User Modeling and User-Adapted Interaction manuscript No.
(will be inserted by the editor)

The Details Matter: Methodological Nuances in the Evaluation of Student Models

Radek Pelánek

Received: date / Accepted: date

Abstract The core of student modeling research is about capturing the complex learning processes into an abstract mathematical model. The student modeling research, however, also involves important methodological aspects. Some of these aspects may seem like technical details not worth significant attention. However, the details matter. We discuss three important methodological issues in student modeling: the impact of data collection, the splitting of data into a training set and a test set, and the details concerning averaging in the computation of predictive accuracy metrics. We explicitly identify decisions involved in these steps, illustrate how these decisions can influence results of experiments, and discuss consequences for future research in student modeling.

Keywords Student modeling · Evaluation · Data · Metrics · Model comparison

1 Introduction

Student modeling is a crucial step in the development of personalized, adaptive learning systems. In recent years extensive research effort has been expended to the development of student models (Desmarais and Baker, 2012; Pelánek, 2017a). Both student modeling research and practical usage of student models involves their optimization (parameter fitting), comparison, evaluation, and interpretation. During these activities, we have to make many methodological decisions: What data do we use for fitting our models? How do we preprocess these data? How do we divide data into a training set and a test set? What performance metric do we use for comparing models? How exactly do we compute the value of a performance metric?

Faculty of Informatics, Masaryk University Brno; pelanek@fi.muni.cz, Phone +420-549 49 6991, Fax +420-549 491 820

These steps may seem like technical details that are not directly related to the core problem of student modeling. Consequently, they do not get much attention, and they are not documented (in sufficient detail) in research papers. This is understandable since some of these steps (like computation of the performance metric) can be performed using a single command in state-of-the-art programming environments. The main point of this paper is that this approach is dangerous. Even seemingly small methodological details can have significant consequences for the interpretation of experiments with student models.

As a specific illustration consider the recent work on deep knowledge tracing (Piech et al., 2015). In this work, the authors proposed a novel student modeling approach based on deep learning and claimed that it leads to a large improvement compared to previous results reported for the same dataset (citing Pardos and Heffernan (2011)). Further analysis, however, pointed out several problems with these results (Khajah et al., 2016; Wilson et al., 2016b; Xiong et al., 2016; Wilson et al., 2016a). One problem is related to the tagging of exercises in the used dataset. Multiple skills were handled by duplicating records in the dataset, and this (artificial) duplication had a significant impact on the performance of student models. Another problem was related to the computation of a performance metric. Although both Piech et al. (2015) and Pardos and Heffernan (2011) used the same performance metric (AUC), the metric was computed in each case in a slightly different way (global computation of the metric versus per-skill computation with averaging), and this inflated the differences between models. Overall, the large improvement reported by Piech et al. (2015) was to a large degree caused by methodological problems in the evaluation.

In this work we study in detail three methodological issues that are relevant to a wide range of student modeling research:

1. Data collection: How is the evaluation of student models influenced by the way data were collected?
2. Data splitting for cross-validation: How do we divide our data into a training set and a test set?
3. Averaging in metrics computation: How exactly do we compute the value of a predictive accuracy metric?

These issues get only marginal attention in the current research, but they can influence the interpretation of experimental results in significant ways.

To have a clear focus, we consider in our discussion only statistical models of student knowledge. The presented issues, however, are relevant also for other types of student models (e.g., models of affect) and other user models.

Our aim in this paper is to make the methodological decisions involved in the evaluation of student models more explicit and to highlight their importance. We also propose specific terminology and notation for their description. We hope this will help to attract more attention to these issues in future research and the communication about these issues will be easier.

Since our goal is to clearly present methodological issues, we use as our primary tool simple simulations that are designed to highlight the core of studied issues. The simulations are based on a very simple model of student learning (simple exponential learning curves)—this setting is sufficient for illustrating all major aspect of our discussion, and it leads to easily understandable and reproducible simulations. We also provide ample pointers to literature to illustrate what approaches are currently used in research papers and to show how the discussed issues are treated in other domains. This work extends previous work reported in Pelánek et al. (2016); Pelánek (2017b). These papers both used simulations, but each in a different setting. In the current submission, the setting is unified to provide a coherent presentation. The scope of covered methodological issues is also extended.

2 Background

Before we delve into specific methodological nuances of student model evaluation, we start by clarifying the context, terminology, and by describing the setting of simulation studies that we use to illustrate the discussed issues.

2.1 Basic Terminology and Setting

In our discussion, we consider only modeling of knowledge. Student modeling may also involve other aspects of student state (e.g., affect), but modeling of knowledge is the most common approach, particularly in practical applications, and it also has more standardized basic structure compared to other types of student models.

The fundamental entities involved in the modeling of student knowledge are denoted in research papers by several different notions. In this work, we use the terms *student* (a user of a learning system; also called a learner) and *item* (something that a student answers; also called a question, a problem, or an exercise). In our examples, we use only simple items, in which students provide answers that are evaluated as correct or incorrect. Real learning systems contain a wide variety of richer items (e.g., interactive multi-step exercises, items with hints and choices, items with partial correctness). The discussed issues are relevant to all common types of items. The more complex items typically lead to additional methodological nuances in the evaluation.

Individual items are grouped into *knowledge components* (also call skills or concepts); we use the terminology of knowledge components as used in the Knowledge-learning-instruction (KLI) framework (Koedinger et al., 2012a). A knowledge component (KC) is the basic unit of knowledge used for student modeling. For our discussion, it is mostly sufficient to view a knowledge component as a "set of related items". Unless explicitly specified, we assume that items within a knowledge component are homogeneous (interchangeable). From the practical point of view, this is an important simplification—we will discuss this point later.

2.2 Structure of Data

Student modeling can be based on rich data about student behavior; specific data depend on a particular application. For our discussion, we consider only the core data that are used for modeling student knowledge—we assume that data are given by tuples of the format *(student ID, item ID, answer, timestamp)*:

- a *student ID* is an anonymous identification of a student,
- an *item ID* is an identification of an item that the student answered,
- an *answer* is the summary of the interaction of the student with the item; for our discussion we consider only the correctness of answers,
- a *timestamp* identifies the moment of the answer (typically a standard database timestamp); from the perspective of student modeling the fundamental aspect is that it allows us to order answers of individual students.

We also assume that we have a mapping of items into knowledge components. For most of our discussion, we are not interested in specific items, i.e., we consider only into which knowledge component each answer belongs.

Typically much richer information is available, e.g., an answer may include not just the correctness, but also a specific value of an answer or a response time, a student ID may not be anonymous but linked to additional data about a student, for items we may have the full content and not just ID. But as it turns out, even the seemingly simple basic data lead to many methodological nuances.

2.3 Student Model

The primary goal of models of student knowledge is to take the data on the previous performance of a student and use it to provide an estimate of knowledge and predictions of future performance. Specifically, for a student s and an item i, a model predicts the probability the student s will answer the item i correctly. Examples of well-known student models are Bayesian knowledge tracing (BKT) or Performance factor analysis (PFA), see Pelánek (2017a) for a recent overview of student modeling techniques.

In our analysis, we want to focus on general methodological issues related to the evaluation of student models, and we do not want to get bogged down in details specific to a particular model. We also want to highlight and show in a clear way the discussed issues. This exposition would be difficult to do using real data and state-of-the-art models, in which these issues interact in complicated ways (as illustrated by the case of the deep knowledge tracing discussed in the introduction, which involved several research papers devoted to its clarification). Therefore, we utilize simulated data based on a very simple model of learning—simple error curve model, where the probability of an error by a student decreases exponentially with the number of attempts for a particular knowledge component. The model is a simplified version of a realistic model of practice (Heathcote et al., 2000). As a model of student learning

within a realistic learning system, it is, of course, very simplistic. Moreover, we utilize simulations where parameters of the model are set in an exaggerated manner to highlight the studied issues. Even though its simplicity, the model is sufficient to illustrate the discussed aspects of evaluation and has the advantage that it can be very easily specified and all described cases are thus easily reproducible. The model can be easily used for both generating data and as a predictive model. All reported experiments use 10,000 simulated students.

3 Data Collection

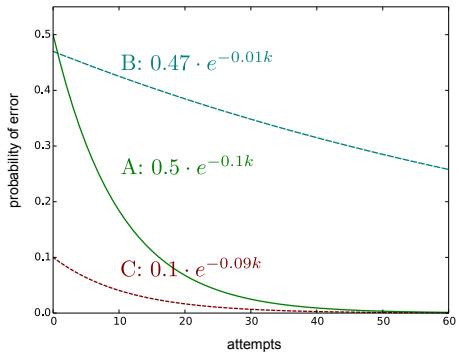
The evaluation of student models is typically done using historical datasets. Many datasets are publicly available, for example in the DataShop repository (Koedinger et al., 2010), but authors of these datasets typically do not provide details of data collection, e.g., under what circumstances the data were collected or what was the behavior of the tool used to collect data. Similarly, research papers mostly do not describe details of used datasets beyond their size. This is understandable since a detailed description of data collection involves technical details that are not in the core interest of researchers. However, we cannot afford the luxury of ignoring the details. If we do not take the data collection procedure and specific properties of datasets into account, we can easily reach misleading results.

