Individualizing Bayesian Knowledge Tracing. Are Skill Parameters More Important Than Student Parameters?

Michael V. Yudelson
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213 USA
+1 (412) 268-5595
yudelson@cs.cmu.deu

ABSTRACT

Bayesian Knowledge Tracing (BKT) models were in active use in the Intelligent Tutoring Systems (ITS) field for over 20 years. They have been intensively studied, and a number of useful extensions to them were proposed and experimentally tested. Among the most widely researched extensions to BKT models are various types of individualization. Individualization, broadly defined, is a way to account for variability in students that are working with the ITS that uses BKT model to represent and track student learning. One of the approaches to individualizing BKT is to split its parameters into per-skill and per-student components. In this work, we are proposing an approach to individualizing BKT that is based on Hierarchical Bayesian Models (HBM) and, in addition to capturing student-level variability in the data, weighs the contribution of per-student and per-skill effects to the overall variance in the data.

Keywords

Student models of practice, Bayesian knowledge tracing, hierarchical Bayesian models, skill vs. student parameterization.

1. INTRODUCTION

Bayesian Knowledge Tracing (BKT) is one of the most popular student modeling techniques in the field of Intelligent Tutoring Systems (ITS). It has been in active use for over two decades and has been confirmed to be the modeling approach researchers can rely on.

Over the years, a large number of extensions to the standard BKT were proposed and tested in posthoc analyses as well as experimentally. Among the most widely researched additions to BKT is the ability to account for students' individual traits. It has been confirmed in the are of modeling student learning in general and in the case of BKT that accounting for student-level variability in the data could benefit the model's statistical goodness of fit, as well as potentially improve the generalizability of the model.

Known approaches could be separated into three groups. The first group, binary multiplexing of the initial skill mastery probability based on the student characteristics, for example, the correctness of the first response (Pardos & Heffernan, 2010). This method has been proven to benefit the overall student model quality, and the implementation of this approach was a runner-up in the 2010 KDD Cup data mining challenge. The second group, fitting BKT parameters not across students for a particular skill, but for a student/skill pair (Lee & Brunskill, 2012). This approach has not been evaluated for predictive correctness. The third group, are the methods separating BKT parameters into per-student and per-skill components (Corbett & Anderson, 1995; Yudelson et al., 2013). The two approaches from the third group were shown to improve model fits reliably.

While the BKT individualization approaches mentioned above were successful in one way or the other, are arguably yet to achieve a sufficient flexibility and rigor of the available parameterization devices. In this paper, we propose and investigate an individualized Bayesian Knowledge Tracing that, on top of refining certain aspects of its predecessor (Yudelson et al., 2013), draws on the flexibility of the Hierarchical Bayesian Models' representation to capture relative weight of student-level and skill-level variability in the learning data as defined by respective parameters. Also, we empirically explore the possibility of clustering student-level factors via mixes of Gaussian distributions.

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 outlines the methods. Section 4 describes the data we used for this investigation. Section 5 talks about the results. Finally, Section 6 closes with a few discussion points.

2. RELATED WORK

2.1 Bayesian Knowledge Tracing

Bayesian Knowledge Tracing (BKT) is a probabilistic framework (Corbett & Anderson, 1995) it is used to assess student progress with a unit of knowledge often referred to as skill. Upon correct or incorrect action, an estimate of student mastery of skill(s) is re-computed. Computationally, BKT is a Hidden Markov Model with two hidden states, representing whether a particular skill is un-mastered or mastered. Observations of student performance on opportunities to practice a skill are binary: a student either solves a problem step correctly or not (due to error or because of a hint request). While students might go through dozens of attempts to get a particular step correct, traditionally, only students' first attempts are considered for updating skill mastery estimates.

