

Casting Curves Before Regressions: Limitations in Refining Domain Models with Learning Curves

Ilya Goldin
2U, Inc.
igoldin@2u.com

April Galyardt
Software Engineering Institute
akgalyardt@sei.cmu.edu

Data from student learning provide learning curves that, ideally, demonstrate improvement in student performance over time. Existing data mining methods can leverage these data to characterize and improve the domain models that support a learning environment, and these methods have been validated both with already-collected data, and in close-the-loop studies that actually modify instruction. However, these methods may be less general than previously thought, because they have not been evaluated under a wide range of data conditions.

We describe a problem space of 90 distinct scenarios within which data mining methods may be applied to recognize posited domain model improvements. The scenarios are defined by two kinds of domain model modifications, 5 kinds of learning curves, and 25 types of skill combinations under 3 ways of interleaving skill practice. These extensive tests are made possible by the use of simulated data. In each of the 90 scenarios, we test three predictive models that aim to recognize domain model improvements, and evaluate their performance.

Results show that the conditions under which an automated method tests a proposed domain model improvement can drastically affect the method's accuracy in accepting or rejecting the proposed improvement, and the conditions can be affected by learning curve shapes, method of interleaving, choice of predictive model, and the threshold for predictive model comparison. Further, results show poor accuracy in accepting a proposed improvement by the Additive Factors Model, made popular in the DataShop software. Other models, namely Performance Factors Analysis and Recent-Performance Factors Analysis, are much more accurate, but still struggle under some conditions, such as when distinguishing curves from two skills where students have a high rate of errors after substantial practice.

These findings bear on how to evaluate proposed refinements to a domain model. In light of these results, historical attempts to test domain model refinements may need to be reexamined.

1. INTRODUCTION

A central aim of the learning sciences is to discover exactly how students may come to master a variety of learning objectives. This knowledge can be used to create conditions for learning that are effective, efficient, motivating and replicable. One instrument that the learning sciences can bring to bear on this research problem is a formal definition of the scope and sequence of instruction. When such a definition is made a component in educational technology, as exemplified by

systems for adaptive learning, it is called a domain model.

For example, when the learning objective is introductory computer programming in Lisp, a domain model should distinguish the sub-skills of defining a first parameter to a Lisp function and of defining subsequent parameters to a function (Corbett and Anderson, 1995). As a second example, when the target domain is a course on algorithms and data structures, we can assume that the intended audience of students has long mastered basic programming skills, a domain model of Algorithms might reasonably assume a baseline level of programming ability in the learner population and decline to distinguish skills such as defining first versus subsequent parameters.

A domain model for some target domain can change over time. Because domain models reflect scientific knowledge about how to teach some set of skills or other knowledge components, they are subject to revision as scientific understanding improves. If we treat different plausible domain models for the same learning objective as hypotheses about instruction and test them with learners, we may find evidence in support of one domain model over another, and thereby gain new knowledge about how people come to master this objective.

This work examines whether or not we can be certain that a posited domain model improvement can actually be detected in observed student behaviors. The focus is on assessing the methodology for domain model refinement, not any specific domain model or predictive model.

Intriguingly, we can test domain models against already collected data. Although the inferences from such data mining studies are not as strong as from controlled experimentation, they can be implemented for the cost of running a statistical procedure, and as such can be tremendously useful (Martin et al., 2011; Koedinger et al., 2012; Goldin et al., 2016), especially when domain model improvements are non-obvious (Koedinger and McLaughlin, 2016). The methodology has been validated through studies that “close the loop” by identifying candidate domain model refinements on existing data, and then testing a refined domain model with new students (Koedinger et al., 2013).

Data mining studies for domain model improvement are essentially statistical tests. Subject matter experts posit that some feature of the domain ought to be reflected in student practice in the domain, and create two domain models that differ only with respect to the feature in question. Then they compare which of the models more accurately predicts student activity in the extant dataset. Any statistical test comparing predictions from two domain models is necessarily subject to potential false negative and false positive errors. These errors can lead us to choose a domain model that fails to describe some feature of the domain correctly, and therefore give rise to inefficient instruction and inaccurate assessment.

In the following sections, we discuss how to test for domain model improvement. We observe that the test implemented in state-of-the-art software (Cen et al., 2006) is prone to false positive and false negative errors, and that such errors are especially likely under certain conditions. We define a comprehensive set of conditions that a test ought to pass, arising from our real-world experience with educational data mining and from the straightforward assumption that learner performance on tasks should improve with practice. We then simulate data in each of these conditions and apply three statistical tests, including the original test using the Additive Factors Model (Cen et al., 2006), and alternative tests using the models Performance Factors Analysis (Pavlik Jr et al., 2009) and Recent-Performance Factors Analysis (Galyardt and Goldin, 2015; Goldin and Galyardt, 2015). We find that some data generative conditions are inherently more challenging than others for detecting domain model improvements. Further, we find that the Additive Factors Model has a low rate of recognizing valid domain model improvements,

and that the alternative tests are more sensitive.

1.1. RELATED WORK

A simple computational representation for a domain model is a Q-matrix (Tatsuoka, 1983). A Q-matrix is a table: each row corresponds to one assessment item in the domain of interest, or a sub-item, such as a step in the solution of a problem; each column corresponds to a latent skill or knowledge component variable hypothesized to matter in the domain; and each table cell contains a (usually binary) value that encodes whether or not some skill is relevant to some assessment item or sub-item. Using this representation, two models for the same domain may differ if they contain different values in one or more cells, or if they contain different columns.

It is possible to try to discover an entire Q-matrix in a data-driven way (Liu et al., 2012; Desmarais, 2012). By using repeated observations of a learner’s performance on a skill, it is possible to discover a domain model and learn student proficiency parameters at the same time (González-Brenes and Mostow, 2013). Nonetheless, it is challenging to understand how loadings on an automatically discovered Q-matrix connect to a curriculum and to human expert understanding of a domain.

One way to bridge data-driven methods and human understanding is to refine an existing Q-matrix rather than learning one from scratch. In domain model refinement, we only need to interpret a change to a Q-matrix (Koedinger et al., 2012). The proposed changes may be human-authored also, such as the “difficulty factor” (Baker et al., 2007), further facilitating interpretability.

Learning Factors Analysis (Cen et al., 2006) automates a heuristic search in a space of Q-matrices, enabling iterative refinement that starts with an existing Q-matrix. LFA consists of three parts: a procedure to search the space of domain models; a set of 3 change operators that may alter a domain model; and a domain model evaluation component, the output of which guides the search. The three change operators are split, merge, and add. The split operator replaces a single latent variable skill variable in the Q-matrix with two distinct variables, each of which only codes for a subset of the activities that bear on the original latent. The merge operator does the reverse, replacing two existing latent variables with a new latent that is their union. The add operator inserts an extra Q-matrix column, creating an additional loading on a latent variable without removing any existing loadings. Each posited domain model refinement calls for an application of these operators and necessitates a new comparison to evaluate the modified Q-matrix.

Cognitive Diagnostic Models also use a Q-matrix to represent how latent skills may combine to generate particular test-taker response patterns. In this literature, domain model refinement is known as correction of a misspecified Q-matrix. For example, the G-DINA model provides a basis for a discrimination index that indicates for each test item whether the item appropriately discriminates between “individuals who possess all attributes required for item” and “individuals who lack one or more required attributes for the item” (de la Torre and Chiu, 2016). This index therefore indicates a possible misspecification in the Q-matrix.

Data from test administrations and from tutoring applications differ in significant ways. In most test administrations, by design, we assume that students do not learn during the test, and therefore models of test data can ignore any order effects among multiple assessments of one skill. By contrast, in tutoring data, we hope for learning, and therefore models of these data need to account for temporal effects. In addition to differences in model structure, the assumption that

performance on a skill will improve over time provides important information to the tutoring-data models that is not available to test-data models, as leveraged in this paper.

Specifically, data from student practice with some set of learning activities, can be plotted as a set of learning curves. It is not trivial to plot one curve per skill because of compensatory skill relationships, condensation rules, and credit assignment. (Rupp and Templin, 2008) Nonetheless, given a dataset from student practice that is defined with respect to a domain model, we can plot learning curves: for each skill in the domain model, for each practice opportunity, we can plot the average population probability of success on the skill (or equivalently, the error rate, flipping the vertical axis).

A domain model may represent information beyond the mapping between assessment items and relevant skills. Depending on the instructional setting, a tutoring system may also represent the prerequisite structure, i.e., the dependencies among the skills, which may inform both instruction and assessment. Prerequisite dependencies may also be learned automatically (Vuong et al., 2011; Scheines et al., 2014; Chen et al., 2016).

A Q-matrix is not the only possible implementation of a domain model. For example, Knowledge Spaces (Falmagne et al., 1990) do not represent the notion of a skill; instead, they represent assessment items and prerequisite relationship among these.

A domain model may be brought to bear on predicting student performance. For example, a variety of Linear Logistic Test Models (Fischer, 1973; de Boeck and Wilson, 2004), including the logistic regression models used in this work (Cen et al., 2006; Pavlik Jr et al., 2009; Galyardt and Goldin, 2015), explicitly incorporate a Q-matrix. (See section Predictive Models) Alternatively, a Bayesian Knowledge Tracing predictive model (Corbett and Anderson, 1995) implicitly uses a Q-matrix by allocating independent parameters for each skill.

The development of predictive models stems from the need for large-scale testing, and is described in decades of psychometric literature. More recently, predictive models have been at the center of literature on educational data mining. Their most prominent use is to predict whether or not a student will answer a problem correctly, which enables decision-making in tutoring systems (Corbett and Anderson, 1995). This use case has prompted investigations of model variants, of their predictive accuracy and of model errors (Yue et al., 2011; Käser et al., 2014; Stamper et al., 2013; Galyardt and Goldin, 2015). Predictive models have also been used to connect human learning to domain model refinement (Cen et al., 2006; Martin et al., 2011; Koedinger et al., 2012; Koedinger et al., 2013).

The structure of a predictive model can reflect different conceptualizations of human learning and performance. For example, a population-level view of learning implies that average population performance on some set of tasks ought to improve as the population learns. By contrast, a mastery-disaggregated view of learning (Murray et al., 2013) holds that individual students may engage in practice for some time before demonstrating correct performance, leading to several models (Käser et al., 2014; Galyardt and Goldin, 2015).

Domain model improvements have multiple uses in tutoring systems, e.g., more accurate reporting, sequencing problem within a skill, creating new tasks or refining instructional messages (Koedinger et al., 2013). Additionally, domain models have multiple uses beyond tutoring. In large-scale testing, a domain model can enable detailed diagnostic reporting and accurate summative assessment, including in adaptive testing settings. In Cognitive Task Analysis, a “task list”, which is effectively the set of skills (rows) in a Q-matrix, enables comparison among human expert instructors in terms of what information they fail to communicate to students (Sullivan et al., 2014). In psychology research, domain models may be used to investigate cog-

nitive hypotheses, such as in comparing a faculty view of cognition against a component view (Koedinger et al., 2016).

Aside from investigating the validity of a posited change, as in this paper, it may be possible to estimate the impact of a posited change, e.g., on student practice in a tutoring system (Cen et al., 2007; Rollinson and Brunskill, 2015; Käser et al., 2016; Gonzalez-Brenes and Huang, 2015). Errors in domain models may have a significant impact on students. Fanscsali et al., (2013) uses simulations to show that when a model is mis-specified, over 50% of the students may be assigned substantial extra practice. When a single latent variable in a Q-matrix improperly represents two distinct skills, practice and assessment on either skill is not distinguished from practice and assessment on the other. A student might master one of the two skills, but a tutoring system may miss the mastery and assign extraneous practice, or, conversely, graduate the student prematurely from studying one skill because of mastery of the other. When two latent variables improperly represent the same skill, the student may be obligated to demonstrate mastery of multiple (unnecessarily split) skills, slowing down instruction and increasing student effort.

Domain model refinements for both adding additional skills and expanding the Q-matrix, as well as choosing a single skill domain model can be found in the literature, and these refinements directly affect the practice of future students. As a first example, in geometry, students may struggle with composite area problems, where “the area of a composite shape must be found by combining (adding or subtracting) the areas of two constituent regular shapes (e.g., what’s left when a circle is cut from a square)” (Koedinger et al., 2013). The researchers posited that a subset “of the composite problems were ‘scaffolded’ such that they included columns that cued students to find the component areas first”. As a revised domain model, problems that had been tagged with one column of the Q-matrix were re-tagged with one of three new columns: “one representing ‘compose-by-addition’ with scaffolding present, a second where the student had to ‘decompose’ a composite area without scaffolding, and a third where the student needs simply to ‘subtract’ in order to execute a decomposition plan.” This split was validated both in terms of predictive accuracy and in an experiment with students (Koedinger et al., 2013).

Another paper from the same research team provides an example where a Q-matrix expansion was not justified. In a dataset of student practice in computing the area of various simple, non-composite shapes, it was theorized that finding forward computation (e.g., find the area of a circle given the radius) may be distinct from backward computation (find the radius given the area). In this case, predictive accuracy only improved by recoding circle-area problems. Analogous recoding for other shapes, including triangle, trapezoid, etc., did not improve predictive accuracy (Koedinger et al., 2012). Therefore the single skill coding for forward and backward computation of area was selected as the better domain model.

2. PROBLEM STATEMENT

We aim to determine whether or not a method is capable of accurately and consistently recognizing true domain model improvements. Our research questions are: How often do predictive models fail to recognize improvements to a domain model, under what conditions are these errors likely, and do these conditions vary across the predictive models?

This section describes three factors that can affect the accuracy of recognizing domain model improvements. These include the structure of the underlying generative domain model, the shapes of learning curves for the skills in question, and the order in which students see problems

from these skills.