We provide several specific examples of the potential impact of data collection. We begin with a simple, but very important aspect of data: the number of answers per student. Then we continue with illustrations of effects of attrition bias and impact of item ordering and selection.

3.1 Number of Answers per Student

We start our exploration of dataset properties by considering an elementary property—the number of answers per student and knowledge component. Some students (knowledge components) have many more answers than others, i.e., the number of answers has typically very skewed distribution. For now, we ignore this aspect and assume that the number of answers per students is constant. The skewness of the distribution, however, adds further complications to the analysis of data. We will analyze this issue later.

The number of answers per student can significantly influence results of a model comparison. Models may differ in their ability to model "initial phase of learning" and "plateau of performance" and the result of comparison thus may depend on what data we use. As a simple illustrative example, consider the setting depicted in Fig. 1. We generate the data according to the learning curve A and then evaluate models that provide predictions according to models B and C. If we use a small number of answers per student, model B achieves better performance. If we use a large number of answers per student, model C is better. This comparison is, of course, a very simplified case with naively



model	RMSE 10 answers	RMSE 50 answers
B	0.48	0.39
C	0.53	0.31

Fig. 1 Illustration of the impact of the number of answer per student.

wrong models. But a similar effect also occurs for a more realistic comparison involving commonly used models like BKT and PFA (Pelánek et al., 2016).

The number of answers can have a substantial impact on fitted parameters of models. For illustration, consider simulated students behaving according to the curve A in Fig. 1. When we fit the data using the BKT model, we get significantly different values for some parameters when using the first 10 answers ($P_{init} = 0.52, P_{learn} = 0.41, P_{slip} = 0.28, P_{guess} = 0.21$) versus using 50 answers ($P_{init} = 0.16, P_{learn} = 0.37, P_{slip} = 0.07, P_{guess} = 0.21$). It is tempting to interpret parameters of student models as features describing student learning. However, as this example shows, the fitted parameters can be more influenced by details of data collection than by properties of learning.

Liu and Koedinger (2017) provide a practical example of the importance of the number of answers per student for the analysis of real student data. In this study authors analyzed variants of popular student models (AFM and BKT) with individualized parameters. They discuss how the number of answers per student influences the validity and reliability of estimated parameters.

3.2 Attrition Bias

One commonly used approach to adaptation in educational systems is to let students solve varying number of items (problems, questions) of similar type and difficulty. A typical method is mastery learning—students solve items until they satisfy a master criterion (Pelánek and Řihák, 2017). The mastery decision can be made by some simple rule (e.g., "3 correct in a row") or with the use of a student model (e.g., "the probability of knowing the skill is at least 95%"). Consequently, in the collected data students differ in the number of items they answer and this difference is not random. This creates a *mastery attrition bias*[1], which can influence several aspects of model evaluation and interpretation.

[1] Attrition bias is a type of selection bias, which is often present for example in medical experiments.

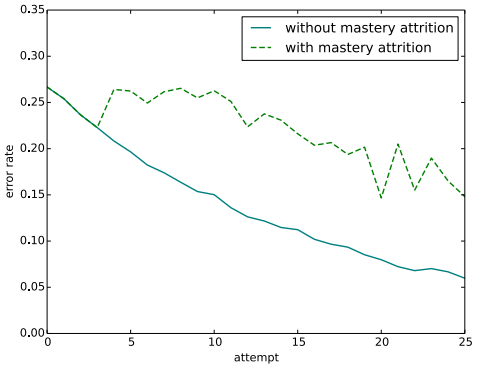


Fig. 2 Learning curves—impact of mastery attrition.

For a specific illustration of the impact of mastery attrition, let us consider the evaluation of learning systems using learning curves (Martin et al., 2011). A learning curve plots the error rate (or another student performance measure like response time) as a function of the number of attempts. A decreasing learning curve is evidence of learning; curves can be used to compare models or find ill-specified knowledge components. Learning curves are based on aggregating behavior across students, and this aggregation may complicate their interpretation (Martin et al., 2011). Particularly, learning curves based on data from adaptive systems are prone to the mastery attrition bias. This issue has been discussed in research (Käser et al., 2014b; Murray et al., 2013; Nixon et al., 2013), but is not taken into account sufficiently in the current practice. For example, recent work on mixture modeling using learning curves (Streeter, 2015) does assume a constant number of answers per student and would require significant modification to work correctly in the presence of mastery attrition.

For illustration, Fig. 2 shows a specific simple example of the mastery attrition bias using simulated data. We use simulated students with the probability of a wrong answer given by a learning curve $0.7e^{-0.06x}$. We simulate a heterogeneous student population of students by using different prior knowledge, which is emulated by assuming for each student "an initial number of attempts" (distributed uniformly from 0 to 40). When we assume that all students answer 25 items, we get a reasonable learning curve. To analyze the impact of mastery attrition, we simulate simple mastery criterion "4 correct in a row"—once a student answers 4 consecutive questions correctly, we do not consider his subsequent answers. For data with mastery attrition, we get much flatter and noisier curve, which underestimates student learning.

Similarly, it is possible to construct simulated scenarios, in which the absence or presence of mastery attrition determines, which student model achieves better predictive accuracy. Pelánek et al. (2016) presents a specific case in which when we compare the fit of PFA and BKT models (two standard

student models), we get that PFA has better performance if we use the constant number of answers per student, whereas BKT has better performance if data are collected using mastery learning.

In learning systems that are used by students voluntarily, there is self-selection bias, which can work in the opposite direction to mastery attrition. In many systems, the length of a study session is determined by students (rather than mastery learning or other system rules). In such cases, the number of answers depends on student motivation, which can be influenced by success. As a simple model scenario consider the following case of heterogeneous student population consisting of two subpopulations with flat learning curves. Students in the first subpopulation have constant success rate 50 %, students in the second subpopulation have constant success rate 80 %, i.e., students do not learn, they just differ in their prior knowledge. Let us assume that the number of answers is positively correlated with success rate, which is often the case in real systems (Lomas et al., 2013; Papoušek and Pelánek, 2015). If we plot the aggregated learning curve, the curve will show improvement in student performance—but this is just an illusory effect caused by student attrition. In some systems, it may even happen that both the mastery attrition bias and self-selection occur (Papoušek et al., 2016). This makes the interpretation of data highly challenging.

3.3 Item Ordering and Selection

It is convenient to assume that all items within knowledge component are sufficiently similar so that we can ignore their identity and treat them as interchangeable. We are using this assumption for most of this paper. The assumption is used (often implicitly) in many student modeling studies, for example in several ASSISTment papers that use models built using tabulating "success rate for the next problem" (Van Inwegen et al., 2015a,b). These models would not work in the case of an adaptive choice of items, where the learning system actively tries to achieve a given target success rate. The application of such models is thus limited to (implicitly assumed) properties of a particular dataset. We discuss several settings in which the policy used for selection and ordering of items (e.g., fixed, adaptive, or random) can have a substantial impact on evaluation and interpretation of student models.

At first, let us assume a fixed ordering of items, which is common when we analyze data coming from common non-personalized learning systems. As a specific illustration of the potential impact on the evaluation of student models consider the following scenario. We assume that student learning is described by learning curves in Fig. 1 and we consider items belonging to knowledge components A and C. We consider two datasets with the fixed ordering of items. In the dataset AC, first 16 items belong to the knowledge component A and the next 16 items belong to the knowledge component C. In the dataset CA, the order is reversed. Over these data we compare two models. Model M_{AC} is the optimal model that uses the correct learning curves and

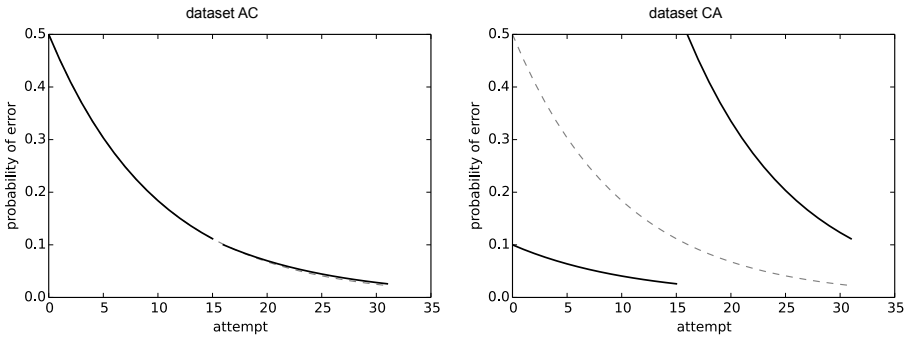


Fig. 3 Illustration of the impact of item ordering. The dataset AC contains 16 A items, followed by 16 C items. The dataset CA contains items in reverse order. The black lines show the true probabilites of correct answers, the gray line shows predicted probabilites according to model that assumes that all items belong to A.

the correct mapping of items to knowledge components. Model M_A is a model that assumes that all items belong to knowledge component A and uses the learning curve for A. Now consider the comparison on our datasets AC and CA (Fig. 3). When we use the dataset AC, the predictions of the two models are nearly indistinguishable. When we use the dataset CA, there are vast differences between predictions of these models.