There are four skill parameters used in BKT: initial probability of knowing the skill a priori – $p(L_0)$ (or *p-init*), probability of student's knowledge of a skill transitioning from not known to known state after an opportunity to apply it – $p(T)$ (or *p-learn*), probability to make a mistake when applying a known skill – $p(S)$ (or *p-slip*), and probability of correctly applying a not-known skill – $p(G)$ (or *p-guess*). Given that parameters are set for all skills, the formulae used to update student knowledge of skills are as follows. The initial probability of student u mastering skill *k* is set to the *p-init* parameter for that skill Equation (1a). Depending on whether the student u applied skill k correctly or incorrectly, the conditional probability is computed either using Equation (1b) or Equation (1c). The conditional probability is used to update the

probability of skill mastery according to Equation (1d). To compute the probability of student *u* applying the skill *k* correctly on an upcoming practice opportunity one uses Equation (1e).

$$p(L_1)_u^k = p(L_0)^k \quad (1a)$$

$$p(L_{t+1}|obs = correct)_u^k = \frac{p(L_t)_u^k \cdot (1 - p(S)^k)}{p(L_t)_u^k \cdot (1 - p(S)^k) + (1 - p(L_t)_u^k) \cdot p(G)^k} \quad (1b)$$

$$p(L_{t+1}|obs = wrong)_u^k = \frac{p(L_t)_u^k \cdot p(S)^k}{p(L_t)_u^k \cdot p(S)^k + (1 - p(L_t)_u^k) \cdot (1 - p(G)^k)} \quad (1c)$$

$$p(L_{t+1})_u^k = p(L_{t+1}|obs)_u^k + (1 - p(L_{t+1}|obs)_u^k) \cdot p(T)^k \quad (1d)$$

$$p(C_{t+1})_u^k = p(L_t)_u^k \cdot (1 - p(G)^k) + (1 - p(L_t)_u^k) \cdot p(G)^k \quad (1e)$$

2.2 Introducing Student-Level Factors to the Bayesian Knowledge Tracing

Having student-level parameters is a regular feature of models of student learning and learning performance. The logistic regression based Rasch model (van der Linden & Hambleton, 1997) that captures test item complexity and its extension –the Additive Factors Model (Cen et al., 2008) both include a parameter to account for variability in the student a priori abilities. Including student-level parameters in these models helps both the fit as well as the interpretability of the models overall.

There were a few attempts to introduce student-specific parameters to otherwise skill-only standard BKY. The original work on BKT (Corbett & Anderson, 1995) discussed fitting skill-level and student-level parameters on respective slices of the data to later combine and apply the two in the context of each student-skill pair. As a result, the correlation of expected and observed within-student accuracies was higher for the thus individualized model.

Another approach to individualization suggests the multiplexing probability of initial skill mastery (*p-init*) based on student cohort (Pardos & Heffernan, 2010). Based on the correctness of the first student's response, the appropriate skill *p-init* is set to the lower or higher predetermined constant. This prior-per-student model outperforms standard BKT on a significant fraction of problem sets authors considered.

According to yet another approach (Lee & Brunskill, 2012), BKT parameters were fit within each student-skill pair's data slice and not across skills or students. Authors did not discuss on the goodness of fit of their individualized models, however. Their primary focus was on whether the individualized model when deployed in an intelligent tutoring system, would schedule fewer or more problems to be solved as compared to standard BKT model. The conclusion was that a considerable fraction of students, as judged by individualized model, would have received a significantly different amount of practice problems.

Finally, another individualization approach that we would be using for comparison in this work suggests something akin to the original discussion of the BKT individualization (Yudelson et al., 2013). Student and skill components of BKT parameters are fit one set after the other using a coordinate gradient descent procedure with an active parameter set maintained throughout the process. In addition to improved fits, BKT models individualized this way were shown to lead to optimized problem-sequences leading to saving students some efforts.

Overall, there is enough evidence that introducing student-level parameters to BKT benefits the fit of the model and could optimize student learning experience.

2.3 Introducing Item-Level Factors to the Bayesian Knowledge Tracing

Recently, a noticeable amount of work focused on addressing item-level variability in BKT models. Pardos & Heffernan (2011) presented their KT-IDEM model that features special nodes that capture item difficulties and, together with skill-level latent variables are influencing the student performance.