Additional factors may also affect accuracy of recognizing improvements, including sample size and levels of noise in the data. A comprehensive investigation of these issues is beyond the scope of our analysis, although we make a preliminary investigation of sample size.

2.1. STRUCTURE OF THE GENERATIVE DOMAIN MODEL

As mentioned in [Related Work](#), student practice in geometry problem solving is at times better represented by expanding a domain model, and on other occasions by using a parsimonious model ([Koedinger et al., 2012](#); [Koedinger et al., 2013](#)). As in these examples, the simplest domain model refinement possible is to decide whether a particular set of problems is better modeled by a single skill or a pair skills.

Accordingly, an automated procedure that recognizes domain model refinements must be evaluated in two ways. Suppose that we could know the true, unobservable domain model underlying a dataset of student practice. When the data arise from a domain model with a single skill, does the procedure correctly select the single skill domain model? When the data is generated by a domain model with two skills, does the procedure correctly select the two skill model? It is important to evaluate an automated refinement recognition procedure in both ways, because a procedure may be biased to reward either expansive or parsimonious domain models. For example, a predictive model that overfits the data would tend to value expansive domain models more often than justified.

2.2. LEARNING CURVE SHAPES ¹

The rate at which students improve with practice and the way this improvement is parameterized by different predictive models should be expected to play a large role in whether or not an improvement in a domain model can be automatically recognized. We identify 5 typical patterns of learner performance on a given skill, inspired by the categorization of learning curves in the PSLC DataShop ([DataShop Team, 2016](#)).

In a Good curve (Figure 1) we observe somewhat idealized student behavior. On early attempts, the percentage of students who answer correctly is fairly low, indicating that the majority of students have yet to master the skill. After some sensible amount of practice, the proportion of students who answer correctly is high, indicating learning.

The High Error curves (Figure 1, top center), represent skills where students have a high probability of making an error despite correct knowledge (i.e., there is a high slip rate for the skill). For example, in basketball there is a low error rate for a free throw, but a three-point shot has an inherently higher error rate as it is taken from a longer distance and opponents are free to move around, and this remains true even for highly skilled players. Assessment of High Error skills must account for this slip rate.

If the population error rate declines slowly (Figure 1, Slow Learning), or almost not at all (Figure 1, Negligible Learning), students may need many practice opportunities to master the skill. Potentially, one curve might in fact be a mixture of two skills, and averaging the skills together obscures the distinct error rates on the two skills. This may indicate an opportunity to revise instruction on this skill, or to consider whether there may be sub-skills that can be called out for targeted instruction and assessment. Alternatively, a Slow Learning skill may just be a skill that takes many more repetitions to master than the idealized Good skills.

¹This section is based in part on ([Goldin et al., 2016](#))

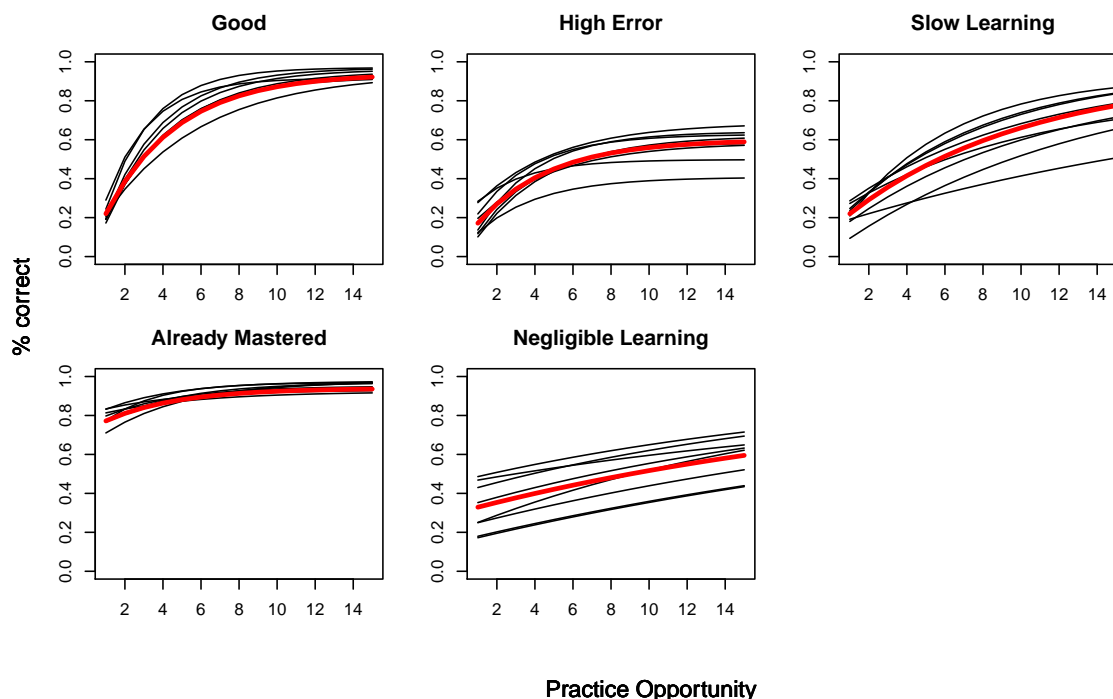


Figure 1: Exemplars of learning curve shapes. Thick red line denotes a mean curve for each shape according to the probability distributions in Table 1.

When the population error rate is low even at early practice opportunities (Figure 1, Already Mastered), the data indicate that by the time the observed students encountered these activities, they must have already mastered the skill in question.

The learning curve shapes described above are similar but not identical to the curves in the PSLC DataShop (DataShop Team, 2016). In brief, the Good curve type is similar in both settings. The DataShop “low and flat” curve is comparable to Already Mastered in this paper. The DataShop “still high” curve is replaced by two distinct shapes: Slow Learning shows that students may need additional opportunities for practice, and High Error shows that the task is inherently difficult or has a high slip rate, and even additional practice may not reduce error over the long run. Finally, the DataShop “no learning” shape is replaced by Negligible Learning, which still represents that students fail to acquire mastery over time, but does allow for a modest aggregate population improvement in performance.

2.3. STRUCTURE OF STUDENT PRACTICE

Consider again the examples from Koedinger et al., (2013) and Koedinger et al., (2012) refining the domain model in high school geometry. These studies were conducted on already collected data. The students encountered problems according to a structure determined by the domain model at that time. When a new domain model is proposed, structure of practice for the new skill(s) is different.

As a more concrete example: as students worked through problems on computing the area of simple shapes, they encountered both forward and backward computations. The forward and

backward computations could have been mixed up in several different ways. At one extreme, students could have encountered all of the forward problems first, and all of the backwards problems later; we call this *blocked interleaving*. At the other extreme, students could have encountered them *evenly interleaved*; forwards, then backwards, then forwards, etc. As a third, more realistic case, students could have encountered mostly forward computations at the beginning of practice, and mostly backwards computations at the end with problems somewhat mixed together in the middle; we call this *gradual interleaving*.

Blocked, even and gradual modes of interleaving, which provide structure to this analysis, are not necessarily tools that are purposefully applied in domain modeling. Comparing these three interleaving regimes to any real-world systems would be besides the point. Rather, developers of a tutoring system may focus on practical concerns when creating domain models, such as limiting the expected amount of student effort (Lee and Brunskill, 2012), and a side effect of that may be some kind of interleaving of skills. Nonetheless, the interleaving regimes circumscribe the continuum of how falsely merged curves might be shaped.

One historically important example for learning curve analysis demonstrates an interleaving of two curves, corresponding to the rule for the first argument and the rule for subsequent arguments to a Lisp function (Corbett and Anderson, 1995, Figures 5-6). The curve for the rule pertaining to the first argument, with a first-opportunity error rate of 30% and subsequently approaching virtually no error after practice, is closest to our category of Already Mastered curves. The curve for the rule on additional arguments, with a first-opportunity error rate of 55% and approaching no error after practice, most closely matches the category of Good curves. The second curve begins at practice opportunity 6 relative to the first curve, suggesting that the curve overlap resembles gradual interleaving more so than blocked or evenly interleaved practice.

2.3.1. Generative model with 2 skills

In the case that the true domain model contains two skills, and a posited domain model inappropriately tags their items with only one skill, the practice opportunities must be ordered somehow in the merged skill. The Blocked, Even, and Gradual *interleaving regimes* described above provide a useful structure to explore the ordering effects of a domain model incorrectly merging what should be two distinct skills.

Figure 2 shows what the learning curve for the new single-skill candidate looks like for when both generative skills have a Good curve shape. Under Blocked practice, students complete practice on one skill before encountering the second skill; and the learning curve for an inappropriately merged skill will have a spike in the error rate when students begin to practice the second skill (Figure 2 Blocked). In Even interleaving (where students practice one activity from each of the skills, then a second activity from each of the skills, etc.), the merged curve is reasonably smooth (Figure 2 Even). It is not clear that applying the LFA split operator to such a curve would result in a statistically detectable difference. Under Gradual interleaving, practice on one skill mostly happens first, practice on the second skill mostly happens last, and the skills overlap in the middle of practice. This leads to several smaller spikes in the incorrectly merged curve as opposed to the single large spike in the Blocked case. Note that Figure 2 shows the case when both generating curves are type Good; for other curve shapes, the combined curves have similar structure, but the spikes may be smaller or larger (Figures 8-10).

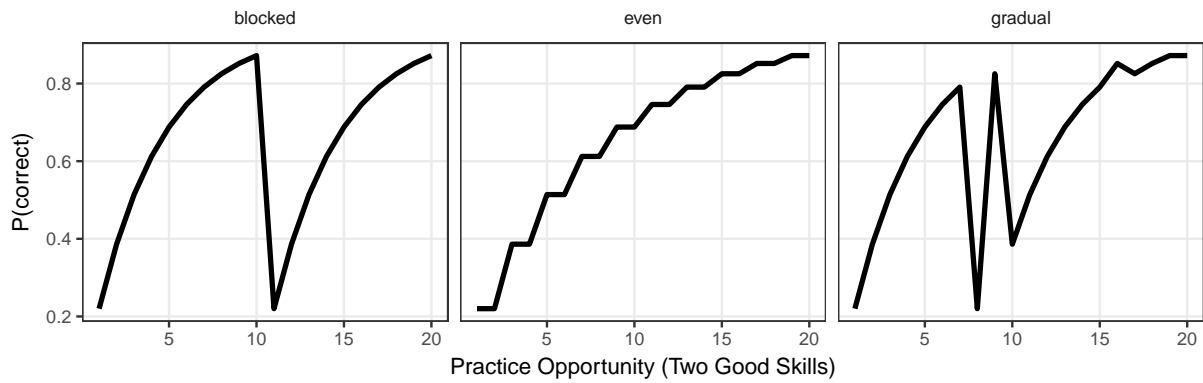


Figure 2: Single-skill learning curves when practice on two Good skills is blocked, evenly interleaved, or gradually interleaved

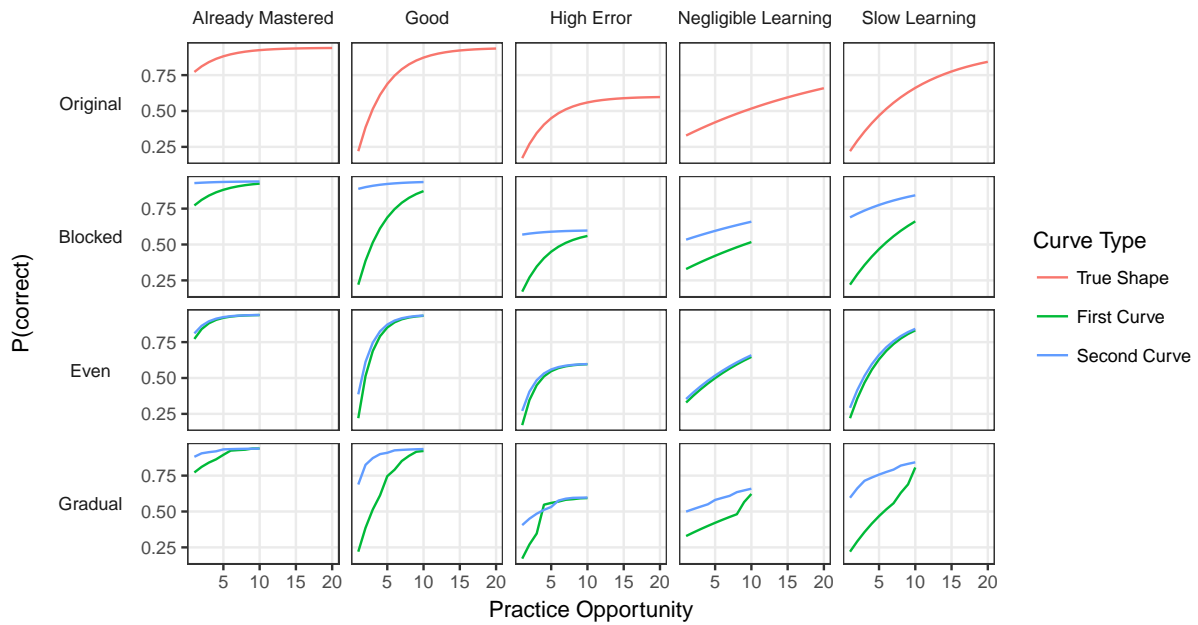


Figure 3: Effects of two-skill coding on the exemplar curve shapes according to blocking, even interleaving, or gradual interleaving.