This example is clearly an artificial, extreme case. Similar situation, however, appears often very naturally due to the differences in difficulty of items. In learning systems, it is quite natural that the order of items is related to their difficulty—students solve easier items first and then proceed to more difficult ones. If all students solve items in similar order, it may be impossible to disentangle increase in problem difficulty and student learning. Similarly to the above-given scenario, we can have two models that have widely different assumptions about item difficulty and speed of student learning and that have nearly indistinguishable predictions on data with fixed ordering, whereas if items are ordered randomly, predictions show significant differences between models (Pelánek et al. (2016) provide a specific example).

When data come from an adaptive system, the effects of item selection and ordering may be even more complicated. One of the goals of adaptive learning systems is to select items of suitable difficulty (neither too difficult nor too easy). Let us consider a hypothetical system that aims at providing students with items where they have 75% chance of answering correctly. Moreover, let us assume that the system uses an excellent student model that enables the system to achieve this target success rate very well. Using this system, we collect student answers and using this dataset we evaluate models. In the evaluation, we compare the predictions of our excellent student model with a simple baseline that provides a constant prediction of success 0.75. Due to the circumstances, we end up with nearly the same performance of the student model and the simple baseline. A naive conclusion would be that the student model is not very useful and we may as well use the naive baseline in our

implementation. But this is an entirely misleading conclusion since the good performance of the simple model is just due to characteristics of the collected data, which were obtained thanks to the powerful student model. If we have used randomly collected data, differences between the model and the baseline would be large. The described case is a simplified scenario, but similar effects can be observed in data coming from real systems, see Pelánek et al. (2016) for specific example in data from adaptive learning system for geography.

This example is a particular case of a feedback loop between student model and data collection—a model influences which data are collected, the collected data are used to evaluate the model. In Pelánek et al. (2016); Nižnan et al. (2015) we explored this feedback loop using simulations, showing that the use of an incorrect student model may lead to a dataset that is insufficient for demonstrating that the model is incorrect.

3.4 Related Work

Potential biases caused by data collection have been discussed in research in related domains. Research on the evaluation of recommender systems (Gunawardana and Shani, 2009; Herlocker et al., 2004; Shani and Gunawardana, 2011) discussed potential biases in data collection, e.g., by filtering users, and data splitting issues when using offline data. A proposal for layered evaluation of adaptive systems (Paramythis et al., 2010) includes the evaluation of data collection but does not address specifically its impact on subsequent steps. The presence of a feedback loop between data collection and model evaluation has been previously discussed in the context of "exploration vs. exploitation problem" (multiarmed bandits), with applications for news (Li et al., 2011; Wager et al., 2014) and advertisement (Bottou et al., 2013; Langford et al., 2008) selection.

In the context of student modeling, issues related to data collection have not been systematically studied before, although specific aspects have already been addressed by previous work, particularly in the context of learning curves (Martin et al., 2011) and mastery learning. The impact of mastery attrition on evaluation using learning curves has been discussed in several research papers (Fancsali et al., 2013; Käser et al., 2014b; Murray et al., 2013; Nixon et al., 2013). Specifically, previous work showed specific illustrations of confounded learning curves (Nixon et al., 2013), discussed methods for disaggregation of learning curves (Murray et al., 2013), and proposed mastery-aligned models (Käser et al., 2014b) to take this bias into account. Confounding effect of item ordering on learning and item difficulty has been mentioned in several works (González-Brenes et al., 2014; Jarušek et al., 2013; Khajah et al., 2014; Pelánek and Jarušek, 2015), but only as a side note. Doroudi and Brunskill (2017) discuss an example showing the impact of the number of answers per students on fitted model parameters.

4 Data Splitting for Cross-validation

When we build our student models, we do not want just to fit available historical data, but we want our models to generalize, i.e., to provide useful predictions for novel students and items. To evaluate the ability of models to generalize, we typically use cross-validation: data are repeatedly divided into training set and a test set, the training set is used to fit model parameters, the test set is used to estimate model performance. Cross-validation is a general machine learning approach, and most aspects of its usage of in student modeling are just special cases of general machine learning methods. Most of these aspects are not specific to student modeling, although some of them deserve more attention by student modeling community; specifically, the issue of data overlap in cross-validation and the risk of type I error (Dietterich, 1998).

Here we focus on one step in the cross-validation method: data splitting. This step is specific to the nature of data and thus deserves specific attention in the context of a particular application domain. For cross-validation, we need to split the data into a training set and a test set and to decide which data exactly are used to perform predictions. This decision needs to take into account the structure of data. In domains where data points are independent (e.g., classification of images), data splitting can be performed by a simple random selection. In student modeling, the data have many dependencies: a single data point corresponds to a student answering an item at a particular time. Such a data point is related to other data points via the student, the item and also via time. When performing data splitting, we need to take at least some of these relations into account. However, it is not clear which relations should be taken into account and how to take them into account. Many specific choices can be made and these choices influence results of an evaluation. The cross-evaluation choices are also often under-documented in research papers, which hampers reproducibility of research.

4.1 Splitting with Respect to Students and Items

At first, let us ignore the temporal aspect and focus only on the structure of data with respect to students and items. With random data splitting, we do not take students and items into account while splitting data. This approach does not correspond to real usage of student models—in practice, we have some historical data on past students and items, we use them to fit and compare our models, and then we use models to make predictions for new students and items. In the related field of recommender systems, some researchers use the terminology of "weak and strong generalization" (Marlin, 2004; Volkovs and Yu, 2015)—weak generalization tests predictions for users that are included in the training set, strong generalization tests predictions for new users.

In student modeling literature researchers sometimes use the terminology "student-stratified" and "item-stratified" cross-validation. This terminology is, however, potentially confusing. In general machine learning, the term "strati-

fied cross-validation" is typically used with the meaning that division into folds is done in such a way that "class distribution in each fold is approximately the same as in the initial dataset" (Diamantidis et al., 2000). Taking this meaning of stratification, the notion of "student-stratified cross-validation" would mean that "folds are balanced with respect to students". In fact, some researchers have used the term in a closely related meaning, for example Sao Pedro et al. (2013b). More often, however, the meaning of student-stratified is "all student data are either exclusively in a training set or exclusively in a testing set". This meaning was used for example by Yudelson et al. (2013); Koedinger et al. (2016); Niznan et al. (2015).

We propose the use of terminology based on the "level" keyword, as illustrated in Fig. 4. This terminology has less overloaded meaning than the terminology using the keyword "stratified" and the "student-level cross-validation" notion has already been used in many papers, for example by Baker (2010); Koedinger et al. (2012b); Pardos et al. (2013); Baker et al. (2012).

The most common type of data splitting in student modeling is "student level", i.e., testing generalization to new students. Such splitting is more common than "item level" since there is an asymmetry between students and items—items are usually rather fixed, whereas new students arrive continuously. Thus, we mainly want to be able to evaluate generalizations across students. In some applications of student models, however, it is useful to focus primarily on item level validation (Koedinger et al., 2012b).

We may also be interested in other kinds of generalization, for example, in generalization to new knowledge components, or in generalization to new student populations. Natural student populations that can be used for validation are students enrolled in a course within a specific semester ("cohorts"); this approach was used for example by (Ren et al., 2017).

4.2 Temporal Aspects of Data and Dynamic Predictions

Another important aspect of data splitting is the temporal nature of student data. Students answer items in some order, and we need to take this order into account. With the random splitting of data, we would end up using future actions for predicting past actions, which does not make sense. Taking the timing into account in a strict manner would mean picking a specific time and splitting data into a training set and a test set using this time. This approach would correspond to a single test set (holdout set).

If we want to perform cross-validation, we need to do multiple different data splits. A common approach is to perform student level cross-validation, i.e., to respect the timing ordering only for individual students and to ignore it across students. This approach assumes that the properties of both items and the student population are stable across time. This assumption is probably reasonably satisfied in most educational applications. As a specific example where the assumption is not fully satisfied we can consider learning of geography facts—the student knowledge of some places may be significantly

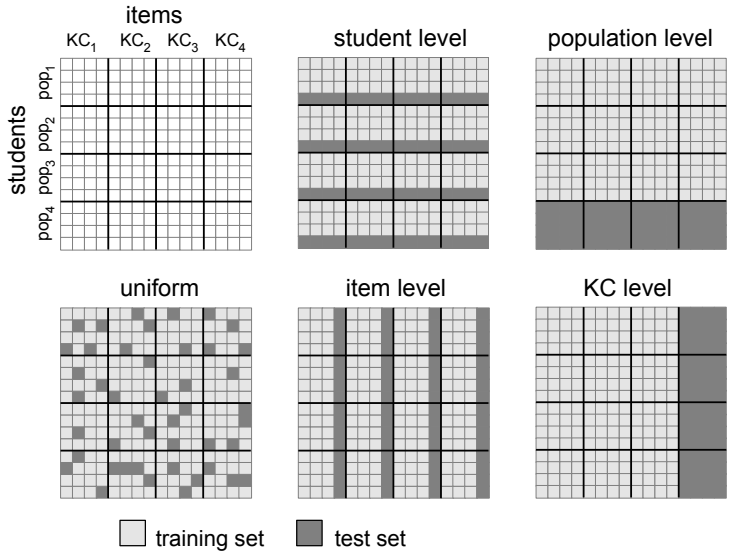


Fig. 4 Data splitting with respect to students and items. Rows correspond to students, columns correspon to items. The first diagram shows division of students into populations and items into knowledge components. The remaining diagrams outline several basic data splitting procedures.

influenced by current events (e.g., war, epidemics) and thus be unstable across time. The impact of such issues on model comparison should be negligible. Nevertheless, the used assumption deserves some supporting analysis or at least an acknowledgment that it is being used.