In the approach Huang and colleagues took (Huang et al., 2015), it is possible to address not just items, but even item level features, adding parameters in a way it is done in regression analysis. In another work (Khajah et al., 2014), authors are discussing merging an IRT model and BKT model. This approach resulted in an HBM that combines features of both. It is worth to note that the latter two use Markov Chain Monte Carlo methods to fit their models.

3. METHODS

Our objective is to introduce further improvements to the approach to individualizing BKT and draw comparisons to regular BKT as well its original version in terms of statistical fitness as well as and to attempt to judge the plausibility of their respective student-level parameters.

3.1 Individualized BKT Model via Parameter-Splitting

Individualization of the BKT that was proposed in (Yudelson et al., 2013) prescribes to put every individualized parameter in the context of a particular student that works on a particular skill. In this context, *p-init*, *p-learn*, *p-slip*, and *p-guess* parameters have two components: a per-skill component and a per-student component. The two are combined using a pairing function shown in Equation 2a. Here, components are first converted from probability scale to log-odds scale using logit function (Equation 2b), added, and the sum is converted back to the probability scale using sigmoid function (Equation 2c). An individualized model, where all per-student components are equal to 0.5 (0 on the log-odds scale) is equivalent to the standard BKT model.

$$f(P_k^i, P_u^i) = S\left(l(P_k^i) + l(P_u^i)\right) \quad (2a)$$

$$l(p) = \ln\left(\frac{p}{1-p}\right) \quad (2b)$$

$$\mathrm{S(x)} = \frac{1}{1 + \mathrm{e}^{-\mathrm{x}}} \quad (2c)$$

Fitting of such individualized BKT (iBKT) model is done by computing gradients of the log-likelihood function given individual student/skill data samples with respect to every iBKT parameter (Levinson et al., 1983). On every odd run, gradients are aggregated across skills to update skill component of the parameters. On every even run, the gradients are aggregated across students to update respective student components. This

block-coordinate descent is performed until all parameter values stabilize up to a pre-set tolerance criterion. An active set of parameter components is maintained to fit only those that still haven't stabilized. An extended discussion of the method, as well as derived formulas for the gradients is given in the original publication of this work (Yudelson et al., 2013).

The standard and individualized model described above we implemented in the tool called `hmm-scalable`. The tool has a suite of solvers, including the classical BKT Expectation Maximization solver for standard BKT, as well as a set of stochastic and conjugate gradient descent solvers. `Hmm-scalable` is freely available on GitHub repository[1] of the International Educational Data Mining Society (standard BKT models only).

3.2 Individualized BKT via Hierarchical Bayesian Model

We have also implemented the BKT as well as the iBKT approach described above in the form of a Hierarchical Bayesian Model (HBM). HBMs allow for a more universal and flexible way of representing iBKT. The HMB BKT just like the hmm-scalable BKT had 4*N* parameters, where *N* is the number of skills. In the iBKT models, both hmm-scalable and HBM version, only the *p-init* and *p-learn* were individualized. Thus, the number of parameters in the hmm-scalable version of iBKT was 4*N*+2*M*, where *M* is the number of students. HBM version of the iBKT treated per-student parameters as being drawn from Gaussian distributions and had 4 hyper-parameters: mean and standard deviation for student-level *p-init* and *p-learn*. While we did not specifically check or prove this, but intuitively, confining a parameter to the bounds of a particular distribution serves as a form of regularization and, theoretically, could improve the generalizability of the model. Although iBKT models 4*N*+2*M* had parameters, the per-student and per-skill parameters, when combined using the pairing function from Equation 2a, could result in up to 2*N*+2*NM* in-context parameters. *P-guess* and *p-slip* were not individualized (2*N*), 2NM represents all possible combinations of students and skills for *p-init* and *p-learn*.

$$f\left(P_k^i, P_u^i, W_0, W_k, W_u, W_{uk}\right) = \\ = S\left(W_0 + W_k l\left(P_k^i\right) + W_u l\left(P_u^i\right) + W_{uk} l\left(P_u^i\right) l\left(P_k^i\right)\right) \quad (3)$$