2.3.2. Generative model with 1 skill

In the case that the true domain model contains one skill, and a posited domain model inappropriately splits its items into two skills, the practice opportunities must be allocated somehow to the two skills. The same *interleaving regimes* above also apply here. Presume, for simplicity, that each inappropriately split off skill has 50% of the practice. In the Blocked case, all practice on one of the skills precedes all practice on the other skill, and the generating curve is effectively cut into a first and a second half (Figure 3, Blocked row). With Even Interleaving, practice on the split skills alternates, which means that the two resulting curves in the two-skill coding are highly similar. Lastly, in the Gradual case, the beginning part of the generating curve is mostly allocated towards one skill, and the ending of the generating curve is mostly allocated towards the second skill, but the resulting curves are more jagged than observed in the Blocked case (Figure 3, Gradual row).

In sum, learning curve shape may reveal opportunities for improving the domain model, but it can also hide distinct underlying skills, or it might be improperly split off from another skill. The shapes of an improperly combined curve or improperly separated curves are necessarily affected by the mechanism of interleaving practice. Some combinations of curves and interleaving mechanisms may be especially challenging for computational methods of testing proposed splits and merges, i.e., curve shape combinations may vary in the rates of errors that they induce.

3. METHODS

We aim to evaluate the methodology of automated recognition of domain model improvements, and to describe the conditions under which the automated procedure commits errors. Although domain models are inherently unobservable, we have access to the ground truth for our domain model because we create a synthetic dataset, thereby controlling the domain model, the skill properties and the interleaving or sampling mechanism.

We use simulated data so that, by construction, we can know whether the correct model is the original or the proposed modification. By contrast, it is difficult to know the ground truth regarding a domain model in data collected from human learners, because domain models are unobservable, latent variables, they are designed by fallible human experts, and noise in human performance data can muddy the waters. Even in trivially simple or artificial domains, noise may arise because individual learners pursue distinct problem-solving strategies and shift strategies over time (Anderson, 2013; Galyardt, 2012).

Moreover, simulated data heads off methodological confounds. First, when working with real data, we conventionally rely on a statistical procedure to tell us whether a particular domain model refinement is an improvement. In this paper, it is that very statistical procedure we wish to evaluate. Suppose that, for a given real-world dataset, we could achieve similar predictive accuracy by positing a richer domain model, or by positing a richer model of student proficiency with a basic domain model. Should we believe that the richer domain model or the basic one is closer to the true model?

Second, the questions pertain to interactions of factors, such as recognizing when a single learning curve should be split into two, where the component curves have particular shapes, and when they are interleaved in a particular way. Such interactions are the focus of our analysis, but they are infrequent events in data from human learners. Simulation allows for an unlimited quantity of data, enabling accurate, reliable measurement.

Table 1: Generative distributions for BKT parameters. Distributions that differ from the Good shape in bold.

BKT Parameter	Good	High Error	Slow Learning	Already Mastered	Negligible Learning
L0	$\sim \beta(3, 18)$	$\sim \beta(3, 18)$	$\sim \beta(3, 18)$	$\sim \beta(\mathbf{52}, \mathbf{13})$	$\sim \beta(\mathbf{3}, \mathbf{8})$
T	$\sim \beta(15, 50)$	$\sim \beta(15, 50)$	$\sim \beta(\mathbf{5}, \mathbf{45})$	$\sim \beta(15, 50)$	$\sim \beta(\mathbf{4}, \mathbf{96})$
G	$\sim \beta(4, 36)$	$\sim \beta(4, 36)$	$\sim \beta(4, 36)$	$\sim \beta(4, 36)$	$\sim \beta(4, 36)$
1-S	$\sim \beta(94, 6)$	$\sim \beta(\mathbf{36}, \mathbf{24})$	$\sim \beta(94, 6)$	$\sim \beta(94, 6)$	$\sim \beta(94, 6)$

As described above, we consider 2 generative domain models (single-skill and two-skill), 5 learning curve shapes, and 3 modes of interleaving. This yields $5 \times 5 \times 3 = 75$ scenarios in the two-skill generative case, and $5 \times 3 = 15$ scenarios in the single-skill case.

In each scenario, we generate a dataset with 20 skills with Good curve shapes (Figure 1). These data serve as a basis for estimating parameters relating to the properties of a typical skill (Section [Predictive Models](#)). These skills are always correctly coded in the domain model. To test the domain model refinement, we generate a sidecar dataset corresponding to the simulation condition. Each skill is practiced by 100 simulated students, who make an average of 10 attempts per skill, or more precisely p_i attempts, where $p_i \sim \text{Poisson}(10)$. We chose a mean of 10 opportunities for each person for each skill, as a *best case* realistic scenario - it is near the average number observed per skill in many real data sets. The sidecar dataset is then combined with the main dataset. In each of the 90 scenarios, the simulation is repeated 500 times.

3.1. SIMULATING DATA FROM LEARNING CURVE ARCHETYPES

We use Bayesian Knowledge Tracing (BKT) to synthesize both the main and sidecar datasets ([Corbett and Anderson, 1995](#)). BKT does not represent complex detail of human cognition, but it is useful to treat it as an idealized, abstract model of observable student behavior. (BKT’s limitations and some alternative models are discussed under [Limitations](#)). Since we are testing whether refinements to the domain model are detectable, we do not have to assume that the generating model is “correct”, we merely need a method for generating correct and incorrect responses by synthetic students that maps well to the learning curve shapes defined in section [Learning Curve Shapes](#).

BKT is a Hidden Markov Model with four parameters and manipulating these four parameters produces curves in each archetypal learning curve shape.

- L0, the probability of having already mastered a skill at the start of practice
- T, the probability of transitioning from a lack of skill mastery to mastery
- G, the probability of guessing correctly in the state of lack of mastery
- S, the probability of answering incorrectly in the state of mastery

For each skill in both the main and sidecar datasets in each simulation, each of the four BKT parameters is drawn from a distribution appropriate to the parameter and intended skill

shape (Table 1). The black lines in Figure 1 are exemplars of curves drawn from each of these distributions. Randomizing the BKT parameters in this way allows each of the 500 replications within each of the scenarios to be subtly different, so that we are thoroughly testing the method under each set of conditions.

3.2. GENERATIVE MODEL WITH 2 SKILLS

In this set of scenarios, we generate the main data set as described above, and the sidecar data set contains data which was generated from two independent skills. We then create a one-skill coding of this sidecar data set that incorrectly combines these two skills into one skill. We then look at which domain model makes more accurate predictions.

Both of the generative skills can have any one of the 5 typical learning curve shapes (Figure 1), leading to $5 \times 5 = 25$ possible combinations, and there are 3 different practice interleaving structures for the one-skill model, resulting in 75 conditions for this set of scenarios. The Blocked and Evenly Interleaved conditions, described above, are both deterministic orderings, while the Gradual Interleaving is stochastic.

For gradually interleaved practice, practice opportunities are combined into a merged Q-matrix column such that early practice is more likely to be sampled from one skill, and late practice from another skill. Specifically, given k observations $o_1..o_k$, the probability that the j^{th} of k observations belongs to skill one is proportional to $\frac{1}{0.1+j(1.9/k)}$. As a result of this heuristic, o_1 is 20 times more likely to be sampled from skill one than o_k , and opportunity $o_{k/2}$ is equally likely to belong to either skill.

In this set of scenarios, it is useful to think of the generating skills as being completely unrelated, such as tying your shoes and learning to sound out words. In the generative model, a student’s mastery of one skill is completely statistically independent of their mastery of the other skill. Such independence between skills could never be observed in real student data (in developed countries, most 5 year olds are practicing both early reading and tying their shoes). In real data, the factors we are seeking to isolate would be confounded; we could never be sure if a learning curve had a particular shape because two skills were really one, or because they were two skills with relatively evenly interleaved practice (Figure 2, Even). By construction, simulated data achieves statistical independence and freedom from lurking variables, and allows us to be certain that if we cannot correctly detect which domain model is better, that it is due to one of the experimental factors, not uncertainty in the underlying domain.

3.3. GENERATIVE MODEL WITH 1 SKILL

In this set of scenarios, the data in the sidecar dataset is generated from a single skill. We then create a two-skill coding of this data which incorrectly splits items relating to this skill into two parts, and then evaluate which domain model makes more accurate predictions. In this set of scenarios the single generating skill can have any one of the 5 typical learning curve shapes, and there are three interleaving modes possible, making 15 conditions in this set of simulations.

To provide sufficient data for model training, we generate $p_i \sim Poisson(20)$ practice opportunities for each student i . In the Blocked and Even Interleaving conditions, these p_i practice attempts are assigned deterministically as described above. And once again, practice opportunities in the Gradual Interleaving condition are assigned to the two skills stochastically such that early practice is more likely to be tagged with one skill, and late practice with another skill. Specifically, given k observations $o_1..o_k$, the probability that the j^{th} of k observations belongs

to skill one is proportional to $\frac{1}{0.1+j(1.9/k)}$. As a result of this heuristic, o_1 is 20 times more likely to belong to skill one than o_k , opportunities $o_2..o_5$ usually include 1-2 trials of the second skill, and opportunity $o_{k/2}$ is equally likely to belong to either skill.

3.4. PREDICTIVE MODELS

In the methodology for recognizing domain model refinements that is the focus of this evaluation, a domain model that encodes skills correctly ought to enable better predictive accuracy and parsimony than a domain model that unnecessarily combines or distinguishes skills.

To create predictions, the domain model must be embedded in a predictive model. We compare the predictive models Additive Factors Model (AFM), as in the original LFA algorithm (Cen et al., 2006), the Performance Factors Analysis (PFA) model (Pavlik Jr et al., 2009), and the Recent-Performance Factors Analysis (R-PFA) model (Galyardt and Goldin, 2015; Goldin and Galyardt, 2015). The models are similar in that they are logistic regressions, modeling the probability of a successful response after some history of practice. They all incorporate a Q-matrix as part of the model structure so that each student activity is described in terms of its underlying skills. To various degrees, they are sensitive to improvements in the domain model for purposes of their predictions. (Goldin and Galyardt, 2015)

The models differ in how they represent prior practice because they embody distinct conceptualizations of learning curves. The AFM model represents the traditional learning curve (aggregating based on absolute counts of practice), based on the consideration that the total quantity of a student’s practice with a skill is predictive of the probability of subsequent success. The PFA model differs from AFM in that it distinguishes the total quantity of past successful practice from the total quantity of past unsuccessful practice. By contrast, a mastery-disaggregated learning curve (Murray et al., 2013) is the basis for the R-PFA model, such that a student’s proportion of recent, not total, successful practice is predictive of subsequent success. (Goldin and Galyardt, 2015)

We fit the AFM, PFA and R-PFA models to each simulated dataset once with the correct Q-matrix, and once with the improper Q-matrix. All models omit the student parameter, because the generative BKT does not distinguish students of different ability.

$$Pr(X_{ijt} = Correct) = Q_j * (\beta_j + \gamma_j T_{ijt}) \quad (\text{AFM})$$

$$Pr(X_{ijt} = Correct) = Q_j * (\beta_j + \alpha_j S_{ijt} + \rho_j F_{ijt}) \quad (\text{PFA})$$

$$Pr(X_{ijt} = Correct) = Q_j * (\beta_j + \delta_j R_{ijt}^{0.7} + \rho_j F_{ijt}^{0.1}) \quad (\text{R-PFA})$$

Where the terms are:

- X_{ijt} : (observed) the correct or incorrect outcome of practice by student i on skill j at opportunity t
- T_{ijt} : (observed) the total count of prior practice opportunities by student i on skill j at opportunity t
- S_{ijt} : (observed) the count of successful prior practice opportunities by student i on skill j at opportunity t

- F_{ijt} : (observed) the count of unsuccessful prior practice opportunities by student i on skill j at opportunity t
- $R_{ijt}^{0.7}$: (observed) the proportion of successful prior practice by student i on skill j at opportunity t
- $F_{ijt}^{0.1}$: (observed) the proportion of unsuccessful prior practice by student i on skill j at opportunity t with exponential decay 0.1
- Q_j : (given) the Q-matrix column for skill j
- $\beta_j, \gamma_j, \alpha_j, \rho_j, \delta_j$: (estimated) model parameters

In all models, all estimated skill parameters (both intercepts and slopes) are random effects, which use pooling to borrow information about typical skill shapes (Gelman and Hill, 2006). The exponential decay weights for R-PFA, $R^{0.7}$, $F^{0.1}$, are the best-performing weights from prior work (Galyardt and Goldin, 2015). By fixing R-PFA weights, the number of parameters is $2 \times m$ in AFM, and $3 \times m$ in PFA and R-PFA, where m is the number of skills.

For each fit, we compute AIC, which is the metric used in the original LFA procedure. Given two models for one dataset, AIC rewards the model that has relatively better predictive accuracy, and penalizes the model that uses relatively more parameters. AIC is preferable to R^2 , because using R^2 as a model selection criterion is equivalent to using the MSE of training data, and it will *always* select a model that has been overfit (Hastie et al., 2009). Further, R^2 is difficult to interpret when the number of parameters differs between domain models (Martin et al., 2011). In contrast, as sample size increases, model selection using AIC converges to be equivalent to model selection using cross-validation (Stone, 1977), and it correlates well with cross-validation on data from learning curve models (Stamper et al., 2013).

We then compute the AIC difference between the models using the two Q-matrices. If the AIC difference is sufficiently high (i.e., exceeds some minimum threshold), the predictive model is sensitive to an improvement in the domain model. (Algorithms 1 and 2) We compute the percentage of simulated datasets for which the AIC difference exceeds threshold a for each simulation condition and for each predictive model, i.e., the accuracy rate for choosing the correct model.

Algorithm 1 *isAccurate2Skill*: Is a predictive model accurate when 2 skills are combined?