An important aspect in which research studies differ is whether predictions for a sequence of student actions are static or dynamic. Fig. 5 illustrates this distinction combined with presence and absence of student-level cross-validation.

- *Static predictions:* Predictions for all testing data of a given student are made at the same time, i.e., predictions are based only on data included in the training set.
- *Dynamic predictions:* Predictions are continuously updated after observing each answer, i.e., predictions also incorporate data in the test set (respecting their ordering).

The dynamic approach is the standard approach used in time series forecasting (Hyndman and Athanasopoulos, 2014; Bergmeir and Benítez, 2012).

The student level cross-validation with dynamic predictions typically most closely corresponds to the actual application of a student model in a learning system. In most cases, it should be the preferable approach to cross-validation. The reason for the use of static predictions is in many cases convenience—for some modeling approaches (e.g., those that utilize Monte Carlo Markov Chains or matrix factorization) it is not easy to recompute the predictions after each

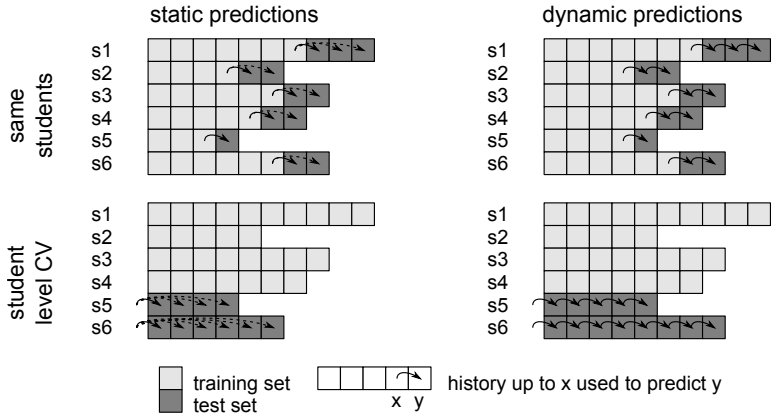


Fig. 5 An illustration of the basic options for cross-validation methodology. Dashed line denote cases where the static predictions differ from dynamic predictions (by not considering the recent data). Similar way to illustrate division into a training set and a test set was previously used by Khajah et al. (2014); Reddy et al. (2016); Pelánek (2017a).

observation. This is, however, a potentially significant limitation for practical applications and it needs to be explicitly discussed.

4.3 Overview of Approaches Used in Current Research

To illustrate the scope of different approaches to cross-validation, we provide examples of specific methods used in previous work. Note that in some cases the exact approach is not explicitly specified and is deduced from the context.

Student-level cross-validation with dynamic predictions was used, for example, by Streeter (2015); Niznan et al. (2015); Pardos and Heffernan (2011); Yudelson et al. (2013); Yudelson and Koedinger (2013); this approach is also typically employed in other research that uses the BKT model. Student-level cross-validation with static predictions was used, for example, by Käser et al. (2014b); Klingler et al. (2015); it is also used in other works using the AFM model. Liu and Koedinger (2017) use student-level cross-validation with both dynamic predictions (for the BKT model) and static predictions (for the AFM model); they note that the results cannot be directly compared. Koedinger et al. (2012b); Liu et al. (2014); Koedinger et al. (2016) consider both student-level and item-level cross-validation and argue that for their purposes the item-level cross-validation is more relevant. Static predictions on the same students (using the last 20% of attempts) were used by Khajah et al. (2014); Pelánek and Jarušek (2015).

There are also other approaches, which go beyond those illustrated in Fig. 4 and Fig. 5. González-Brenes et al. (2014) use offline evaluation on new students, but only the second half of sequence is used for evaluation. Some researchers use only a single attempt per student for the evaluation of predictions (Pardos and Heffernan, 2010; Reddy et al., 2016).

4.4 Interaction with Data Collection

The choice of a data splitting approach needs to take into account the data collection procedure. Particularly, when data are collected using mastery learning, the data splitting based on "same students" will lead to a bias towards correct answers in a test set—whatever mastery learning criterion is used, it is based on seeing correct answers.

As a specific example, consider the approach to building the test set as the last 20% of attempts of some students—this approach has been used for example by Khajah et al. (2014). Such a test set can bias results of a model comparison. As a simplified illustrative situation, consider students who answer entirely randomly, i.e., with a constant probability of success 0.5. Let us compare two simple student models: model A predicting the probability of success 0.5, model B predicting the probability of success 0.7. Model A corresponds exactly to the assumptions of our simulated setting, so it is by construction the better model. What happens when we perform the evaluation on the last attempts of data collected by simple mastery learning condition "3 correct in a row"? Model B achieves better performance as correct answer dominate in this test set, even though their occurrence is just due to the data collection condition, not due to some inherent aspect of student behavior.

5 Metrics for Predictive Accuracy

Once we have determined a cross-validation approach, we can train our models on a training set and let them provide predictions on a test set. To evaluate a model we need to quantify its predictive accuracy over the test set. To this end, we need to summarise the difference between predictions and observed values by a single number—a performance metric. The choice of this metric, details of its computation, and interpretation of the obtained values also involve several methodological nuances.

We discuss commonly used metrics, their interpretation, and usage in the current research. We focus specifically on the insufficiently documented issue of metric averaging.

5.1 Definition of Metrics

Many performance metrics can be used for measuring predictive accuracy of student models; see Pelánek (2015) for an extensive overview. In this work, we focus on two metrics that are most commonly used for evaluation of models of student knowledge, RMSE and AUC, and we analyze methodological details of their usage. Other student metrics are for example log-likelihood (which in practice behaves similarly to RMSE), mean absolute error (which is unsuitable for evaluation of binary predictions), and metrics based on the quantitative understanding of errors like accuracy or F1 measure (which are more suitable for models of affect rather than knowledge).

We start with a basic definition of metrics. We assume that we have data about n answers, numbered $i \in \{1, \ldots, n\}$, a student model provides predictions $p_i \in [0, 1]$, and the observed value is given by the binary value $o_i \in \{0, 1\}$. Root mean square error (RMSE) is then given as $RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (o_i - p_i)^2}$. RMSE is an "error metric", i.e., lower values mean better predictive performance.

The second metric that we consider is the area under the receiver operating curve (AUC). The receiver operating curve (ROC) summarizes the performance of a binary classification over all possible thresholds. The curve has "false positive rate" on the x-axis and "true positive rate" on the y-axis, each point of the curve corresponds to a choice of a threshold; for a detailed introduction to ROC curve construction and interpretation see Fawcett (2006). The area under the ROC curve (AUC) provides a summary performance measure across all possible thresholds. It is equal to the probability that a randomly selected positive observation has higher predicted score than a randomly selected negative observation. AUC is 1 for a perfect model and 0.5 for random predictions, i.e., it is interpreted as a reward (higher is better). The area under the curve can be approximated using a metric called A'; this metric is equivalent to the well-studied Wilcoxon statistics (Fogarty et al., 2005).

5.2 Choice and Interpretation of Metric

The choice of a metric used for comparing student models is important since different metrics lead to different results; this has been documented previously using data from real systems (Pelánek, 2015). Here we use our simulated scenarios with learning curves to provide a clarification related to the interpretation of absolute and relative values of metrics (by the relative value we mean the difference in values for two models). Sometimes these values are used to make judgments about the quality of a model or the significance of model improvement. Such use of metrics is, however, rather misleading. From the perspective of model evaluation, it makes sense to consider only the ordering of metric values; the magnitude and differences of metrics values are dependent mainly on the data available, not on the quality of models.

For the RMSE metric, the value of the metric is closely related to the average error rate. When the average error rate is near 50%, the RMSE value will be near 0.5, unless we have an excellent predictor, which is in the case of predicting noisy student behavior unlikely. If the average error rate is low, the RMSE value will go towards zero even for a simple constant predictor. For the AUC metric, the value will be high (near 1) even for a simple model when there is high heterogeneity in data (e.g., differences among knowledge components, students, or pronounced learning leading to a large difference between the beginning and the end of each student's sequence). With homogeneous data, the AUC value will be typically low (near 0.5) even for a complex model.