The main contribution of this paper is to not only mix per-student and per-skill parameters together but to weight each component of the mixture in an attempt to define whether either one has a larger impact on the resulting in-the-context parameter value. We have taken Equation (2a) and changed into Equation (3). Here we have the bias term (W_0), the weights for the per-skill and per-student components (W_k and W_u respectively), and also the interaction term for the two with the weight (W_{uk}). The $W_{\cdot}$ weights are drawn from Gaussian distribution. Each of them is constrained to [0, 1], and the sum is fixed at 2. We have used the same $W_{\cdot}$ weights for mixing both *p-init* and *p-learn*. Thus, we have 8 additional hyperparameters and this new model, that we will refer to as iBKT-W HBM, has 4*N*+2*M*+4 parameters and 12 hyper-parameters. If $\{W_0, W_k, W_g, W_{gk}\}$ weights were set to {0, 1, 1, 0} respectively, the model would we equivalent to the iBKT HBM model.

When exploring the per-student parameter values if the iBKT-W HBM model, we have noticed that, in spite of being drawn from the Gaussian distribution, the actual distribution has a hint of being binomial (rf. Figure 1). It is especially visible for the distribution of the per-student values of *p-init*. In order to address this phenomenon, we have created yet another HBM model, that we will call iBKT-W-2G HBM, where the per-student *p-init* and *p-learn* parameters will be drawn from a mixture of 2 Gaussian distributions. In this new model, there are 4 means of the Gaussians distributions (2 for per-student *p-init* and 2 per-student for *p-learn*), 2 variances (1 for per-student *p-init* and 1 per-student for *p-learn*) instead of 4 as in iBKT-W HBM. The membership in one or the other mixture is modeled by a 2-parameters categorical distribution based on Dirichlet(1,1) distribution. Thus, there are, just as before, 4*N*+2*M*+4 parameters, while the number of hyperparameters is 16. Table 1 summarizes the information about parameters of all of the models we have considered in this work.

HBM versions of the three iBKT models are not supported by `hmm-scalable`. To build them we used BUGS language (Lunn et al., 2009) implemented as `rjags` package in `R` (Plummer, 2016). As opposed to `hmm-scalable`, that uses a form of exact inference, BUGS models were build using the Gibbs Sampler implemented in the `rjags` package.

To fit HBM iBKT models we used 10 chains running in parallel for the duration of 500 iterations. Unfortunately, it is not possible whether a model fit using a Gibbs sampler has converged. It is, however, possible to say whether it did not. In our experimental runs, we have confirmed there were no signs that the models failed to converge. Each model took roughly 1 hour to finish.

Table 1. Model parameters and hyper-parameters. Number of skills – *N*, number of students – *M*

Model	Parameters	Hyper-parameters
Majority Class	0	0
Standard BKT hmm-scalable	4N	0
Standard BKT JAGS	4N	0
iBKT hmm-scalable*	4N+2M	0
iBKT HBM*	4N+2M	4
iBKT-W HBM *	4N+2M+4	12
iBKT-W-2G HBM *	4N+2M+4	16

* for all iBKT models we only individualize *p-init* and *p-learn*.

4. DATA

We used the data from the KDD Cup 2010 Educational Datamining Challenge[2]. The data was donated by Carnegie Learning Inc., a publisher of mathematics curricula and a producer of intelligent tutoring system – Carnegie Learning's Cognitive Tutor – for middle school, high school, and college. The KDD Cup 2010 datasets are quite large. Algebra dataset has close to 10 million student transactions, and pre-algebra dataset has a little over 20 million transactions.

Although computational capabilities of the `hmm-scalable` tool allow fitting BKT and iBKT models within minutes, R

[1] `https://github.com/IEDMS/standard-bkt`

[2] `http://pslcdatashop.web.cmu.edu/KDDCup`

implementation of the Gibbs Sampler and the BUGS language are not as scalable. Because of that, we have selected a subset of the pre-algebra dataset, namely, a sample where students worked on Linear Inequalities unit. This sample consisted of 66,307 transactions of 336 students. This sample only contained transactions labeled with the skills that the Carnegie Learning's Cognitive Tutor tracks. There were 30 skills that the unit on linear inequalities taught.