Require: *simCondition* simulation condition {CurveA, CurveB, InterleavingRegime}

Require: *model* predictive model

Require: *minAICDif* minimum AIC difference threshold

$dataA \leftarrow simulateBKT(simCondition.CurveA)$

$dataB \leftarrow simulateBKT(simCondition.CurveB)$

$dataTwoCurves \leftarrow combine(dataA, dataB, simCondition.InterleavingRegime)$

$fit1 \leftarrow fitModel(model, dataTwoCurves, Q_1)$

$fit2 \leftarrow fitModel(model, dataTwoCurves, Q_2)$

$AICDif \leftarrow AIC(fit1) - AIC(fit2)$

return $minAICDif < AICDif$ indicates correct model is more accurate.

We consider three different threshold values for *similar*: $a = (10, 5, 3)$. Based on the definition of AIC and our practical experience, an AIC difference of 10 is relatively conservative,

Algorithm 2 *isAccurate1Skill*: is a predictive model accurate when 1 skill is split?

Require: *simCondition* simulation condition {TrueCurve, InterleavingRegime}

Require: *model* predictive model

Require: *minAICDif* minimum AIC difference threshold

$dataOneCurve \leftarrow simulateBKT(simCondition.TrueCurve)$

$dataTwoCurves \leftarrow sample(dataOneCurve, simCondition.InterleavingRegime)$

$fit1 \leftarrow fitModel(model, dataOneCurve, Q_1)$

$fit2 \leftarrow fitModel(model, dataTwoCurves, Q_2)$

$AICDif \leftarrow AIC(fit1) - AIC(fit2)$

return $minAICDif > AICDif$ indicates correct model is more accurate.

i.e., a difference greater than 10 implies that one domain model leads to much greater predictive accuracy than another; the higher-AIC model is 0.007 times as probable as the lower-AIC model to minimize information loss. An AIC difference of 3 is sensitive to relatively smaller improvements in predictive accuracy; the higher-AIC model is 0.22 times as probable as the lower-AIC model to minimize information loss.

If the difference between two AIC scores is less than the threshold, then the fit of the models is similar. In this case, standard practice is to choose the more parsimonious model, because there is no evidence that the model with more parameters is substantially better. Accordingly, when the difference between the AIC scores is below the threshold, we consider the 1-skill model to be better, as it is a simpler representation of how student practice predicts future performance.

The set of simulations in each scenario yields a distribution of AIC differences, analogous to the distribution of cross-validation error over multiple folds. Thresholds are a useful data summary for this analysis, which covers so many models and datasets. In investigating a specific domain model refinement on a real dataset, an AIC threshold is no substitute for a close analysis. Because most domain models are laboriously produced by experienced educators and domain experts, it is prudent to be conservative in testing proposed changes through data mining.

4. RESULTS

Our interest is in the relative difficulty of untangling different learning curve shapes under various interleaving regimes. Under what conditions are predictive models least sensitive to domain model improvement? Are there salient differences among the models?

For each of the 75 scenarios generated from the two-skill model, and the 15 scenarios generated from the one-skill model, there is a distribution of differences in AIC scores across the 500 replications of that scenario. In fact the value of having so many replications is that we can see this distribution of possible results.

The left side of Figure 4 shows the distributions for the two-skill generative domain model when practice on two good skills is interleaved gradually. From this distribution, we can see that AFM, PFA and R-PFA all select the correct two-skill in almost all of the 500 replications. AFM's margin for AIC differences is smaller than that of PFA and R-PFA, but even AFM's margin is greater than even the conservative AIC threshold $a = 10$.

An alternative visualization summarizes the full distribution with respect to a specific AIC difference threshold. The top horizontal line ("gradual") of the right side of Figure 4 shows the

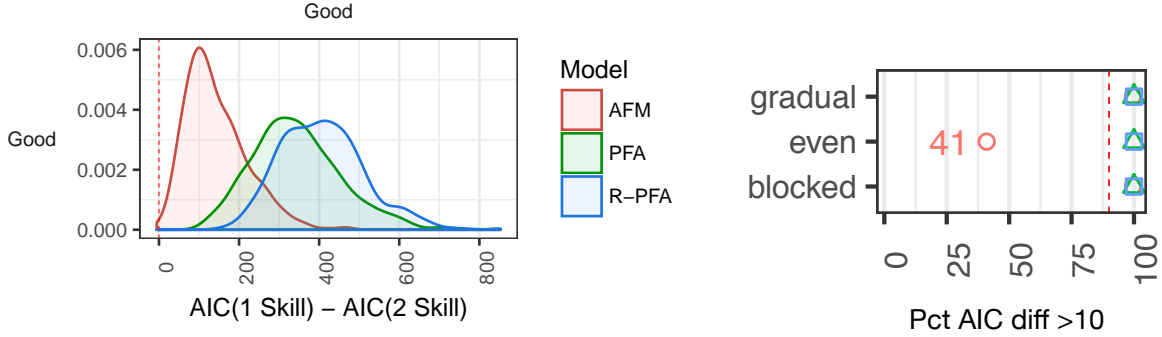


Figure 4: Left: Density of AIC differences for a Good curve Gradually interleaved with another Good curve. Right: Percent of simulations where the 2 skill domain model was correctly selected in scenarios where two good curves are interleaved.

percent of times that each predictive model correctly selected the two-skill model with an AIC threshold of 10. Even though AFM has a much smaller decision margin than PFA or R-PFA, in this gradual interleaving scenario AFM still makes the correct decision almost 100% of the time. However, the middle row indicates that when two good curves are evenly interleaved, AFM correctly selects the two-skill model only 41% of the time.

Using this concise summary visualization, we first discuss the two-skill generative model, and then the one-skill generative model.

4.1. RESULTS FOR THE GENERATIVE MODEL WITH 2 SKILLS

There are 75 scenarios where the data was generated with a two-skill model. When the results are aggregated across all 75 scenarios, the predictive models perform fairly well at correctly selecting the two-skill domain model (Table 2). The three AIC thresholds, from the relatively sensitive $a = 3$ to the relatively conservative $a = 10$, affect marginal accuracy rates only slightly, ranging from a drop of 0.22% in the marginal performance of R-PFA (from $a = 3$ to $a = 10$) to a drop of 3.39% in the marginal performance of AFM. Among all the margins, the lowest accuracy rate is 89% for AFM averaged across all simulation scenarios. The three types of interleaving present grossly similar levels of challenge to the predictive models, ranging from 92.04% to 98.34%. Similarly, the different shapes of the first learning curve or the second learning curve all lead to accuracy rates of 93% or higher.

However, these aggregate results mask important differences among models, among interleaving methods, and among curve shapes. To present a more detailed analysis, we collect figures like those in Figure 4 into small multiples in Figure 5 with $a = 3$ (top) and $a = 10$ (bottom); results for $a = 5$ are not shown, but similar. This display is essentially a four-way interaction plot, showing the effects of predictive model (redundantly indicated by point color and shape), the shape of the first curve (Y axis), the shape of the second curve (X axis), and mode of interleaving (subsidiary Y axis in each cell).

For example, the graph from Figure 4 is shown in the second row, second column of on the bottom of Figure 5. Similarly, the bottom left cell (in both the $a = 3$ and $a = 10$ figures) shows that for Slow Learning curves interleaved with Already Mastered curves, all three models have accuracy rates above 90% no matter the mode of interleaving. However, for Slow Learning curves interleaved with other Slow Learning curves (bottom right cell), only PFA and R-PFA

Table 2: Marginal accuracy rates in two-skill generative model scenarios under different AIC thresholds

Margin	Facet	$a = 3$	$a = 5$	$a = 10$
Interleaving	blocked	99.01	98.86	98.34
Interleaving	even	94.14	93.46	92.04
Interleaving	gradual	98.64	98.34	97.57
Model	AFM	92.38	91.38	88.99
Model	PFA	99.66	99.59	99.42
Model	R-PFA	99.74	99.69	99.53
First Curve	Already Mastered	97.36	97.00	96.32
First Curve	Good	98.58	98.39	98.04
First Curve	High Error	95.72	95.20	93.69
First Curve	Negligible Learning	97.25	96.78	95.62
First Curve	Slow Learning	97.40	97.07	96.24
Second Curve	Already Mastered	97.03	96.62	95.69
Second Curve	Good	98.53	98.36	98.03
Second Curve	High Error	95.87	95.28	93.88
Second Curve	Negligible Learning	97.16	96.72	95.59
Second Curve	Slow Learning	97.72	97.46	96.71

have accuracy rates above 90%, while AFM only correctly selects the two-skill domain model in $\sim 63\%$ when curves are evenly interleaved.

MODEL DIFFERENCES The simulations show that the PFA and R-PFA models are generally very sensitive in teasing apart inappropriately merged skills, and AFM is less sensitive. Under the $a = 3$ threshold for AIC differences (top figure), R-PFA has an accuracy rate above 90% in all of the simulation scenarios, and PFA in all but one scenario: High Error curves evenly interleaved with other High Error curves, where it is 79%. When we raise the threshold to $a = 10$, the accuracy rate for PFA in this scenario falls further to 65%, and the accuracy rate for R-PFA falls to 88%. By contrast, AFM often has low accuracy rates, including under $a = 3$ (16 of 75 simulation scenarios) and $a = 10$ (21 scenarios).

INTERLEAVING EFFECTS PFA and R-PFA handle all three interleaving types equally well. However, AFM’s two-skill accuracy rates differ dramatically based on the combination of curve shapes. For example, under the liberal threshold $a = 3$, for an Already Mastered curve combined with another Already Mastered curve, AFM’s accuracy ranges from 83% under blocked interleaving to 36% when the curves are evenly interleaved (i.e., worse than flipping a coin). At the conservative threshold $a = 10$ in the same scenario (two Already Mastered curves, evenly interleaved), AFM only recognized a correct split in 22% of the simulations.

For AFM, curves that are interleaved evenly are consistently more difficult to tease apart than gradually interleaved and blocked curves, and this is in large part driving the overall lower performance of AFM. At the $a = 3$ threshold, AFM struggles to correctly select the two-skill

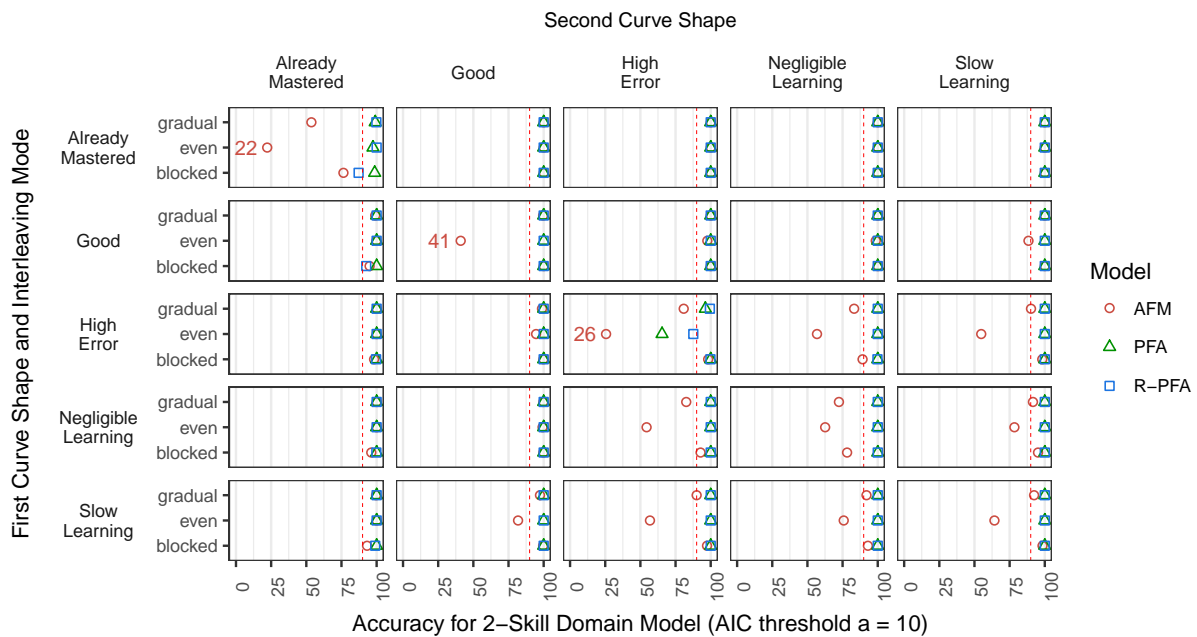
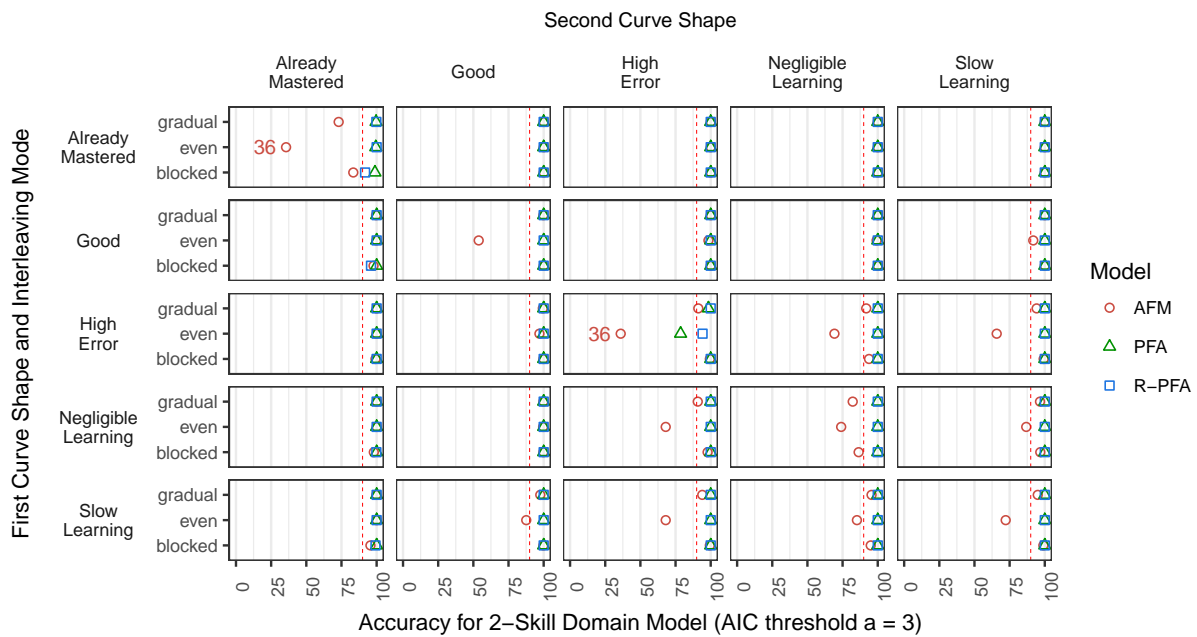


Figure 5: Predictive model accuracy in correctly selecting the two-skill model at sensitive (top) and conservative (bottom) AIC thresholds. The red reference line in each cell is 90% accuracy.

model with evenly interleaved curves in 12 of 25 curve shape pairs. Under gradual or blocked interleaving with the $a = 3$ threshold, AFM struggles with only 2 pairs of shapes. At $a = 10$, AFM has low accuracy rates in 13 of 25 evenly interleaved pairs, 5 gradually interleaved pairs, and 3 blocked pairs.