As a specific example, consider the error curves in Fig. 6. We consider only cases where we fit the data by the same model that generated them. If

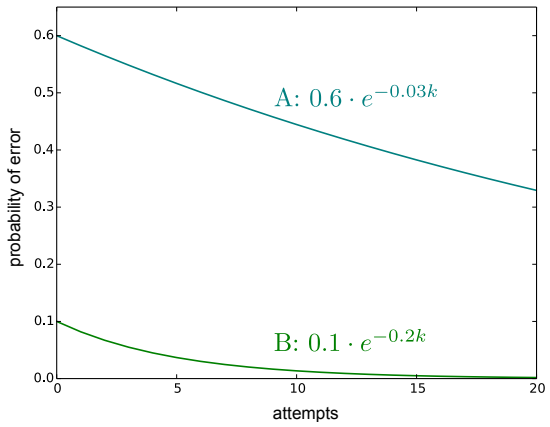


Fig. 6 Error curves used for illustration of metric values.

we consider only the curve A, the metric values are poor: RMSE 0.492, AUC 0.593. If we consider only the curve B, RMSE is much better: 0.160 (because the error rate is low). If we consider a model with both curves A and B, AUC is much better: 0.834 (because of the heterogeneity in data). Note that in all cases we are evaluating optimal predictions, i.e., the differences in metric values are not caused by fundamental changes in the predictive ability of models, but by the characteristics of data.

We typically use metrics for model comparison, and thus the focus is not on their absolute values, but on the relative performance of models (a difference between metric values for different models). These relative values also cannot be easily interpreted. Specifically, with the AUC metric, we can have vastly different models and obtain the same or nearly the same values of the metric. For example, if we divide all predictions by two, the value of the AUC metric remains the same, since the metric considers only relative ordering of predictions. In the error curve model with a single knowledge component, even an arbitrary model for which error predictions decrease with k achieves the same AUC value as the optimal model. A substantial difference in AUC typically means a model improvement, but a lack of difference in AUC clearly does not mean "absence of improvement" (see also discussion by (Marzban, 2004)). This means that by relying solely on the AUC metric, researchers can miss important results!

5.3 Averaging Issues

The basic definitions of metrics treat data as one dimensional. But in student modeling, we typically have two natural ways to group data: with respect to students and to knowledge components. Data (both observations and predictions) can thus be seen as a matrix, typically with missing values. The basic definitions of metrics are based on computations over a flattened matrix. Al-

ternatively, we can compute metrics per row (or column) of the matrix and then compute an average value of the metric.

Thus there are three main approaches to computing any metric:

- *Global computation.* In the metric computation, we do not differentiate between students and knowledge components and treat all data points as equal. This is the most straightforward approach (and also probably the most common), and thus this should be a default meaning of metric (when the computation is not further specified). If we want to make the computation explicit, we denote this approach to the computation of the metric using the subscript g, e.g., $RMSE_g$, AUC_g.
- *Averaging across knowledge components.* We compute the metric for each knowledge component and then take an average (in the case of a low number of knowledge components we may also report the value for each of them). We denote this approach to the computation of the metric using the subscript $\overline{KC}$, e.g., $RMSE_{\overline{KC}}$, $AUC_{\overline{KC}}$.
- *Averaging across student.* We compute the metric for each student and then take an average. We denote this approach to the computation of the metric using the subscript $\overline{s}$, e.g., $RMSE_{\overline{s}}$, $AUC_{\overline{s}}$.

None of these approaches is "the correct one" since the suitability of each approach depends on a particular application. At the same time, the choice of the approach is important—we will show that these approaches can lead to quite different results.

To get a basic intuition why the results may differ, consider a case of highly uneven distribution of answers, i.e., some students (knowledge components) have a much larger number of answers than others—such a situation is in fact very typical in real learning systems. With the global computation of a metric all data points have the same weight, and thus the results are influenced mainly by students (knowledge components) with many answers. On the other hand, the per student (or per KC) computation gives equal weight to all students (KC) without regard to the number of answers, i.e., answers for students (KC) with many answers have less weight.

5.3.1 Averaging Across Students

Now we turn to the discussion of issues related to different averaging methods, starting with averaging across students. The differences in the global computation of metrics and averaging across students are important in cases where the number of available data from individual students is unevenly distributed. This corresponds to a typical case in any user data—typically we have many users with few responses and few users with many responses. The global computation of metrics gives the same weight to all responses, whereas averaging across users gives the same weight to all users (and thus lower weight to responses by users with many responses).

For an illustration, we consider a specific simulated scenario. We again use learning curves from Fig. 1 (in Section 3.1). The data are generated according

Table 1 Illustration of the per student computation using error curves from Fig. 1. Data are generated according to curve A, with 70% of students having 5 attempts and 30% of students having 60 attempts. The table compares predictions by curves B and C.

model	$RMSE_g$	$RMSE_{\overline{s}}$
B	0.40	0.46
C	0.35	0.48

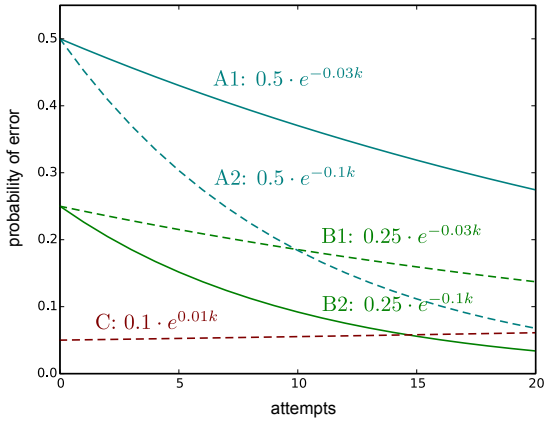
to the curve A with an uneven distribution of the number of answers among students: 70% of students have only 5 attempts, 30% of students have 60 attempts. Data are then fitted with two models. The first one (curve B) fits only the beginning of a sequence; the second one (curve C) fits only the end of the sequence. Table 1 gives the comparison of RMSE values for these models. If we compare the models with respect to the RMSE metric computed globally, model C is better. If we compare the models with respect to the RMSE metric averaged across students, model B is better.

The use of the AUC metric with averaging across students brings one additional problem. The AUC metric is not well defined when all responses are the same (e.g., all answers are correct). When we consider the computation of the metric per student, such cases are likely to happen, particularly for students with a small number of answers. It is not clear how to treat these cases. The basic approach is to ignore these undefined cases—this was done for example by Sao Pedro et al. (2013a). However, this approach is not entirely fair—we want predictors to behave well even for these students, and thus we should take these predictions somehow into account.

5.3.2 Averaging Across Knowledge Components

The essential difference between global computation and averaging across knowledge components is the same as in the case of averaging across students—each method distributes weights differently to the available data points. The issue can be again quite pronounced for practical systems, as often the distribution of responses among KCs is very highly uneven—in the case of KCs even more than in the case of students since popular basic KCs often have orders of magnitude more responses than very specific or advanced KCs.

The AUC metric again brings some specific issues. When AUC is computed per knowledge component, the metric does not require any calibration of the model, since the metric only considers relative ordering of predictions. For example, in our simple error curve model, the only relevant aspect of predictions is that they are decreasing, it does not matter what the exact shape of the curve is (all decreasing curves lead to the same value of the AUC metric). The global computation of the AUC metric takes into account the relative calibration among KCs, e.g., if predictions for one of the KCs are too low relative to other KCs, it will decrease the metric value. However, in this case the over-



model	AUC_g	$AUC_{\overline{KC}}$
A1, B2 (correct)	0.73	0.63
A2, B1 (speed mismatch)	0.60	0.63
A1, C (negative learning)	0.68	0.45

Fig. 7 Illustration of the impact of AUC computation. Data generated by a model with knowledge components A1, B2 and fitted by three models.

all AUC is easily dominated by "differences among KCs" with only a limited effect of the "ability to predict within KC".

As a specific example, consider the case demonstrated in Fig. 7. We generate the data with a model with two knowledge components A1 and B2 (full lines). For model comparison, we consider three models and compare the AUC values when computed globally and averaged across KCs. A model with "speed of learning mismatch" (using knowledge components A2, B1) achieves the same performance as the correct model when AUC is computed per KC. whereas for globally computed AUC it achieves poor performance (because the curves A2 and B1 cross, whereas the correct curves A1 and B2 do not). A model which uses one correct KC (A1) and one very poor KC (C, which models "negative" learning) has the results the other way around. It achieves very poor AUC when computed across KCs (due to the inappropriate model of negative learning), whereas it achieves quite a good AUC when computed globally (due to substantial differences between KCs that are captured correctly in the model).

5.4 Usage of Metrics in the Current Research

We conclude our discussion of metrics with an overview of their usage in the current research. A detailed overview of metrics used for evaluation of predictive accuracy of student models is provided by Pelánek (2015). Several other works describe general methodological issues connected with performance metrics. Dhanani et al. (2014) compare metrics in the case of learning model

parameters; they conclude that RMSE is better than AUC for this purpose. Pardos and Yudelson (2013) study the ability of models to identify "moment of learning" and analyze the relationship between this ability and predictive accuracy metrics; the AUC metric again shows poor results. González-Brenes and Huang (2015) briefly mention the differences between global computation and computation per knowledge component and possible relation to the Simpson's paradox.