From the rich feature set of the data we took four columns: success at first attempt at a problem step (student activity is blocked and sequenced into working on individual problem steps and BKT traditionally only looks at the first attempt; anonymous student id; concatenation of curriculum unit, section, and problem (was not necessary for our analyses, but required by `hmm-scalable`); and relevant skill(s) practiced at that particular step.

5. RESULTS

5.1 Model Fits

The results of statistical fitness of the models we have discussed are in Table 2. There we list four fitness metrics, the Deviance Information Criterion (van der Linde, 2005), root mean squared error, Accuracy and area under ROC curve (A'). DIC is a metric based on log-likelihood. It is often used for Bayesian model selection. Accuracy is a point measure of how often the model guesses the correct response (here whether the student was correct or incorrect). RMSE goes a little further by quantifying how close the each prediction is to the correct classification of a correct or incorrect response. The area under the ROC curve is a measure of how well the model can tell the classes or responses apart. As the name suggests, it is a curve metric, without a working point, like accuracy (with which a 0.5 threshold is often used).

As wee can see in Table 2, the majority class model performance is low as expected A' is at 0.50 (as it should be), accuracy is about 72%. There are usually more correct responses in the Carnegie Learning's Cognitive Tutor data since the tutor breaks problems into steps and guides students towards the correct solution.

As we move down in Table 2, we can see that model accuracies start improving. Standard BKT models outperform Majority Class. There is a small advantage of the HBM model fit using `R` implementation of JAGS over the `hmm-scalable`. iBKT models (here we only individualize *p-init* and *p-learn*) are a further improvement of the fit, again, with a small advantage for the HBM version of the model. The weighted version of the iBKT (iBKT-W) is only implemented as an HBM and, again, shows an improvement overall (in terms of DIC, RMSE, and A').

Table 2. Performance of the models

Model	DIC	RMSE	Acc.	A'
Majority Class		0.52516	0.7242	0.5000
Standard BKT hmm-scalable	66230	0.40571	0.7561	0.7649
Standard BKT HBM	65347	0.40299	0.7569	0.7728
iBKT hmm-scalable*	64215	0.39376	0.7680	0.7990
iBKT HBM *	63644	0.39287	0.7692	0.7992
iBKT-W HBM *	63587	0.39236	0.7687	0.8005
iBKT-W-2G HBM *	63412	0.39252	0.7689	0.8005

* for all iBKT models we only individualize *p-init* and *p-learn*.

In addition to observing model fits, we have performed one round of 3-fold item-stratified cross-validation to verify whether the differences between the iBKT model fit by `hmm-scalable` and the iBKT-W model fit by JAGS become more visible. Although the fit metrics deteriorated a bit, the partial order of the models regarding the goodness of fit did not change.

5.2 Per-Skill and Per-Student Parameters

When we plotted the densities of per-student *p-init* and *p-learn* parameters for the weighted iBKT, we have noticed that the distributions had a hint of bimodality, especially the distribution of per-student *p-init* (rf. Figure 1). Given that the HBM is drawing parameter values from a Gaussian distribution, the bi-modality is quite pronounced. To check our intuition, we have constructed a modified version of the weighted iBKT where per-student p-init and p-learn are mixstures of two Gaussians. The new model, iBKT-W-2G, did not show improvement in fit statistics, except for DIC. However, the distributions of the corresponding per-student *p-init* and *p-learn* were visibly bimodal (rf. Figure 6). The two means for the p-init parameters are 0.280 and 0.786. The two means for the p-learn parameters are 0.277 and 0.630.

The weights for pairing the per-student and per-skill parameters for both of the weighted iBKT models are given in Table 3. Both the bias weight W_0 and interaction W_{uk} seem to be sufficiently small. Although there is no exact agreement between the two models, in both the weight of the per-skill parameters (W_k) are two to three times smaller than that of per-student parameters (W_u).