GENERATING CURVE SHAPE PFA and R-PFA are able to accurately discern a variety of curve shape pairs. PFA has a relatively low accuracy rate in only one simulation scenario, High Error curves evenly interleaved with other High Error curves. R-PFA is also challenged by the same scenario, but only at the conservative $a = 10$ threshold, and even then, it still correctly recognizes a split in 88% of the simulations.

If we consider the more conservative $a = 10$ threshold, AFM has more difficulty with the High Error, Negligible Learning and Slow Learning curves (below 90% in 15/27 scenarios), than with Good and Already mastered (below 90% in 4/12 scenarios).

SHAPE OF COMBINED SINGLE SKILL CURVE Visualization of sample curve shape pairs (Figures 8, 9, 10) illustrate the challenge of teasing them apart. AFM consistently had lower accuracy rates with Evenly interleaved curves (Figure 9). The curve pairs that challenge AFM the most produce relatively smooth curves when the skills are evenly interleaved. A Slow Learning skill interleaved with an already mastered skill produces a very jagged learning curve in the single-skill coding, and all 3 models chose the correct two skill model almost 100% of the time, at any threshold. In contrast, when two High-Error curves are interleaved, the resulting single-skill coding has a fairly smooth curve, and all 3 models had more difficulty selecting the correct 2 skill model. Specifically, AFM preferred the wrong model 74% of the time ($a = 10$). Other interleaving types can also produce somewhat smooth falsely merged curves, such as gradual or blocked interleaving of two Already Mastered curves; in these scenarios, AFM also has poor accuracy.

4.1.1. Sample Size

While the findings above show strong performance by PFA and R-PFA, there must necessarily be limitations to these simple logistic models. A full analysis of these limitations is a substantial project in its own right, but we took a first step: examining the question of how much data is necessary. It is often the case in real data, that some skills are practiced by fewer students than other skills. To model this, we re-ran the simulation with the number of simulated students per skill decreased from 100 to 20. Reducing the sample size highlights some of the differences among predictive models, interleaving modes and curve shapes that were not apparent with a larger sample size (Table 3, Figure 6).

PREDICTIVE MODELS (N=20) AFM accuracy drops with the sample size reduction, with aggregate accuracy rates ranging from 79% ($a = 3$) to 66.47% ($a = 10$). At $a = 10$, AFM's accuracy rate remains above 90% in only 23 of the 75 simulation scenarios.

PFA and R-PFA remain generally robust to the change in sample size, with aggregate accuracy rates above 92% even at $a = 10$. However, some curve combinations become challenging to both models.

In 53 of 75 scenarios, both PFA and R-PFA have accuracy rates of 97% or higher; in another 9 scenarios, both models' accuracy rates are 88% or higher. That leaves 13 scenarios which

Table 3: Marginal accuracy rates in two-skill generative model scenarios under different AIC thresholds; 20 students

Margin	Facet	$a = 3$	$a = 5$	$a = 10$
Interleaving	blocked	91.03	88.85	83.03
Interleaving	even	88.22	86.53	82.89
Interleaving	gradual	92.21	90.60	86.69
Model	AFM	78.77	75.02	66.47
Model	PFA	96.66	95.86	93.91
Model	R-PFA	96.03	95.10	92.22
First Curve	Already Mastered	92.10	90.89	88.22
First Curve	Good	94.29	93.22	90.08
First Curve	High Error	84.00	80.68	73.03
First Curve	Negligible Learning	90.62	88.84	84.33
First Curve	Slow Learning	91.42	89.65	85.34
Second Curve	Already Mastered	85.91	83.51	77.00
Second Curve	Good	95.11	93.99	90.96
Second Curve	High Error	86.56	83.99	78.16
Second Curve	Negligible Learning	91.66	90.13	86.62
Second Curve	Slow Learning	93.20	91.68	88.28

these two models find particularly challenging (Table 4), as discussed below.

INTERLEAVING MODES (N=20) Considered at an aggregated level, the interleaving regimes provide similar levels of challenge for a predictive model to detect a domain model improvement (Table 3). The relative order of difficulty is not affected by the reduced sample size: even interleaving is the most difficult, followed by gradual and blocked.

GENERATING CURVE SHAPE (N=20) The same cases which were difficult with 100 students only become more difficult with 20 students, as should be expected. Any combination of High Error, Negligible Learning or Slow Learning is challenging for AFM. Scenarios that have Already Mastered or Good as the second skill in particular induce lower accuracy rates than with the higher sample size.

SHAPE OF COMBINED SINGLE SKILL CURVE (N=20) Given how many curve pairs presented difficulty to AFM, we infer that the 23 scenarios in which AFM could distinguish the two skills must be somehow inherently easier cases. This includes the 12 scenarios where an Already Mastered curve is followed by any other curve shape. This is apparent in the consistently high accuracy rates in the Already Mastered row in Figure 6. Similarly, AFM can often tease apart a Good curve that is Blocked or Gradually interleaved with High Error, Negligible Learning or Slow Learning (Figure 6, row 2). These “inherently easy” scenarios produce a learning curve for the single-skill coding of the domain model that is quite jagged. For example, there’s a dramatic

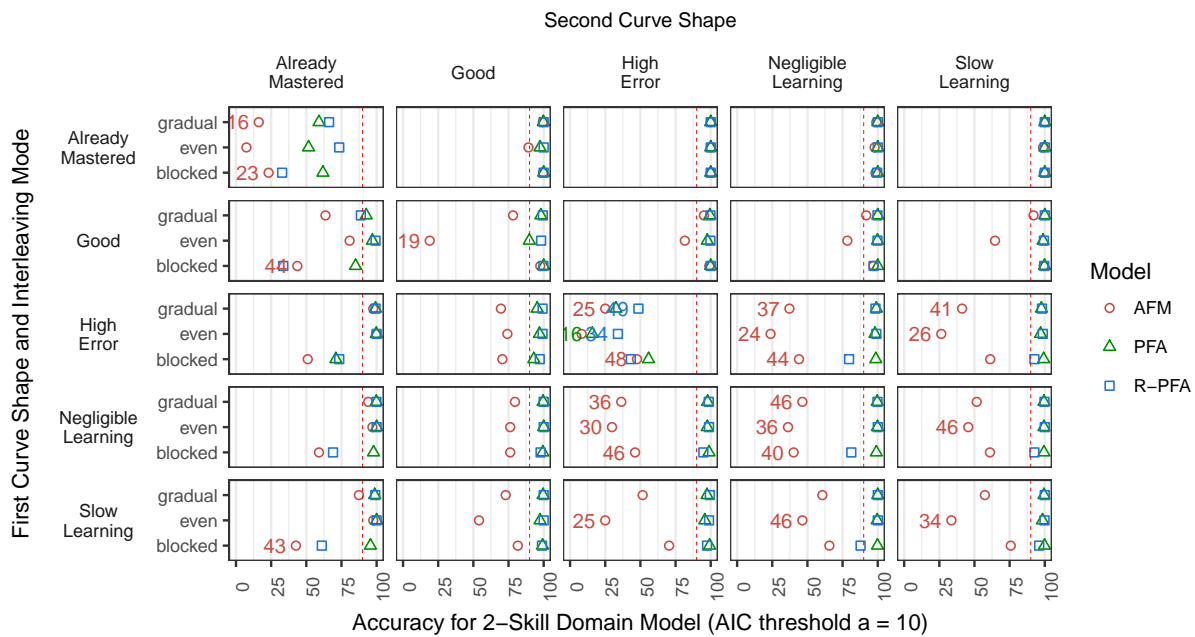
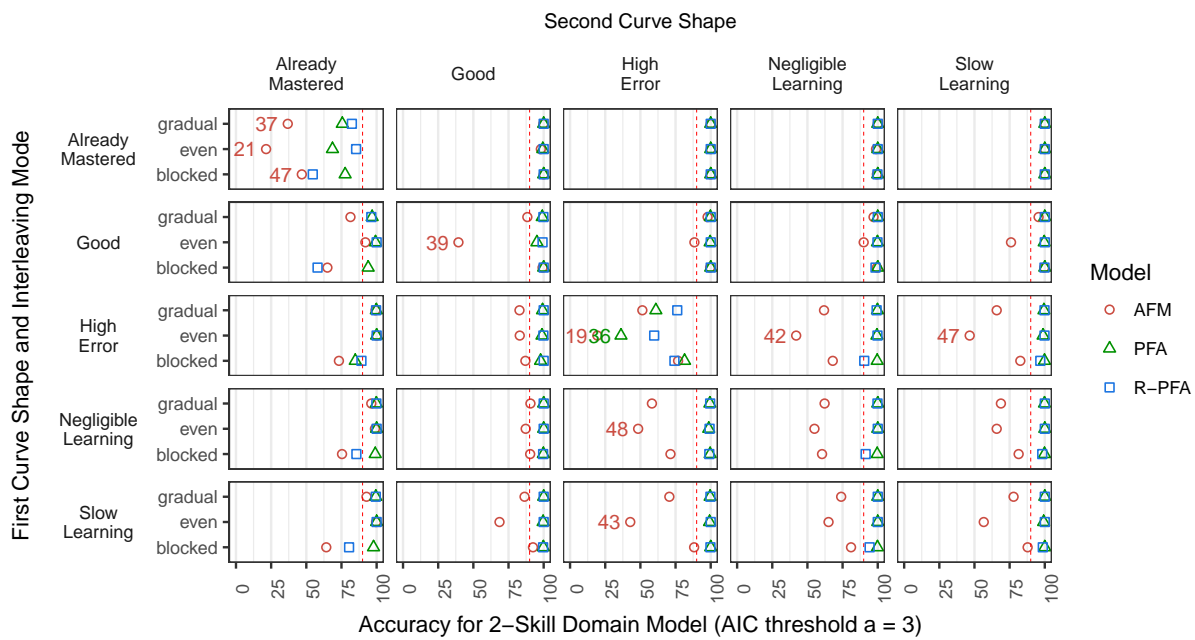


Figure 6: Predictive model accuracy in correctly selecting the two-skill model at sensitive (top) and conservative (bottom) AIC thresholds. The red reference line in each cell is 90% accuracy. 20 students.

drop in success rate when an Already Mastered curve is blocked with any other curve shape (Figure 8).

In the scenario when two good skills are evenly interleaved, AFM prefers the false one-skill domain model over the correct two-skill model 81% of the time. But when practice is blocked, AFM correctly detects the two skills with an accuracy well above 90%, even at this small sample size.

Revisiting a visual example of these curve pairs (Figure 2), it does seem plausible that interleaving can cause these differences in accuracy of recognizing a two-skill domain model. Under Blocked interleaving, the single-skill coding produces a learning curve with a dramatic spike, but in Even interleaving, the single-skill learning curve is relatively smooth. Under Gradual interleaving, this curve pairing may (probabilistically) still have dramatic spikes in success rate, but the accuracy nonetheless drops, depending on the sample size and AIC threshold.

Table 4: Accuracy rates, 20 students, scenarios in which PFA or R-PFA are below 90% accuracy.

Interleaving	First Curve	Second Curve	PFA $a = 10$	PFA $a = 3$	R-PFA $a = 10$	R-PFA $a = 3$
blocked	High Error	High Error	55.8	81.4	43.2	74.2
even	High Error	High Error	15.8	36.2	34.0	59.8
gradual	High Error	High Error	32.6	61.0	48.6	76.2
blocked	Already Mastered	Already Mastered	61.8	77.6	32.8	54.6
even	Already Mastered	Already Mastered	51.6	68.6	73.4	85.4
gradual	Already Mastered	Already Mastered	59.0	75.6	66.2	82.4
blocked	Good	Already Mastered	85.0	94.0	33.6	58.0
blocked	High Error	Already Mastered	71.0	84.8	73.4	89.2
blocked	High Error	Negligible Learning	98.6	99.6	79.6	90.4
blocked	Negligible Learning	Already Mastered	97.8	99.0	69.0	85.6
blocked	Negligible Learning	Negligible Learning	99.0	99.6	81.2	91.4
blocked	Slow Learning	Already Mastered	95.6	97.8	61.0	80.4
blocked	Slow Learning	Negligible Learning	99.8	99.8	87.8	94.2

We turn now to the 13 scenarios in which PFA and R-PFA had accuracy rates less than 88% (Table 4). Any mode of interleaving two High Error curves or two Already Mastered curves is severely challenging for both PFA and R-PFA (6 scenarios). In these scenarios, the single-skill

coding produces a relatively smooth learning curve, so that the incorrect single skill model is chosen as more predictive between 20% and 70% of the time, depending on threshold.