Both RMSE and AUC are widely used for evaluation of student models. In most cases, however, the exact approach to computation is typically not explicitly specified in papers. In most cases, probably, the used approach is either global (particularly for the RMSE metric) or averaging per knowledge component.

The RMSE metric has been used for example by Beck and Xiong (2013); Gong et al. (2010); Wang and Beck (2013); Wang and Heffernan (2013); Yudelson et al. (2013); Papoušek et al. (2014); Niznan et al. (2015); Liu and Koedinger (2017). RMSE was also used as a metric in the KDD Cup 2010, which focused on student performance evaluation. Examples of papers that use both the AUC metric and some other metric are Käser et al. (2014a,b); Beck and Xiong (2013); Gong et al. (2010, 2011); Cook et al. (2017). There are also many papers that use only the AUC metric for evaluation, for example Beck and Chang (2007); González-Brenes and Mostow (2013); Pardos et al. (2013); Piech et al. (2015); Beck (2007); González-Brenes (2015).

Some papers that use the AUC metric explicitly describe computation per knowledge component computation. Pardos and Heffernan (2011) report AUC for individual knowledge components. González-Brenes et al. (2014) discuss both global computation and averaging over knowledge components. Khajah et al. (2016) use both global computation and averaging over knowledge components and discuss the impact of the choice on comparison with previous work. Several works compute AUC per student and report averages and results of statistical comparisons (Pardos et al., 2012; Baker et al., 2008; Sao Pedro et al., 2013a).

Performance metrics are also used in many other research areas. Result and observations from these areas may provide useful insight for evaluation of student modeling techniques.

The RMSE metric is closely connected to the sum of square errors and mean square of errors. From the perspective of model comparison, all these metrics are equivalent since averaging and square root are monotone operations. The exact equivalence, however, does hold only for the global computation. When the metric is computed with the use of averaging, the result may slightly different. In some domains (particularly in weather forecasting) the mean square error (RMSE without the square root) is called a Brier score (Brier, 1950; Toth et al., 2003) or a quadratic scoring rule (Gneiting and Raftery, 2007). The Brier score is sometimes decomposed into additive components (Murphy, 1973), which provide further insight into the behavior of predictive models.

The ROC curve and AUC metric are successfully used in many different research areas, but their use is criticized for several reasons (Lobo et al., 2008;

Hand, 2009), e.g., because the metric summarizes performance over all possible thresholds, even over those for which the classifier would never be practically used. Marzban (2004) discusses AUC in the meteorology context and shows that "AUC discriminates well between good and bad models, but not between good models".

Fawcett (2004) provides a detailed discussion of the ROC curve and the AUC metric, discussing also averaging issues (with a focus on the construction of the ROC curve). Hamill and Juras (2006) use the context of meteorology to discuss the issue of metric interpretation in the case when the frequency of observed events is not invariant in all samples (which is closely relevant to varying success rates for different knowledge components in student models).

6 Discussion and Recommendations

We have described several methodological nuances that can influence results of experiments with student models. We conclude with a discussion of specific consequences for research practice.

6.1 Understanding Our Data

Researchers who use published data that they did not collect themselves should inquire into details of the used data collection mechanism. It is useful to perform exploration of dataset properties to get the understanding of the data and its potential biases. We should make sure that our results are not superficially created by the data collection mechanism. Careful attention should also be paid to the division of data between training and test set—a proper procedure should be selected according to the purpose of the model.

It is also useful to check the robustness of achieved results (comparison of model performance, fitted parameter values). What happens when we use an artificially shorter number of answers per student? What happens when we use different performance metric (or different averaging approaches in metric computation)? Do the results of model comparison stay the same?

In some cases, it may be impossible to perform required probes—for example, if we have access only to historical data and we care about ordering of items, we cannot perform experiments with different orderings. Specifically, the basic "easier to difficult" progression in item difficulty is desirable. In some form, it is present in most educational datasets. We should take this issue into account, analyze the available data to understand its properties, and explicitly discuss potential limitations. Useful descriptive statistics for this specific purpose include analysis of mean presentation order of each item (used by Khajah et al. (2014)), analysis of the correlation between orderings of different students (used by Pelánek and Jarušek (2015)), or analysis of transitions between items (used by Lopes et al. (2015)).

6.2 Data Collection

Our results also have consequences for the data collection itself. We have illustrated how the use of adaptive techniques leads to datasets that make it difficult or even impossible to compare student models and find mistakes in their specification (e.g., in knowledge components or prerequisites). The adaptive behavior is the purpose of student modeling and is beneficial for students. At the same time, however, it is detrimental for evaluation purposes.

Nevertheless, we can find a reasonable compromise between our different goals. We can modify the behavior of our learning systems in a way that would enable easier evaluation without hampering their primary goal (i.e., student learning). Specifically, we may employ the controlled use of randomization. If some items are chosen randomly (from a reasonably defined set of items), the impact on user experience may be negligible and the collected data can be used for evaluation in a much more straightforward manner than adaptively chosen items—a specific version of this approach was used by Papoušek et al. (2016).

6.3 Choice and Computation of Metrics

We have presented examples that demonstrate potentially substantial differences between different methods ("global", "per knowledge component", "per student") of computation of metrics of predictive accuracy. A natural question is: "Which method is the correct one?" Unfortunately, there is no simple answer to this question—the choice of an appropriate method depends on the specific use case. In some applications we may care mainly about "long-term users", and we do not worry about users who just try a system for a short while, e.g., for systems used schoolwide in formal educational settings. In other cases the "initial impression" is essential and we want the model to work well even for users with few responses, e.g., for commercial systems targeting individual students where the initial impression influences the decision whether to buy a license. Each of these cases requires a different approach to the evaluation of predictive accuracy.

Thus the solution is not to choose a single universal metric and to apply it in all student modeling research. The choice of metric, however, clearly deserves more attention in research. Researchers should provide a rationale for the choice of metric and also enough technical details about the computation of the metric to make their research reproducible. Our analysis of literature suggests that the current state-of-the-art is inadequate in this respect. In many cases, it is not possible to determine whether the reported metric was computed globally or averaged over knowledge components or students.

Our examples also show that the AUC metric can be potentially misleading in several ways. Some of these features have been already noted in research outside of student modeling. In student modeling, however, the AUC metric remains to be heavily used, and in many studies it is the only metric that is

reported. In the light of discussed deficiencies, these kinds of results should be reevaluated using other metrics and taking into account different methods of metric averaging.

From a wider perspective, we should not forget that the ultimate goal of student modeling is to contribute to student learning. Metrics that evaluate model performance on historical data are indispensable for optimization of models and for choosing a suitable candidate from a wide range of model variants. To fully evaluate the merit of a model, however, we need to explore its impact when applied within a learning system.

6.4 Reporting of Data, Experiments, and Results

This paper explicitly highlights many decisions that need to be done in an evaluation and that can influence the results, e.g., the choice of a dataset and potential treatment of present biases, the choice of metric and its computation, the data splitting procedure. For most of these choices there is no "universally correct decision", e.g., we can not pick one universal metric or a universal approach to splitting data into training and a test set. But we should be aware of the choices made and make them explicit in our research reports.

The current research in student modeling often does not report experimental methodology in sufficient detail. This makes the interpretation of results and reproducibility of research more difficult. One obstacle to proper reporting is the diverse notation and terminology used in the field. One goal of this paper is to explicitly identify aspects of experimental evaluation that need more attention and to propose terminology that would facilitate communication about data, experiments, and results.

6.4.1 Publication of Datasets

Currently, most published dataset document only the data itself, but not the way in which the data were collected. As the data collection can have a substantial impact on the interpretation of the data, it is necessary to document data collection mechanism as well. The behavior of adaptive systems is quite complicated (e.g., many parameters often influence the exact choice of items) and typically it is not feasible to document the data collection mechanism up to all details. But authors of datasets should explicitly discuss all major issues and potential limitations due to the data collection mechanism, notably:

- The distribution of answers and attrition biases in data. What is the distribution of the number of answers per student and knowledge component? What influences this number? Is there mastery attrition due to the behavior of the learning system? Is there self-selection bias due to characteristics of user population?
- Item ordering and selection. How is the specific choice and ordering of items from a single knowledge component determined? Is the ordering fixed, ran-

dom, or adaptive? Is there a feedback loop, i.e., are the answers of students used to influence the future choice of items?

6.4.2 Data Splitting

Evaluation of student models is typically done with the use of cross-validation. This is a well-known and general methodology, but it is insufficient to report just "evaluation was done using 10-fold cross-validation". In student modeling, it is necessary to explicitly state what approach has been used for splitting data into a training set and a test set. Specifically, research papers should report:

- Data splitting with respect to items and students. Here we propose to abandon the use of potentially misleading "student stratified" and "item stratified" terminology. Instead, we propose to use terminology based on the "level" keyword (as illustrated in Fig. 4).
- The treatment of the temporal aspect of the data; specifically, whether the predictions are computed dynamically after each answer, or statically at one chosen time point (as illustrated in Fig. 5).

As the primary data splitting procedure, which most closely corresponds to typical applications in learning systems, we propose "student level cross-validation with dynamic prediction updates".

