MANNING, C. D., AND SCHÜTZE, H. 1999. *Foundations of statistical natural language processing*. MIT Press.

MERRIAM-WEBSTER. 2021. Efficiency. https://www.merriam-webster.com/dictionary/efficiency.

MURAKI, E. 1992. A generalized partial credit model: Application of an EM algorithm. *Applied Psychological Measurement 16,* 2, 159–176.

NATIONAL ASSESSMENT GOVERNING BOARD. 2017. *Mathematics Framework for the 2017 National Assessment of Educational Progress*. National Assessment Governing Board, Washington, DC.

NATIONAL CENTER FOR EDUCATIONAL STATISTICS. 2020. NAEP questions tool. https://nces.ed.gov/nationsreportcard/nqt/.

POHL, S., GRÄFE, L., AND ROSE, N. 2014. Dealing with omitted and not-reached items in competence tests: Evaluating approaches accounting for missing responses in item response theory models. *Educational and Psychological Measurement 74,* 3, 423–452.

R CORE TEAM. 2020. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria.

RASCH, G. 1960/1980. *Probabilistic models for some intelligence and attainment tests.* University of Chicago Press, Chicago, IL.

ROBITZSCH, A., KIEFER, T., AND WU, M. 2019. *TAM: Test analysis modules*. R package version 3.3-10.

SAHIN, F. AND COLVIN, K. F. 2020. Enhancing response time thresholds with response behaviors for detecting disengaged examinees. *Large-scale Assessments in Education 8*, 1–24.

SCHNIPKE, D. L. AND SCRAMS, D. J. 1997. Modeling item response times with a two-state mixture model: A new method of measuring speededness. *Journal of Educational Measurement 34,* 3, 213–232.

STUDER, M. 2013. WeightedCluster Library Manual: A practical guide to creating typologies of trajectories in the social sciences with R. *LIVES Working Papers 24*.

TOURANGEAU, R., RIPS, L. J., AND RASINSKI, K. A. 2009. *The Psychology of Survey Response*, 10. print ed. Cambridge University Press, Cambridge.

VAN DER LINDEN, W. J. 2006. A lognormal model for response times on test items. *Journal of Educational and Behavioral Statistics 31,* 2, 181–204.

VAN DER LINDEN, W. J. 2007. A hierarchical framework for modeling speed and accuracy on test items. *Psychometrika 72,* 3, 287–308.

WARD, J. H. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association 58,* 301, 236–244.

WARM, T. A. 1989. Weighted likelihood estimation of ability in item response theory. *Psychometrika 54,* 3, 427–450.

WISE, S. L. 2017. Rapid-guessing behavior: Its identification, interpretation, and implications. *Educational Measurement: Issues and Practice 36,* 4, 52–61.

WISE, S. L. 2019. An information-based approach to identifying rapid-guessing thresholds. *Applied Measurement in Education 32,* 4, 325–336.

WISE, S. L. AND DEMARS, C. E. 2005. Low examinee effort in low-stakes assessment: Problems and potential solutions. *Educational Assessment 10,* 1, 1–17.

WISE, S. L. AND KONG, X. 2005. Response time effort: A new measure of examinee motivation in computer-based tests. *Applied Measurement in Education 18,* 2, 163–183.

ZEHNER, F., HARRISON, S., EICHMANN, B., DERIBO, T., BENGS, D., ANDERSEN, N., AND HAHNEL, C. 2020. The NAEP Data Mining Competition: On the value of theory-driven psychometrics and machine learning for predictions based on log data. In *Proceedings of the Thirteenth International Conference on Educational Data Mining*, A. N. Rafferty, J. Whitehill, C. Romero, and V. Cavalli-Sforza, Eds. 302–312.