In the condition that the sample size is small and the AIC difference threshold is conservative, there is a pattern of differences between PFA and R-PFA. Specifically, they tend to differ when two curves are blocked, and the population success rates are somewhat close at the end of the first curve and the beginning of the second. For example, when a Good curve is gradually interleaved with an Already Mastered curve, both PFA and R-PFA have accuracy rates near 90%. However, when Good is blocked with an Already Mastered curve, PFA accuracy drops only slightly, whereas R-PFA's accuracy drops precipitously. There are 7 similar scenarios (bottom section of Table 4). In each of these 7 scenarios, the learning curve for the single-skill coding has a relatively small spike in the shape of the curve (Figure 8).

It is likely that the cause of this difference in R-PFA's sensitivity is that R-PFA readily accommodates the spike (or trough) as noise, and the influence of the noise on prediction fades as the model gleans additional observations. This is precisely the purpose of R , the proportion of correct attempts in the R-PFA model, and the decay factor applied to R . Given a small sample of 20 students, there is insufficient data for R-PFA to learn a skill parameter that reflects the spike (or trough) at the join of the curves. Both PFA and R-PFA improve in accuracy rate at the more liberal AIC threshold $a = 3$, but more importantly, both models attain very high true accuracy even at $a = 10$ when the sample size is sufficiently large.

4.2. RESULTS FOR THE GENERATIVE MODEL WITH 1 SKILL

In this set of 15 scenarios, we test the accuracy of the predictive models in recognizing the generative (correct) one-skill model rather than an alternative domain model that improperly splits that skill into two skills. This set of scenarios would be analogous to the example where forwards and backwards computations with simple geometric figures were proposed to be two skills, but found to be a single skill (Koedinger et al., 2012).

Aggregate accuracy rates are generally lower in this case than in the two-skill generative case, implying that the detection of a single skill is a relatively more difficult task than detecting two distinct skills (Table 5). We measure sensitivity at a range of AIC thresholds, from $a = 3$ to $a = 10$. In this set of scenarios, as the AIC difference threshold increases, the models become more likely to choose the one-skill model, because the two-skill model must meet the higher, more conservative, threshold to be considered a better model. There are substantial differences across all facets: predictive models, modes of interleaving, and true curve shapes.

The small-multiples plot (Figure 7) shows accuracy rate (x-axis) across the three-way interaction of true curve shape (secondary x-axis), interleaving mode (y-axis), and predictive model (point color and shape). At either threshold, and in all facets of the interaction, R-PFA is consistently the most accurate model for the single-skill model. PFA and AFM vie for second place, depending on the interaction.

PREDICTIVE MODELS AFM and PFA both perform poorly, with the highest aggregate accuracy rates at 81% and 72%, respectively. Both models struggle in all 5 blocked scenarios, though there is an interaction discussed below. On true Negligible Learning curves, performance is borderline (AFM has accuracy of 89% with $a = 3$); however, in the High Error and Good cases, AFM prefers the incorrect two-skill coding in more than 75% of replications.

R-PFA performs well, with aggregate accuracy rate above 95% for all thresholds. When

Table 5: Marginal accuracy rates for one-skill generative model under different AIC thresholds

Margin	Facet	$a = 3$	$a = 5$	$a = 10$
Interleaving	blocked	54.11	56.35	61.87
Interleaving	even	98.93	99.41	99.84
Interleaving	gradual	82.39	84.05	87.71
Model	AFM	73.99	76.01	80.71
Model	PFA	66.72	68.29	71.63
Model	R-PFA	94.72	95.51	97.08
True Curve	Already Mastered	91.16	92.51	95.29
True Curve	Good	66.04	67.11	70.24
True Curve	High Error	55.69	59.13	66.82
True Curve	Negligible Learning	94.71	95.13	96.00
True Curve	Slow Learning	84.78	85.80	87.33

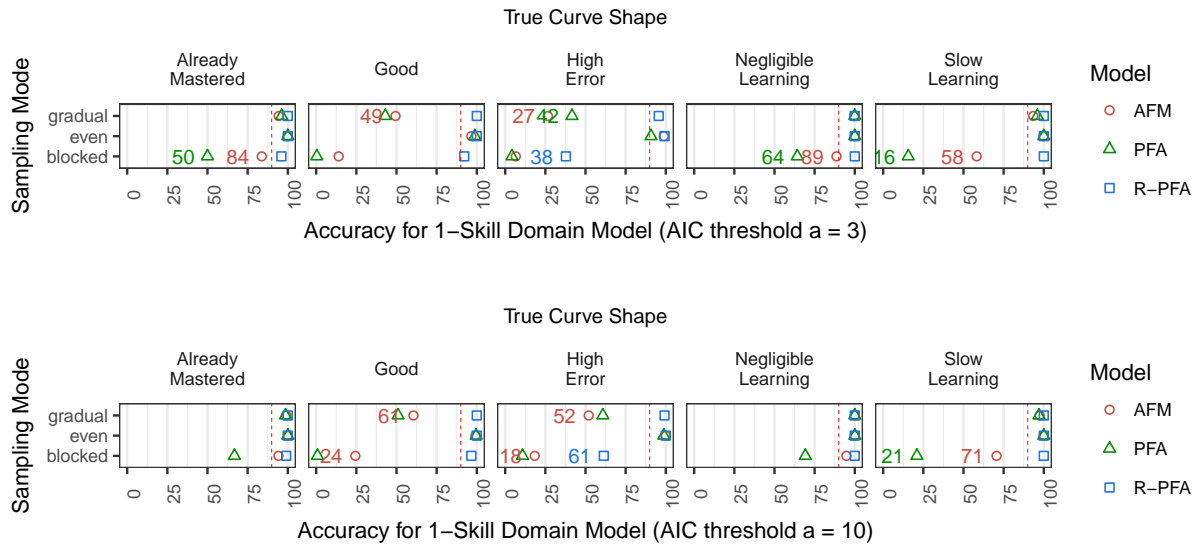


Figure 7: Predictive model accuracy for detecting the single-skill domain model.

the generative curve is Already Mastered, Negligible Learning or Slow Learning, R-PFA has almost-perfect accuracy rates across all interleaving modes. For Good curves, the accuracy is still above 90%. It is only in the case where a High Error curve is split into two-curves using blocked interleaving that R struggles.

INTERLEAVING There are strong effects of interleaving regime. When the a one-skill curve is split using even interleaving, the predictive models are generally accurate at recognizing the one-skill model (accuracy above 82%). Accuracy decreases under gradual interleaving (above 82%), and under blocked interleaving, accuracy is little better than a coin flip (between 54% and 62%). This is the opposite of the two-skill generative model, where accuracy is lowest with even interleaving and higher with gradual or blocked interleaving.

GENERATING CURVE SHAPE The generative curve shape affects the accuracy rate dramatically. The Already Mastered and Negligible Learning generative curves can be recognized with high accuracy (above 91% and 95%, respectively). These two curves are similar to straight lines with relatively flat slopes, and have the least curvature out of all 5 curve types; and no matter the interleaving regime, the two-skill coding produces learning curves that are parallel, if not almost identical (Figure 3). Thus, it makes sense that a predictive model could easily recognize the one-skill model.

In contrast, the Good and High Error generative curves have aggregate accuracy rates below 70% and 67%, respectively, with the exception that in the even interleaving case, all three predictive models have accuracy above 90%. It is under the gradual and blocked interleaving that the predictive models tend to prefer the two-skill coding. For example, the accuracy of PFA is near zero in recognizing a generative Good curve when it is split with blocked interleaving. Both of Good and High Error curves contain relatively sharp elbows; one explanation for low accuracy in these scenarios is that when practice is blocked, a predictive model may well prefer to treat the curve to the left of the elbow as distinct from the part after the elbow.

5. DISCUSSION

5.1. VALIDITY CHECKS

The findings satisfy some basic plausibility criteria. First, AIC thresholding worked as intended (Figure 4), producing a useful data summary.

Second, although predictive accuracy of student performance is not the object of this study, it is important that the predictive model rankings are consistent with prior work. Specifically, R-PFA had the highest predictive accuracy in 100% of the simulations in each interleaving scenario in both the one-skill and two-skill generating conditions. This is only a new result insofar as this comparison omits a student parameter in all the models while a prior comparison on BKT-simulated data included the student parameter (Galyardt and Goldin, 2015), but a different result would have raised a red flag.

A third plausibility check pertains to overfitting, i.e., to the potential for a predictive modeling procedure to select models that are unnecessarily complex. Here, predictive models may overfit the data if they tend to exhibit a bias toward a more complex two-skill model. PFA and R-PFA had high accuracy rates in correctly recognizing the two-skill generating model, so we might worry that these models were biased to selecting the more complex model; but this is

not the case. R-PFA also had very high accuracy in the one-skill conditions, and so we can be confident that it is not overfitting the data. Overfitting does remain a concern for PFA, which shows a strong bias towards a two-skill model in some scenarios generated from the one-skill model. AFM definitively suffers from overfit, but that is arguably not even the primary problem for AFM.

5.2. SPECIFIC FINDINGS

In light of these validity checks, the findings are as follows.

In general, data mining techniques can accurately discern one-skill vs two-skill domain models for a variety of curve shapes and combinations. However, there are important caveats to this result.

Overall, recognizing a one-skill domain model is more challenging than recognizing a two-skill model. This is notable under gradual interleaving, which we suspect might be the most prevalent kind of overlap in real datasets. Even at the threshold that puts the highest burden on the more complex model ($a = 10$), the aggregate accuracy rate under gradual interleaving is 88% for the one-skill domain model and 98% for the two-skill domain model. Nonetheless, R-PFA's accuracy in the one-skill case is far higher than the other models', including above 90% in all gradually interleaved cases, so another perspective is that recognizing a single-skill model is particularly challenging when one uses an inappropriate predictive model.

Accuracy in detecting two-skill generative model drops when the dataset size is small. This is to be expected; with less data, there is less evidence that a more complex model is needed. Further study is needed to fully understand the power calculations for these comparisons.

Under blocked interleaving, PFA and AFM tended to select a two-skilled model instead of a one-skilled, whereas under even interleaving, this tendency was reversed for AFM. Although it is unlikely that deployed tutoring systems would have purely blocked or purely even interleaving, these interleaving types are corner cases on a continuum, and it is possible to imagine "almost blocked" or "almost even" interleaving in some systems. This concern should bear on one's choice of a predictive model.

Some curve combinations are more challenging with others. As suspected, two curves that have the same basic shape can be hard to tease apart when they are evenly interleaved, e.g., High Error with High Error.

Choice of predictive model directly affects accuracy rates. Of the models studied here, AFM often underperforms relative to PFA and R-PFA, and never outperforms them.

These challenges are cumulative; under some conditions, the correct model may be very hard to detect. For example, for two Good curves gradually interleaved with each other in a data set of 20 students, AFM achieves an accuracy rate of 19%.

5.3. IMPLICATIONS FOR TESTING DOMAIN MODEL IMPROVEMENTS

When designing a test of domain model improvements, one can choose the predictive model and the threshold that determines whether a difference in predictive accuracy is meaningful. The shape of the curves is due to the nature of the material and is beyond our control (until we can refine the domain model). The mode of interleaving, we cannot control with already collected data, but it would be possible to structure this in the collection of new data. The shape of the curves and the mode of interleaving affect the difficulty of recognizing a posited improvement; the predictive model and the decision threshold determine the sensitivity of the test.

For example, in a tutoring context, increasing the number of skills in the domain model may entail an increase in the amount practice expected of all students; in this case, we may wish to require very strong evidence that a two-skill model truly is an improvement and set a higher threshold. Conversely, consider targeted remediation, which focuses an individual's skill practice on just the skills where that student is struggling, and all students are not expected to practice all skills. In this use case, moderate evidence that there are two skills may be sufficient, and a lower threshold would be appropriate. This decision is exactly analogous to the decision of whether a p-value of 0.01 is required, or 0.05 is sufficient.

The levels of noise in a dataset may also affect choice of test sensitivity. Real data are noisier than the simulated data used here because of item properties (e.g., user interface issues or confusing language in a word problem), because of insufficient practice per skill or per student, and because of non-random missingness due to student graduation from a skill, because of behaviors such as slips, guesses, gaming the system, WTF behaviors (Wixon et al., 2012), and because of inaccurate item-skill mapping (e.g., mapping at a problem level rather than a problem-step level). Real data would be noisier both for the skill curves under investigation (i.e., those that represent the domain model improvement) and for the remainder of the dataset that might be used to estimate parameters for other skills and for the students. High prevalence of such noise may call for a predictive model that is robust to noise.

Model choice affects test sensitivity; a model that is robust to noise may be an asset with real-world data, or a detriment if it glosses over the junction of two skills. R-PFA was generally more sensitive to both the one-skill and two-skill domain models than either PFA or AFM, but there were cases in the smaller sample size where R-PFA could not discriminate between a noisy learning curve and an incorrect domain model.

Given poor sensitivity from AFM, how should one choose a predictive model to test domain model improvement, whether PFA, R-PFA, or some alternative? The findings indicate that it is critical to evaluate the predictive model under a variety of interleaving regimes and curve shapes. The 90 scenarios simulated here cover many likely data contingencies, but a specific use case might call for other scenarios.