6.4.3 Metrics Computation

The predictive accuracy of models is typically summarised by a performance metric like *RMSE* or AUC. These metrics may be computed in several different ways and the differences can lead to important discrepancies in the evaluation. It is thus necessary to explicitly specify the used approach to averaging in metric computation. To this end we propose the following notation:

- Global computation of a metric, in which all data points treated equally, is the default that is denoted by the plain name of the metric, e.g., *RMSE*, AUC.
- An average of per student metric values is denoted by the metric name with a subscript $\overline{s}$, e.g. $RMSE_{\overline{s}}$, $AUC_{\overline{s}}$.
- An average of per knowledge component metric values is denoted by the metric name with a subscript $\overline{KC}$, e.g. $RMSE_{\overline{KC}}$, $AUC_{\overline{KC}}$.

6.5 Research Priorities

Through the paper, there are pointers to papers whose results may be influenced by the described methodological nuances. The point of this paper is not to pick on specific research studies. Criticizing is easy, doing proper evaluation is hard. Student modeling is a difficult task with many hidden influences. No

evaluation is perfect—there will always be some undocumented details and arbitrary choices. Nevertheless, continuous improvement of evaluation standards is possible. We just have to give the topic enough attention.

This leads us to conclude with a high-level reflection on research priorities for the student modeling community. At the current state of the research, simple models of learning like BKT or logistic models have been extensively explored and there is a natural trend towards building more complex models (e.g., based on Bayes networks or deep learning).

Our discussion, however, shows that the evaluation of student models is complicated by many methodological pitfalls and even the evaluation of simple models is not properly settled. Without having a very clear methodology and notation for performing experiments, we risk that exploration of complex models will lead to misleading results. This danger is illustratively documented by the deep knowledge tracing experiment described in Introduction. Specifically, the issue of robustness of estimated parameters and comparative results needs more attention even for basic models.

Research of the community, of course, does not need a single priority. It is worthwhile to explore both complex models and methodology of evaluation using simple models. The important point is that the methodology should not be neglected.

Acknowledgments

The author thanks members of the Adaptive Learning group at Masaryk University for interesting discussions about methodological issues in the evaluation of adaptive learning systems, particularly Jan Papoušek, Jiří Řihák and Juraj Nižnan, who performed some of the experiments on which the discussion is based.

References

Baker RS (2010) Mining data for student models. In: Nkambou R, Bourdeau J, Mizoguchi R (eds) Advances in intelligent tutoring systems, Springer, pp 323–337

Baker RS, Corbett AT, Aleven V (2008) More accurate student modeling through contextual estimation of slip and guess probabilities in bayesian knowledge tracing. In: Proceedings of Intelligent Tutoring Systems, Springer, pp 406–415

Baker RS, Gowda SM, Wixon M, Kalka J, Wagner AZ, Salvi A, Aleven V, Kusbit GW, Ocumpaugh J, Rossi L (2012) Towards sensor-free affect detection in cognitive tutor algebra. In: Proceedings of Educational Data Mining, ERIC

Beck J (2007) Difficulties in inferring student knowledge from observations (and why you should care). In: Proceedings of Educational Data Mining, pp 21–30

Beck JE, Chang Km (2007) Identifiability: A fundamental problem of student modeling. In: Proceedings of User Modeling, Springer, pp 137–146

Beck JE, Xiong X (2013) Limits to accuracy: How well can we do at student modeling. In: Proceedings of Educational Data Mining, pp 4–11

Bergmeir C, Benítez JM (2012) On the use of cross-validation for time series predictor evaluation. Information Sciences 191:192–213

Bottou L, Peters J, Quinonero-Candela J, Charles DX, Chickering DM, Portugaly E, Ray D, Simard P, Snelson E (2013) Counterfactual reasoning and learning systems: The example of computational advertising. The Journal of Machine Learning Research 14(1):3207–3260

Brier GW (1950) Verification of forecasts expressed in terms of probability. Monthly weather review 78(1):1–3

Cook J, Lynch CF, Hicks AG, Mostafavi B (2017) Task and timing: Separating procedural and tactical knowledge in student models. In: Proceedings of Educational Data Mining, pp 186–191

Desmarais MC, Baker RS (2012) A review of recent advances in learner and skill modeling in intelligent learning environments. User Modeling and User-Adapted Interaction 22(1-2):9–38

Dhanani A, Lee SY, Phothilimthana P, Pardos Z (2014) A comparison of error metrics for learning model parameters in bayesian knowledge tracing. Tech. rep., Technical Report UCB/EECS-2014-131, EECS Department, University of California, Berkeley

Diamantidis N, Karlis D, Giakoumakis EA (2000) Unsupervised stratification of cross-validation for accuracy estimation. Artificial Intelligence 116(1-2):1–16

Dietterich TG (1998) Approximate statistical tests for comparing supervised classification learning algorithms. Neural computation 10(7):1895–1923

Doroudi S, Brunskill E (2017) The misidentified identifiability problem of bayesian knowledge tracing. In: Proceedings of Educational Data Mining

Fancsali SE, Nixon T, Vuong A, Ritter S (2013) Simulated students, mastery learning, and improved learning curves for real-world cognitive tutors. In: AIED Workshops Proceedings

Fawcett T (2004) Roc graphs: Notes and practical considerations for researchers. Machine learning 31(1):1–38

Fawcett T (2006) An introduction to roc analysis. Pattern recognition letters 27(8):861–874

Fogarty J, Baker RS, Hudson SE (2005) Case studies in the use of ROC curve analysis for sensor-based estimates in human computer interaction. In: Proceedings of Graphics Interface 2005, pp 129–136

Gneiting T, Raftery AE (2007) Strictly proper scoring rules, prediction, and estimation. Journal of the American Statistical Association 102(477):359–378

Gong Y, Beck JE, Heffernan NT (2010) Comparing knowledge tracing and performance factor analysis by using multiple model fitting procedures. In: Proceedings of Intelligent Tutoring Systems, Springer, pp 35–44

Gong Y, Beck JE, Heffernan NT (2011) How to construct more accurate student models: Comparing and optimizing knowledge tracing and performance factor analysis. International Journal of Artificial Intelligence in Education 21(1-2):27–46

González-Brenes J, Huang Y (2015) Your model is predictive - but is it useful? theoretical and empirical considerations of a new paradigm for adaptive tutoring evaluation. In: Proceedings of Educational Data Mining

González-Brenes J, Huang Y, Brusilovsky P (2014) General features in knowledge tracing: Applications to multiple subskills, temporal item response theory, and expert knowledge. In: Proceedings of Educational Data Mining, pp 84–91

González-Brenes JP (2015) Modeling skill acquisition over time with sequence and topic modeling. In: Proceedings of Artificial Intelligence and Statistics, pp 296–305

González-Brenes JP, Mostow J (2013) What and when do students learn? fully data-driven joint estimation of cognitive and student models. In: Proceedings of Educational Data Mining, pp 236–240

Gunawardana A, Shani G (2009) A survey of accuracy evaluation metrics of recommendation tasks. The Journal of Machine Learning Research 10:2935–2962

Hamill TM, Juras J (2006) Measuring forecast skill: is it real skill or is it the varying climatology? Quarterly Journal of the Royal Meteorological Society 132(621C):2905–2923

Hand DJ (2009) Measuring classifier performance: a coherent alternative to the area under the roc curve. Machine learning 77(1):103–123

Heathcote A, Brown S, Mewhort D (2000) The power law repealed: The case for an exponential law of practice. Psychonomic bulletin & review 7(2):185–207

Herlocker JL, Konstan JA, Terveen LG, Riedl JT (2004) Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems 22(1):5–53

Hyndman RJ, Athanasopoulos G (2014) Forecasting: principles and practice. OTexts

Jarušek P, Klusáček M, Pelánek R (2013) Modeling students' learning and variability of performance in problem solving. In: Proceedings of Educational Data Mining, pp 256–259

Käser T, Klingler S, Schwing AG, Gross M (2014a) Beyond Knowledge Tracing: Modeling Skill Topologies with Bayesian Networks. In: Proceedings of Intelligent Tutoring Systems, pp 188–198

Käser T, Koedinger KR, Gross M (2014b) Different parameters – same prediction: An analysis of learning curves. In: Proceedings of Educational Data Mining, pp 52–59

Khajah M, Lindsey RV, Mozer MC (2016) How deep is knowledge tracing? In: Proceedings of Educational Data Mining

Khajah MM, Huang Y, González-Brenes JP, Mozer MC, Brusilovsky P (2014) Integrating knowledge tracing and item response theory: A tale of two frame-

works. In: Proceedings of Personalization Approaches in Learning Environments

Klingler S, Käser T, Solenthaler B, Gross M (2015) On the Performance Characteristics of Latent-Factor and Knowledge Tracing Models. In: Proceedings of Educational Data Mining

Koedinger KR, Baker RS, Cunningham K, Skogsholm A, Leber B, Stamper J (2010) A data repository for the edm community: The pslc datashop. Handbook of educational data mining 43

Koedinger KR, Corbett AT, Perfetti C (2012a) The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. Cognitive science 36(5):757–798