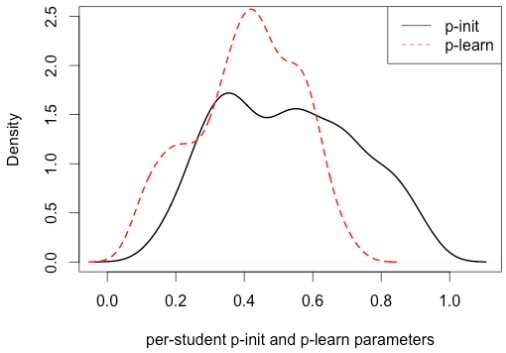

Figure 1. Density plots for per-student *p-init* and *p-learn* parameters of iBKT-W HBM model.

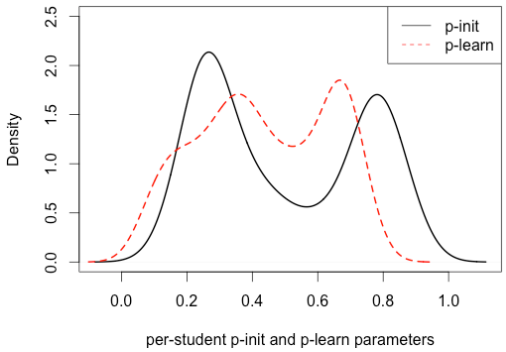

Figure 2. Density plots for per-student *p-init* and *p-learn* parameters of iBKT-W-2G HBM model.

Table 3. Skill-student weights in iBKT-W models

Model	W_0	W_k	W_u	W_{uk}

iBKT-W HB	0.012	0.565	1.420	0.004
iBKT-W-2G HBM	0.019	0.700	1.274	0.007

5.3 Extra Look At Per-Skill and Per-Student Parameters

In an attempt to investigate the differences between iBKT model fit using `hmm-scalable` and the iBKT-W-2G fit using JAGS, we have plotted the per-student *p-init* and *p-learn* parameters for both. The respective plots are in Figure 3 and Figure 4. As we can see in Figure 3, where per-student parameters of iBKT `hmm-scalable` model are plotted, correlation of *p-init* and *p-learn* is mid-range and is equal 0.55. Notably, a tangible portion of students, as estimated by the model, have low *p-init* and high *p-learn* parameters. If we interpret *p-init* as student's overall prior preparation and *p-learn* as student's overall rate of learning, these would be the students that came in with the low level of knowledge and quickly caught up. Using the same logic, there are also a few students that came in with high prior knowledge but suffered from low learning rate.

The plot of per-student *p-init* and *p-learn* parameters of iBKT-W-2G HBM model is entirely different (rf. Figure 8). The correlation is very high – 0.90. Although the student points are lined up almost linearly, it is possible to discern two clusters (lower left, and upper right) that roughly correspond to two mixed Gaussians represented by a categorical node in the model. Here, there are effectively no students in the upper left or bottom right corners of the graph. Namely, those arriving with lower preparation, but the high rate of learning, or, vice-versa, high preparation, but the lower rate of learning. The former is unfortunate since the unprepared students that can quickly close the gap are, arguably, the most desired ones since they make the application that assisted them (e.g., Carnegie Learning's Cognitive Tutor) shine.

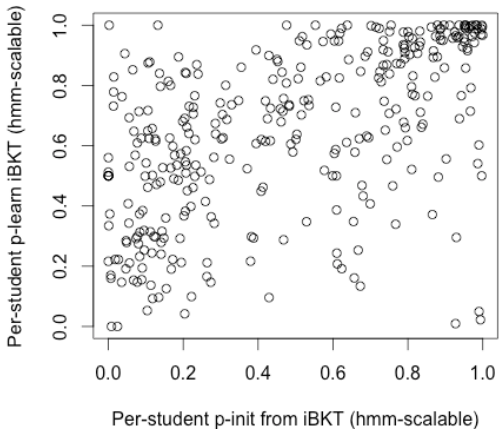

Figure 3. Scatter plot of per-student *p-init* (x-axis) and *p-learn* (y-axis) from iBKT model fit by hmm-scalable. The correlation between the two is 0.55 (significant at 0.001 level).