As additional predictive models are published, the simulation framework proposed here should inform model evaluation for testing domain model refinements.

5.4. LIMITATIONS AND FUTURE WORK

The methodology of this analysis has some inherent limitations. Can the results be generalized to real-world settings?

The primary contribution of this analysis is to call out the complexity imposed by varied curve shapes and interleaving regimes. Further, it is a strength of data simulation to define the truth by construction (whereas it may never be knowable in real student data), and to exhaustively explore data interactions that may escape our attention in an arbitrary real-world data set.

One limitation, however, is that this analysis treats all learning curve shapes and combinations as equally likely and important. In the context of a specific educational application, some curve combinations may be unlikely, and domain model refinement may be more important for some skills than others. Another limitation is that simulated data contain a knowable and controllable amount of simulated noise. In this light, the accuracy rates estimated here may reflect best-case performance by the predictive models.

The specific simulation technique used here has limitations of its own. The sequences of correct and incorrect item responses emitted by Bayesian Knowledge Tracing may differ from the sequences that human students would emit. BKT is naive in multiple ways, among them that it treats all students as having the same ability and learning at the same rate; it does not consider that skills may be unexpectedly difficult for some students, or that students may attempt to game the system; it also assumes that students never forget a skill. However, some of these issues mainly affect BKT as it is popularly used for prediction, rather than, as in this work, for data generation. Nonetheless, future work should consider alternative data-generation methods.

There are many possible alternative simulation approaches, which may be superficially appealing in that they purport to represent varying real-world phenomena. However, such variety is not necessarily a relevant contribution to our analysis. For instance, one could generate student data using a distribution of learning curves, as in (Fancsali et al., 2013) section 4. This would serve to add further noise to the data without changing the mean curve. The additional noise would degrade the performance of AFM and PFA, but R-PFA is robust to student differences, and we would not expect the performance of R-PFA to be substantially degraded (Galyardt and Goldin, 2015). Further, richer simulation techniques entail a concomitant increase in mathematical complexity and run-time computational demands.

Alternatively, data could have been easily generated using an AFM, PFA, or R-PFA models (or all three), but testing a predictive model against data generated from that model gives it an unfair advantage that it would never have in real data.

Given a history of validated uses for predictive models in education, and given the theoretical justification for models that incorporate a learning curve in particular, the Linear Logistic Test Models in this analysis are an attractive and useful tool. Nonetheless, predictive models, by definition, do not reflect true complexity of human cognition, and different model structures may produce parameters that do or do not correlate with each other. (Liu and Koedinger, 2017; Käser et al., 2014) Future work should extend this analysis to predictive models of other families.

Future work should also tackle additional kinds of complexity in the domain model itself, e.g., compensatory and non-compensatory relationships and condensation functions.

6. CONCLUSIONS

It is possible to improve domain models by splitting or merging their constituent skills. However, the validity of proposed model refinements cannot be taken for granted. Predictive models provide a way to validate a proposed split or merge. In many scenarios, it is possible to achieve a rate of almost 100% in recognizing when two distinct skills are improperly merged together, or when one coherent skill is unnecessarily split into two. However, the findings show that some curve pairs are more challenging to tease apart, that correctly recognizing single-skill domain model improvements may be particularly challenging, and that accuracy can be affected by choice of predictive model, sample size, and comparison threshold.

To arrive at these conclusions, we defined a framework to compare predictive models in a variety of scenarios. The framework considers that a worthwhile domain model improvement may be falsely rejected, or an incorrect proposed change may be falsely accepted. It further considers a set of 5 typical shapes of learning curves, and three different ways of interleaving the curves. This resulted in 90 scenarios to evaluate the accuracy of recognizing domain model improvements. The framework further shows how to compute accuracy rate as a descriptor of predictive model sensitivity to domain model improvement. Finally, this paper illustrates how

to apply the framework by measuring sensitivity to domain model improvement for 3 predictive models on data simulated in each of the distinct 90 scenarios.

As one example of this analysis, the expected accuracy rate to recognize a gradual interleaving of an Already Mastered curve with a Good curve is 100% even for AFM, the least-sensitive model. This curve combination matches the apocryphal case of the distinct skills for the first parameter and the subsequent parameters to a Lisp function (Corbett and Anderson, 1995). However, because some curve combinations are much more challenging to tease apart, historical attempts to test domain model refinements may need to be reexamined. For example, if some well-justified difficulty factor did not pass an AFM-based test, it may be helpful to consider whether its constituent curve shapes and method of interleaving would tend to challenge AFM. This concern applies both to one-off tests of domain model refinements, as well as to automated search in LFA.

The curves used in this paper represent an idealized, broad-brush coverage of the space of possibilities. The primary purpose of this study is not to be exhaustive, but to create an awareness that some scenarios present unexpected challenges to recognizing a domain model improvement, and to lay out some of the factors that affect that accuracy. Future research should define clear conditions, derived from statistical theory, that provide complete guidelines to which conditions are difficult and which would have high accuracy; such effort would have higher dividends than attempting to design an exhaustive set of simulation conditions.

Accurate formative assessment, efficient instruction and a number of other uses require a domain model that is valid and that captures fine-grained distinctions among skills targeted for instruction and assessment. Domain modeling is one of the main contributions of the learning sciences to the enterprise of education, and learning curve analysis can be an excellent tool to refine domain models.

Aside from the comparison of the three predictive models considered here, this approach is applicable in any setting where a domain model might be considered for improvement based on observational data. The criteria for applicability are that the domain can be described in terms of skills, the skills can be described by learning curves, and skill knowledge transfers across multiple tasks that constitute assessments.

ACKNOWLEDGMENTS

[Distribution Statement A] Approved for public release and unlimited distribution.

REFERENCES

- ANDERSON, J. R. 2013. Discovering the structure of mathematical problem solving. In *Proceedings of the 6th International Conference on Educational Data Mining (EDM 2013)*, S. K. D’Mello, R. A. Calvo, and A. Olney, Eds. 2.
- BAKER, R. S. J. D., CORBETT, A. T., AND KOEDINGER, K. R. 2007. The Difficulty Factors Approach to the Design of Lessons in Intelligent Tutor Curricula. *International Journal of Artificial Intelligence and Education* 17, 4 (Dec.), 341–369.
- CEN, H., KOEDINGER, K., AND JUNKER, B. 2006. Learning Factors Analysis – A General Method for Cognitive Model Evaluation and Improvement. M. Ikeda, K. D. Ashley, and T.-W. Chan, Eds. Vol. 4053. Springer Berlin Heidelberg, Jhongli, Taiwan, 164–175.

- CEN, H., KOEDINGER, K. R., AND JUNKER, B. W. 2007. Is over practice necessary? – Improving learning efficiency with the Cognitive Tutor through educational data mining. In *Proceedings of the 2007 Conference on Artificial Intelligence in Education: Building Technology Rich Learning Contexts That Work*. IOS Press, Amsterdam, The Netherlands, The Netherlands, 511 – 518.
- CHEN, Y., GONZÁLEZ-BRENES, J. P., AND TIAN, J. 2016. Joint Discovery of Skill Prerequisite Graphs and Student Models. In *Proceedings of the 9th International Conference on Educational Data Mining*. International Educational Data Mining Society.
- CORBETT, A. T. AND ANDERSON, J. R. 1995. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction* 4, 4, 253–278.
- DATASHOP TEAM. 2016. Learning Curve. <https://pslcdatashop.web.cmu.edu/help?page=learningCurve>.
- DE BOECK, P. AND WILSON, M., Eds. 2004. *Explanatory Item Response Models: A Generalized Linear and Nonlinear Approach*. Statistics for social science and public policy. Springer, New York.
- DE LA TORRE, J. AND CHIU, C.-Y. 2016. A General Method of Empirical Q-matrix Validation. *Psychometrika* 81, 2 (June), 253–273.
- DESMARAIS, M. C. 2012. Mapping Question Items to Skills with Non-negative Matrix Factorization. *SIGKDD Explor. Newsl.* 13, 2 (May), 30–36.
- FALMAGNE, J.-C., KOPPEN, M., VILLANO, M., DOIGNON, J.-P., AND JOHANNESSEN, L. 1990. Introduction to knowledge spaces: How to build, test, and search them. *Psychological Review* 97, 2, 201.
- FANCSALI, S. E., NIXON, T., AND RITTER, S. 2013. Optimal and worst-case performance of mastery learning assessment with bayesian knowledge tracing. In *Proceedings of the 6th International Conference on Educational Data Mining (EDM 2013)*, S. K. D’Mello, R. A. Calvo, and A. Olney, Eds. 35–42.
- FISCHER, G. H. 1973. The linear logistic test model as an instrument in educational research. *Acta Psychologica* 37, 6 (Dec.), 359–374.
- GALYARDT, A. 2012. *Mixed Membership Distributions with Applications to Modeling Multiple Strategy Usage*. Carnegie Mellon University, PhD Dissertation, Pittsburgh, PA 15213.
- GALYARDT, A. AND GOLDIN, I. M. 2015. Move your lamp post: Recent data reflects learner knowledge better than older data. *Journal of Educational Data Mining* 7, 2, 83–108.
- GELMAN, A. AND HILL, J. 2006. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press.
- GOLDIN, I. M. AND GALYARDT, A. 2015. Convergent Validity of a Student Model: Recent-Performance Factors Analysis. In *Proceedings of the 8th International Conference on Educational Data Mining*. Madrid, Spain.
- GOLDIN, I. M., PAVLIK JR, P., AND RITTER, S. 2016. Discovering Domain Models in Learning Curve Data. In *Design Recommendations for Intelligent Tutoring Systems: Domain Modeling*. Vol. 4. US Army Research Laboratory.
- GONZALEZ-BRENES, J. AND HUANG, Y. 2015. Your model is predictive– but is it useful? theoretical and empirical considerations of a new paradigm for adaptive tutoring evaluation. In *The 8th International Conference on Educational Data Mining*, C. Romero and M. Pechenizkiy, Eds. *Proceedings of the 8th International Conference on Educational Data Mining*.
- GONZÁLEZ-BRENES, J. P. AND MOSTOW, J. 2013. What and When do Students Learn? Fully Data-Driven Joint Estimation of Cognitive and Student Models. In *Proceedings of 6th International Conference on Educational Data Mining*. Memphis, TN.

- HASTIE, T., TIBSHIRANI, R., AND FRIEDMAN, J. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd Edition ed. Springer Series in Statistics. Springer, New York, NY.
- KÄSER, T., KLINGLER, S., AND GROSS, M. 2016. When to Stop?: Towards Universal Instructional Policies. In *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge*. LAK '16. ACM, New York, NY, USA, 289–298.
- KÄSER, T., KOEDINGER, K., AND GROSS, M. 2014. Different parameters-same prediction: An analysis of learning curves. In *Proceedings of the 7th International Conference on Educational Data Mining*.
- KOEDINGER, K. R. AND MCLAUGHLIN, E. A. 2016. Closing the Loop with Quantitative Cognitive Task Analysis. In *Proceedings of the 9th International Conference on Educational Data Mining*. Raleigh, NC, USA.
- KOEDINGER, K. R., MCLAUGHLIN, E. A., AND STAMPER, J. C. 2012. Automated student model improvement. In *Proceedings of 5th International Conference on Educational Data Mining*. Chania, Greece.
- KOEDINGER, K. R., STAMPER, J. C., MCLAUGHLIN, E. A., AND NIXON, T. 2013. Using data-driven discovery of better student models to improve student learning. In *Artificial Intelligence in Education*, H. C. Lane, K. Yacef, J. Mostow, and P. Pavlik, Eds. Number 7926 in Lecture Notes in Computer Science. Springer Berlin Heidelberg, 421–430.
- KOEDINGER, K. R., YUDELSON, M. V., AND PAVLIK, P. I. 2016. Testing theories of transfer using error rate learning curves. *Topics in Cognitive Science* 8, 3 (July), 589–609.
- LEE, J. I. AND BRUNSKILL, E. 2012. *The Impact on Individualizing Student Models on Necessary Practice Opportunities*. International Educational Data Mining Society.
- LIU, J., XU, G., AND YING, Z. 2012. Data-Driven Learning of Q-Matrix. *Applied Psychological Measurement* 36, 7 (Oct.), 548–564.
- LIU, R. AND KOEDINGER, K. 2017. Towards reliable and valid measurement of individualized student parameters. In *Proceedings of the 10th International Conference on Educational Data Mining*. Wuhan, China.
- MARTIN, B., MITROVIC, A., KOEDINGER, K. R., AND MATHAN, S. 2011. Evaluating and improving adaptive educational systems with learning curves. *User Modeling and User-Adapted Interaction* 21, 3, 249–283.
- MURRAY, R. C., RITTER, S., NIXON, T., SCHWIEBERT, R., HAUSMANN, R. G. M., TOWLE, B., FANCSALI, S. E., AND VUONG, A. 2013. Revealing the Learning in Learning Curves. In *Artificial Intelligence in Education*, H. C. Lane, K. Yacef, J. Mostow, and P. Pavlik, Eds. Lecture Notes in Computer Science, vol. 7926. Springer, Heidelberg, 473–482.
- PAVLIK JR, P., CEN, H., AND KOEDINGER, K. 2009. Performance Factors Analysis - A New Alternative to Knowledge Tracing. In *Proceedings of 14th International Conference on Artificial Intelligence in Education*, V. Dimitrova and R. Mizoguchi, Eds. Brighton, England, 531–538.
- ROLLINSON, J. AND BRUNSKILL, E. 2015. From Predictive Models to Instructional Policies. International Educational Data Mining Society.
- RUPP, A. A. AND TEMPLIN, J. L. 2008. Unique Characteristics of Diagnostic Classification Models: A Comprehensive Review of the Current State-of-the-Art. *Measurement: Interdisciplinary Research & Perspective* 6, 4 (Nov.), 219–262.
- SCHEINES, R., SILVER, E., AND GOLDIN, I. M. 2014. Discovering Prerequisite Relationships among Knowledge Components. In *Proceedings of 7th International Conference on Educational Data Mining*. London, UK.