Koedinger KR, McLaughlin EA, Stamper JC (2012b) Automated student model improvement. International Educational Data Mining Society

Koedinger KR, Yudelson MV, Pavlik PI (2016) Testing theories of transfer using error rate learning curves. Topics in cognitive science 8(3):589–609

Langford J, Strehl A, Wortman J (2008) Exploration scavenging. In: International Conference on Machine learning, ACM, pp 528–535

Li L, Chu W, Langford J, Wang X (2011) Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In: Web search and data mining, ACM, pp 297–306

Liu R, Koedinger KR (2017) Towards reliable and valid measurement of individualized student parameters. In: Proceedings of Educational Data Mining, pp 135–142

Liu R, Koedinger KR, McLaughlin EA (2014) Interpreting model discovery and testing generalization to a new dataset. In: Processing of Educational Data Mining, pp 107–113

Lobo JM, Jiménez-Valverde A, Real R (2008) AUC: a misleading measure of the performance of predictive distribution models. Global ecology and Biogeography 17(2):145–151

Lomas D, Patel K, Forlizzi JL, Koedinger KR (2013) Optimizing challenge in an educational game using large-scale design experiments. In: SIGCHI Conference on Human Factors in Computing Systems, ACM, pp 89–98

Lopes M, Clement B, Roy D, Oudeyer PY (2015) Multi-armed bandits for intelligent tutoring systems. Journal of Educational Data Mining 7(2):20–48

Marlin B (2004) Collaborative filtering: A machine learning perspective. University of Toronto

Martin B, Mitrovic A, Koedinger KR, Mathan S (2011) Evaluating and improving adaptive educational systems with learning curves. User Modeling and User-Adapted Interaction 21(3):249–283

Marzban C (2004) The roc curve and the area under it as performance measures. Weather and Forecasting 19(6):1106–1114

Murphy AH (1973) A new vector partition of the probability score. Journal of Applied Meteorology 12(4):595–600

Murray RC, Ritter S, Nixon T, Schwiebert R, Hausmann RG, Towle B, Fancsali SE, Vuong A (2013) Revealing the learning in learning curves. In: Pro-

ceedings of Artificial Intelligence in Education, Springer, pp 473–482
Nižnan J, Pelánek R, Papoušek J (2015) Exploring the role of small differences in predictive accuracy using simulated data. In: Proceedings of AIED Workshop on Simulated Learners
Nixon T, Fancsali S, Ritter S (2013) The complex dynamics of aggregate learning curves. In: Proceedings of Educational Data Mining
Niznan J, Pelánek R, Rihák J (2015) Student models for prior knowledge estimation. In: Proceedings of Educational Data Mining, pp 109–116
Papoušek J, Pelánek R (2015) Impact of adaptive educational system behaviour on student motivation. In: Proceedings of Artificial Intelligence in Education, Springer, vol 9112, pp 348–357
Papoušek J, Pelánek R, Stanislav V (2014) Adaptive practice of facts in domains with varied prior knowledge. In: Proceedings of Educational Data Mining, pp 6–13
Papoušek J, Stanislav V, Pelánek R (2016) Evaluation of an adaptive practice system for learning geography facts. In: Gasevic D, Lynch G, Dawson S, Drachsler H, Rosé CP (eds) Proceedings of Learning Analytics & Knowledge, ACM, pp 40–47
Paramythis A, Weibelzahl S, Masthoff J (2010) Layered evaluation of interactive adaptive systems: framework and formative methods. User Modeling and User-Adapted Interaction 20(5):383–453
Pardos ZA, Heffernan NT (2010) Modeling individualization in a bayesian networks implementation of knowledge tracing. In: Proceedings of User Modeling, Adaptation, and Personalization, Springer, pp 255–266
Pardos ZA, Heffernan NT (2011) Kt-idem: Introducing item difficulty to the knowledge tracing model. In: Proceedings of User Modeling, Adaption and Personalization, Springer, pp 243–254
Pardos ZA, Yudelson MV (2013) Towards moment of learning accuracy. In: AIED 2013 Workshops Proceedings Volume 4
Pardos ZA, Gowda SM, Baker RS, Heffernan NT (2012) The sum is greater than the parts: ensembling models of student knowledge in educational software. ACM SIGKDD explorations newsletter 13(2):37–44
Pardos ZA, Bergner Y, Seaton DT, Pritchard DE (2013) Adapting bayesian knowledge tracing to a massive open online course in edx. In: Proceedings of Educational Data Mining, pp 137–144
Pelánek R (2015) Metrics for evaluation of student models. Journal of Educational Data Mining 7(2)
Pelánek R (2017a) Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques. User Modeling and User-Adapted Interaction 27(3):313–350
Pelánek R (2017b) Measuring predictive performance of user models: The details matter. In: Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization, ACM, pp 197–201
Pelánek R, Jarušek P (2015) Student modeling based on problem solving times. International Journal of Artificial Intelligence in Education pp 1–27

Pelánek R, Řihák J (2017) Experimental analysis of mastery learning criteria. In: Proceedings of User Modelling, Adaptation and Personalization, ACM, pp 156–163

Pelánek R, Řihák J, Papoušek J (2016) Impact of data collection on interpretation and evaluation of student model. In: Proceedings of Learning Analytics & Knowledge, ACM, pp 40–47

Piech C, Bassen J, Huang J, Ganguli S, Sahami M, Guibas LJ, Sohl-Dickstein J (2015) Deep knowledge tracing. In: Advances in Neural Information Processing Systems, pp 505–513

Reddy S, Labutov I, Banerjee S, Joachims T (2016) Unbounded human learning: Optimal scheduling for spaced repetition. In: Proceedings of Knowledge Discovery and Data Mining, ACM

Ren Z, Ning X, Rangwala H (2017) Grade prediction with temporal course-wise influence. In: Proceedings of Educational Data Mining, pp 48–55

Sao Pedro M, Baker RS, Gobert JD (2013a) Incorporating scaffolding and tutor context into bayesian knowledge tracing to predict inquiry skill acquisition. In: Proceedings of Educational Data Mining, pp 185–192

Sao Pedro MA, Baker RS, Gobert JD (2013b) What different kinds of stratification can reveal about the generalizability of data-mined skill assessment models. In: Proceedings of Learning Analytics and Knowledge, ACM, pp 190–194

Shani G, Gunawardana A (2011) Evaluating recommendation systems. In: Recommender systems handbook, Springer, pp 257–297

Streeter M (2015) Mixture modeling of individual learning curves. In: Educational Data Mining

Toth Z, Talagrand O, Candille G, Zhu Y (2003) Forecast Verification: A Practitioner's Guide in Atmospheric Science, Wiley, chap Probability and ensemble forecasts, pp 137–163

Van Inwegen E, Adjei S, Wang Y, Heffernan N (2015a) An analysis of the impact of action order on future performance: the fine-grain action model. In: Proceedings of Learning Analytics And Knowledge, ACM, pp 320–324

Van Inwegen EG, Adjei SA, Wang Y, Heffernan NT (2015b) Using partial credit and response history to model user knowledge. In: Proceedings of Educational Data Mining

Volkovs M, Yu GW (2015) Effective latent models for binary feedback in recommender systems. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, pp 313–322

Wager S, Chamandy N, Muralidharan O, Najmi A (2014) Feedback detection for live predictors. In: Ghahramani Z, Welling M, Cortes C, Lawrence N, Weinberger K (eds) Advances in Neural Information Processing Systems 27, Curran Associates, Inc., pp 3428–3436

Wang Y, Beck J (2013) Class vs. student in a bayesian network student model. In: Proceedings of Artificial Intelligence in Education, Springer, pp 151–160

Wang Y, Heffernan N (2013) Extending knowledge tracing to allow partial credit: using continuous versus binary nodes. In: Proceedings of Artificial

Intelligence in Education, Springer, pp 181–188
Wilson KH, Karklin Y, Han B, Ekanadham C (2016a) Back to the basics: Bayesian extensions of irt outperform neural networks for proficiency estimation. In: Processing of Educational Data Mining, pp 539–544
Wilson KH, Xiong X, Khajah M, Lindsey RV, Zhao S, Karklin Y, Van Inwegen EG, Han B, Ekanadham C, Beck JE, et al. (2016b) Estimating student proficiency: Deep learning is not the panacea. In: Proceedings of Neural Information Processing Systems, Workshop on Machine Learning for Education
Xiong X, Zhao S, Van Inwegen E, Beck J (2016) Going deeper with deep knowledge tracing. In: Proceedings of Educational Data Mining, pp 545–550
Yudelson MV, Koedinger KR (2013) Estimating the benefits of student model improvements on a substantive scale. In: EDM 2013 Workshops Proceedings
Yudelson MV, Koedinger KR, Gordon GJ (2013) Individualized bayesian knowledge tracing models. In: Proceedings of Artificial Intelligence in Education, Springer, pp 171–180

Radek Pelánek received his Ph.D. degree in Computer Science from Masaryk University for his work on formal verification. Since 2010 his research interests focus on areas of educational data mining and learning analytics. Currently, he is the leader of the Adaptive Learning group at Masaryk University and is interested in both theoretical research in user modeling and practical development of adaptive educational systems.