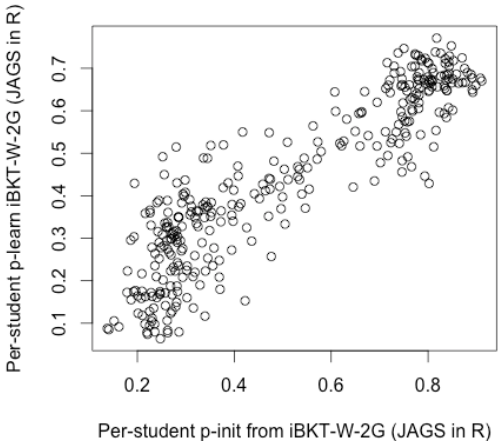

Figure 4. Scatter plot of per-student *p-init* (x-axis) and *p-learn* (y-axis) from iBKT-W-2G model fit by JAGS in R. The correlation between the two is 0.90 (significant at 0.001 level).

6. DISCUSSION

6.1 Small Differences in Statistical Fits

Arguably the most pressing question about comparing the `hmm-scalable`-fit iBKT model and the HBM models is why the differences in statistical accuracy are so small. Given that some of the changes in per-student parameters are quite large (rf. Figures 3 and 4), we are to expect more pronounced differentiation, especially since the fitting method and parameterization changed.

We would like to refer to an earlier work where we examined alternative parameterizations of a logistic regression model of student math learning (Yudelson et al., 2011). As we have found there, despite virtually no difference in statistical fit, the parameter values and especially their interpretability improved. We did not estimate the interpretability of the parameter values of the HBM models, however, the relative distribution of the iBKT-W-2G HBM per-student parameters is, arguably, more realistic than that of the iBKT `hmm-scalable`.

Besides, as we were able to show in (Yudelson & Ritter, 2015), the absence of a *tangible* difference in statistical fit between two models may, none the less, correspond to considerable variance in assigned practice when the models compared are deployed in the actual system and used for knowledge tracking and problem selection.

6.2 What Do The Gaussians Mixtures Represent?

We have followed the trace of the possible bi-modal distributions of per-student *p-init* and *p-learn* parameters in the iBKT-W and constructed iBKT-W-2G model where per-student parameters are represented as mixtures of 2 Gaussian distributions with the same standard deviation.

To reverse-engineer the fuzzy mixture variable that *clusters* students we have attempted to correlate it with a set of student performance metrics. These included: overall number of problems solved, time spent, hints requested (both on the first attempt at a step and overall), errors committed (both on the first attempt at a step and overall), percent correct (both on the first attempt at a step and overall), time spent per problem, errors committed and hints requested per problem. None of them correlated with the fuzzy mixture variable reliably. It is likely that the resulting

clustering represents some latent student factor, we just could not interpret it.

6.3 Weighting Per-Skill and Per-Student Parameters

We have tried more models than the two HBM iBKT-W's we reported. The models included those individualizing *p-init* and *p-learn* separately or together, with weighting or without, mixing 1, 2, or 3 Gaussians (18 variants overall) – in all cases per-student parameter component weight was two-to-three times larger than that of per-skill components. One explanation for that could be possible over-fitting. There are 336 students and 30 skills. Even though the model is hierarchical and both per-skill and per-student parameter values are regularized, they are an order of magnitude more per-student values. To confirm or disconfirm the over-fitting hypothesis we would have to perform multiple sample-and-fit rounds where the number of students is equal to the number of skills.

7. ACKNOWLEDGMENTS

The author would like to give special thanks to Mr. Christopher MacLellan for introducing him to the BUGS language, Dr. Ilya Goldin for sharing his draft of a single-skill BKT implementation in BUGS, and Dr. Kenneth R. Koedinger for useful feedback while this work took shape.