- STAMPER, J., KOEDINGER, K., AND MCLAUGHLIN, E. 2013. A comparison of model selection metrics in Datashop. In *Proceedings of 6th International Conference on Educational Data Mining*.
- STONE, M. 1977. An asymptotic equivalence of choice of model by cross-validation and Akaike's criterion. *Journal of the Royal Statistical Society. Series B (Methodological)* 39, 1, 44–47.
- SULLIVAN, M. E., YATES, K. A., INABA, K., LAM, L., AND CLARK, R. E. 2014. The Use of Cognitive Task Analysis to Reveal the Instructional Limitations of Experts in the Teaching of Procedural Skills. *Academic Medicine* 89, 5 (May), 811–816.
- TATSUOKA, K. K. 1983. Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of educational measurement* 20, 4, 345–354.
- VUONG, A., NIXON, T., AND TOWLE, B. 2011. A Method for Finding Prerequisites Within a Curriculum. In *Proceedings of 4th International Conference on Educational Data Mining*. 211–216.
- WIXON, M., D BAKER, R. S. J., GOBERT, J. D., OCUMPAUGH, J., AND BACHMANN, M. 2012. WTF? Detecting Students Who Are Conducting Inquiry Without Thinking Fastidiously. In *User Modeling, Adaptation, and Personalization*, J. Masthoff, B. Mobasher, M. C. Desmarais, and R. Nkambou, Eds. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 286–296.
- YUE, G., E, B. J., AND T, H. N. 2011. How to construct more accurate student models: Comparing and optimizing knowledge tracing and performance factor analysis. *International Journal of Artificial Intelligence in Education* 1-2, 27–46.

7. APPENDIX A: INTERLEAVING SCENARIOS

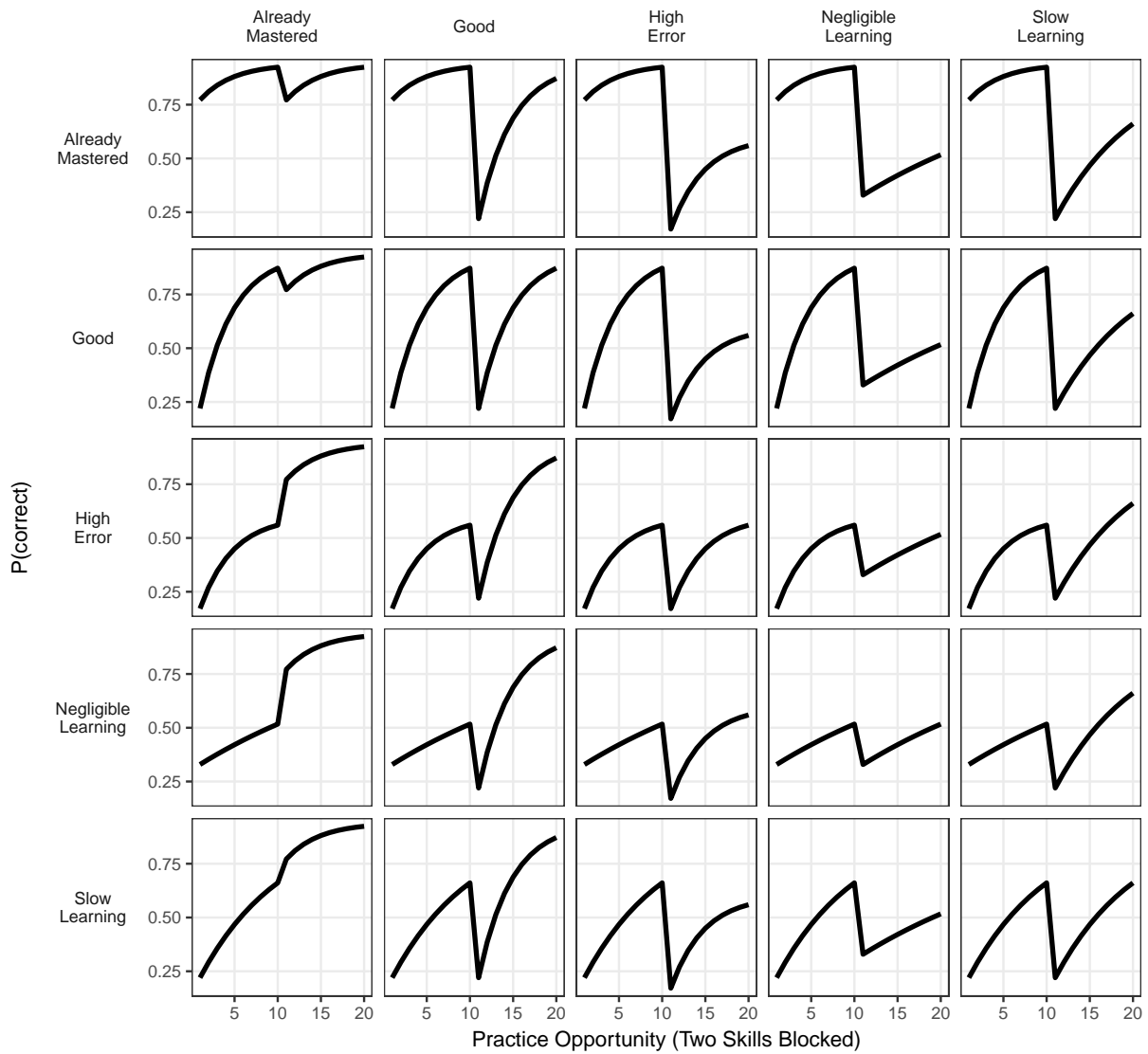


Figure 8: True learning curves when practice on two skills is perfectly blocked

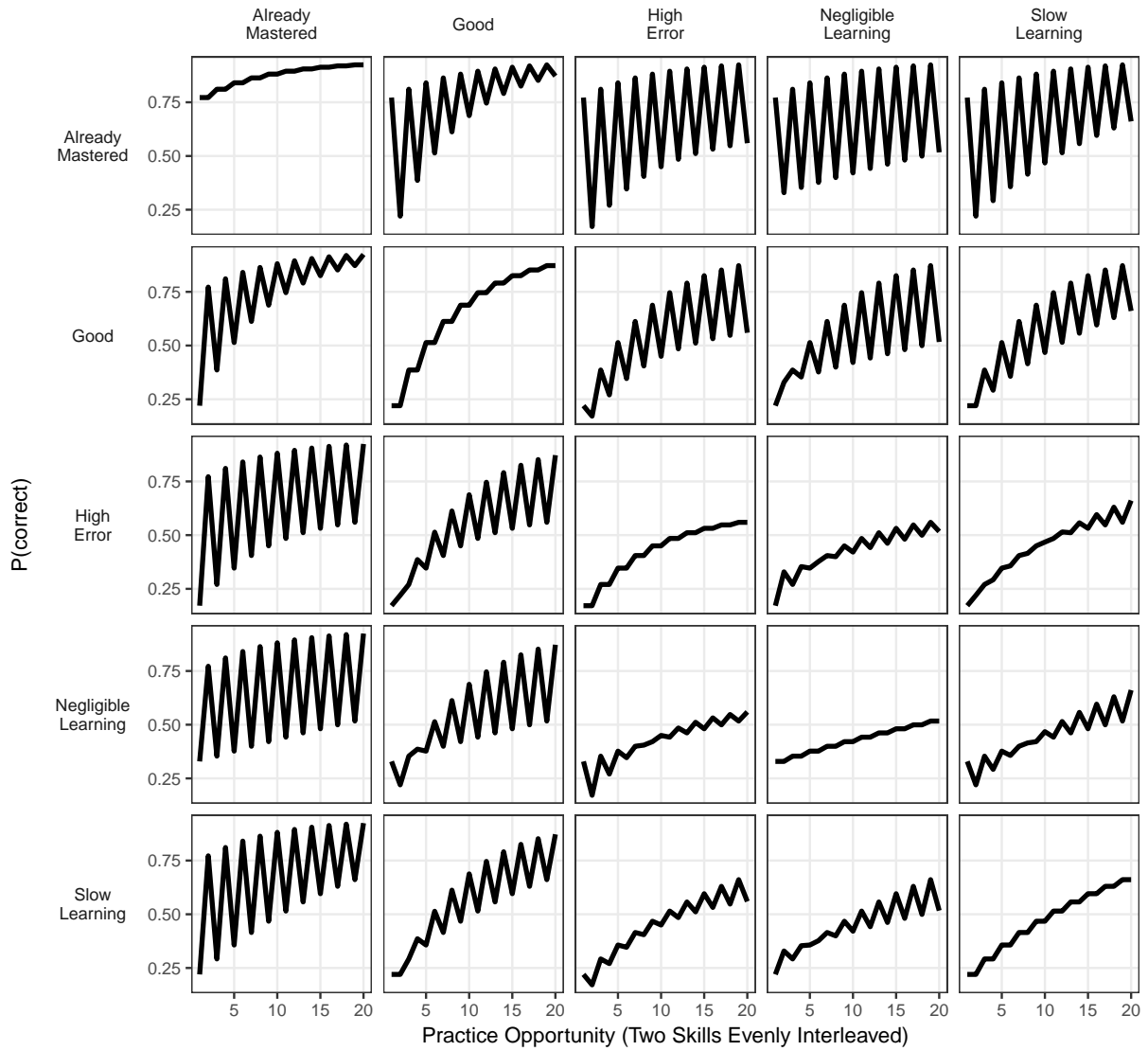


Figure 9: True learning curves when practice on two skills is evenly interleaved

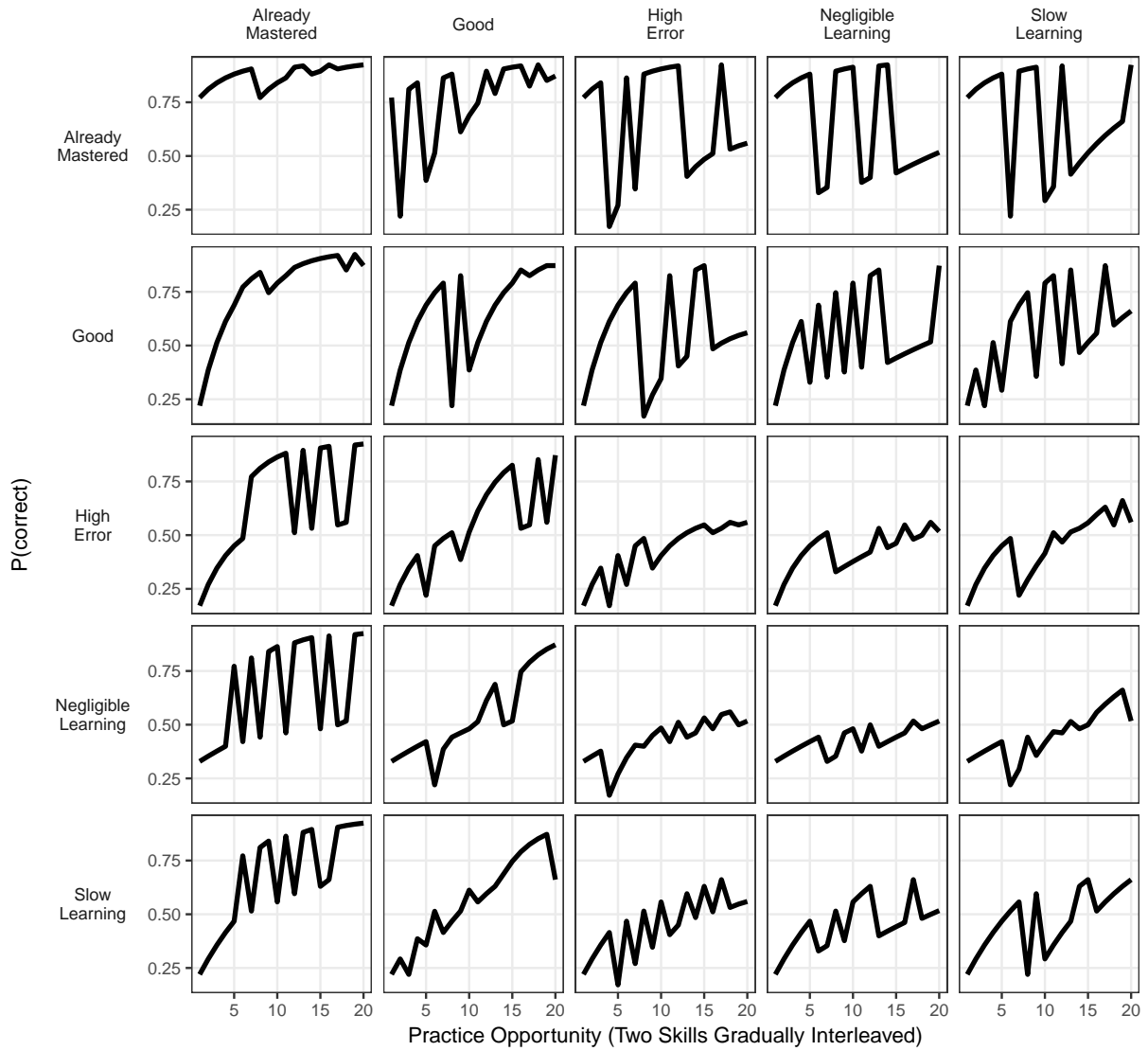


Figure 10: True learning curves when practice on two skills is gradually interleaved