8. REFERENCES

[1] Cen, H., Koedinger, K.R., Junker, B.: Comparing Two IRT Models for Conjunctive Skills. In: Woolf, B.P., A ̈imeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 796–798. Springer, Heidelberg (2008)

[2] Corbett, A.T., Anderson, J.R.: Knowledge tracing: Modeling the acquisition of procedural knowledge. User Modeling and User-Adapted Interaction 4(4), 253–278 (1995)

[3] Fawcett, T. (2006) An introduction to ROC analysis. Pattern Recognition Letters, 27, 861–874.

[4] Huang, Y., Gonzalez-Brenes, J. P., and Brusilovsky, P. (2015) The FAST toolkit for Unsupervised Learning of HMMs with Features. In: The Machine Learning Open Source Software Workshop at the 32nd International Conference on Machine Learning (ICML-MLOSS 2015).

[5] Khajah, M., Wing, R. M., Lindsey, R. V., & Mozer, M. C. (2014) Incorporating latent factors into knowledge tracing to predict individual differences in learning. In J. Stamper, Z. Pardos, M. Mavrikis, & B. M. McLaren (Eds), Proceedings of the 7th International Conference on Educational Data Mining (pp. 99-106).

[6] Lee, J.I., Brunskill, E.: The Impact on Individualizing Student Models on Necessary Practice Opportunities. In: Yacef, K., Zaïane, O.R., Hershkovitz, A., Yudelson, M., Stamper, J.C. (eds.) Proceedings of the 5th International Conference on Educational Data Mining (EDM 2012), pp. 118–125 (2012)

[7] Levinson, S.E., Rabiner, L.R., Sondhi, M.M.: An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition. Bell System Technical Journal 62(4), 1035–1074 (1983)

[8] van der Linde, A. (2005). DIC in variable selection. *Statistica Neerlandica*, 59: 45-56.

[9] van der Linden, W.J., Hambleton, R.K.: Handbook of Modern Item Response Theory. Springer, New York (1997)

[10] Lunn D, Spiegelhalter D, Thomas A, Best N. (2009) The BUGS project: Evolution, critique, and future directions. *Statistics in Medicine*, 28:3049-67.

[11] Pardos, Z.A., Heffernan, N.T.: Modeling Individualization in a Bayesian Networks Implementation of Knowledge Tracing. In: De Bra, P., Kobsa, A., Chin, D. (eds.) UMAP 2010. LNCS, vol. 6075, pp. 255–266. Springer, Heidelberg (2010)

[12] Pardos, Z. & Heffernan, N. (2011) KT-IDEM: Introducing Item Difficulty to the Knowledge Tracing Model. In Konstant et al. (eds.) Proceedings of the 20th International Conference on User Modeling, Adaptation and Personalization (UMAP). (pp. 243-254), Girona, Spain. Springer.

[13] Plummer, M. (2016). rjags: Bayesian Graphical Models using MCMC. R package version 4-5. https://CRAN.R-project.org/package=rjags

[14] Yudelson, M., Koedinger, K., Gordon, G. (2013) Individualized Bayesian Knowledge Tracing Models. In: Lane, H.C., and Yacef, K., Mostow, J., Pavlik, P.I. (eds.) Proceedings of 16th International Conference on Artificial Intelligence in Education (AIED 2013), Memphis, TN. LNCS vol. 7926, (pp. 171–180).

[15] Yudelson, M., Pavlik, P.I., Koedeinger, K.R. (2011) User Modeling – a Notoriously Black Art. In J.A. Konstan, R. Conejo, J.L. Marzo, and N. Oliver (Eds.) Proceedings of the 19th International Conference on User Modeling Adaptation and Personalization (UMAP 2011), Girona, Spain, (pp. 317-328).

[16] Yudelson, M. & Ritter, S. (2015) Small Improvement for the Model Accuracy – Big Difference for the Students. In: Industry Track Proceedings of 17th International Conference on Artificial Intelligence in Education (AIED 2015), Madrid, Spain.