Regularizing Student Parameters of Individualized Bayesian Knowledge Tracing Model

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

ABSTRACT

We propose a workflow to be implemented in a new workflows architecture of the LearnSphere We present the motivation and initial validation of the value of regularizing student-level parameters in an individualized Bayesian Knowledge Tracing model. Theoretically grounded, regularization of iBKT models, as we show, leads to increased model accuracy.

Keywords

Bayesian Knowledge Tracing, student-level factors, regularization, model-fitting.

1. INTRODUCTION

Bayesian Knowledge Tracing (BKT) model individualization has been an active research topic recently. Individualization is accounting for student population variance that may come in different forms. One of the ways to individualize BKT is to introduce student-specific parameters. The new individualized BKT (iBKT) has been shown to fit the data better and even some promise, if deployed, to be able to save students time by better accounting for personal learning paths [7].

One of the toolkits that support iBKT models and is available via the LearnSphere workflow is `hmm-scalable`[1]. iBKT via `hmm-scalable`, as compared to regular BKT [5], splits every BKT parameter into per-skill and per-student components and allows a subset or all BKT parameters to be individualized. Both student- and skill-level parameter are fit with the help one of the gradient-based approaches including stochastic or conjugate gradient descent, or a Lagrange updates method [6].

The solvers implemented in the core `hmm-scalable` utility that the LearnSphere iBKT wrapper workflow uses are fitting both skill and student parameters in fixed-factor fashion. However, it is often advised to treat student-related variables as random factors. The logic behind this is that students at hand are sampled from a larger population of students [2]. Treating student factors as fixed effects could lead to *fixed fallacy* [4] and deflated generalizability of the model.

In this paper, we propose to take advantage of the regularization feature in `hmm-scalable`. This feature adds a penalty to the objective log-likelihood function in the form of the L^2 norm. That is an approximation of treating the regularized parameter as a random factor. The influence of the L^2 term is weighted by the λ coefficient. The larger the λ, the stronger the effect of the regularization. In simpler terms, regularization penalizes deviation of parameter values from the centroid value. In regularized regression, the centroid is usually 0, and regularization takes a form of scarcity-inducing mechanism. In the case of iBKT, as it is implemented in `hmm-scalable`, an appropriate centroid is 0.5. If student-level parameters have a value of 0.5, this effectively means no student-level effect on the relevant still parameter.

Making the right choice of the λ that would lead to an improved fit (and, potentially, the generalizability) is a problem in an of itself. The `hmm-scalable` does not support the search for an optimal λ directly. We propose to perform this search as part of the wrapping LearnSphere workflow.

2. WORKFLOW METHOD

2.1 Data Inputs

The student-parameter-regularizing workflow takes the same data inputs with regular BKT workflow. These are integer correctness variable (correct encoded as 1, incorrect as 2), student as a character factor, the problem as a character factor, and delimited skill(s) as character factor.

2.2 Workflow Model

The workflow performs a sequence of runs of a chosen iBKT model with a set of λ weights for the regularizing L^2 parameter penalty, where the first value is 0 (the default), and the rest could be defined by the user. For example, from 0.1 to 4, with a 0.1 increment. In this model, the skill parameters are not regularized.

2.3 Workflow Outputs

The output of the workflow is built on the standard output of the BKT workflow. An array of statistical fit metrics (Log-likelihood, AIC, BIC, RMSE, Accuracy) is presented along with the λ weight used for the regularization. The main outcome is whether the regularization improves the fit of the model, as compared to the non-regularized version, and at what λ value. A graphical web-friendly plot of a selected via drop-down fit metric across all tested λ values could help better visualize the effect of the regularization. Skill and student level parameters could also be printed for the winning model.

3. DISCUSSION

As part of pilot-testing the method we propose to be implemented as a LearnSphere workflow, we have run a series of iBKT models with student-level parameter regularizations for a set of λ weights. Initially, we used values from 0 to 6 with a step of 1. Later, we added values 0.1 to 0.9 with a step of 0.1, 1.5, 2.5, and 3.5.

We used the 2010 KDD Cup Challenge Set B (available via PSLC DataShop[2]), the largest available dataset of student learning data.

[1] hmm-scalable toolkit, the public version https://iedms.github.io/standard-bkt

[2] 2010 KDD Cup https://pslcdatashop.web.cmu.edu/KDDCup

The overall number of student-step transactions in that dataset is over 20 million. The data in that dataset was collected by the application called Cognitive Tutor. It was donated by Carnegie Learning Inc. and contained data of students working on a Bridge to Algebra course.

For each setting, we performed 5 runs of 2-fold student-stratified cross-validation. As identified in [4], this setup of the cross-validation allows for the most potent way to reliably rank alternative models when using cross-validation. One of the guarantees of this cross-validation approach is appropriately sized confidence interval around the accuracy metrics.

For the sake of selecting the best λ, we used simple average across 5x2=10 statistical goodness of fit values. We picked RMSE and Accuracy as our metrics. Figures 1 and 2 show the plots of the RMSE and Accuracy (respectively) for the set of tested λ weights.

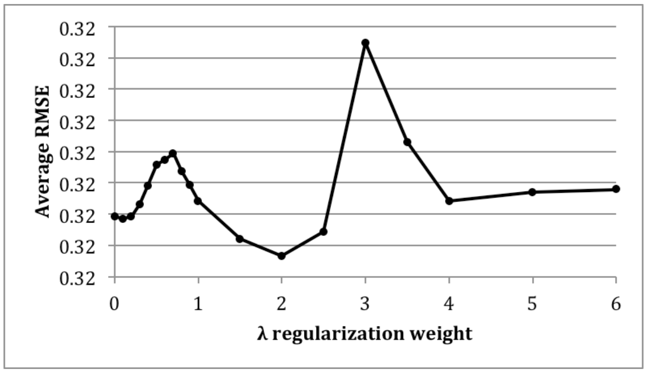

Figure 1. λ regularization weights vs. average RMSE

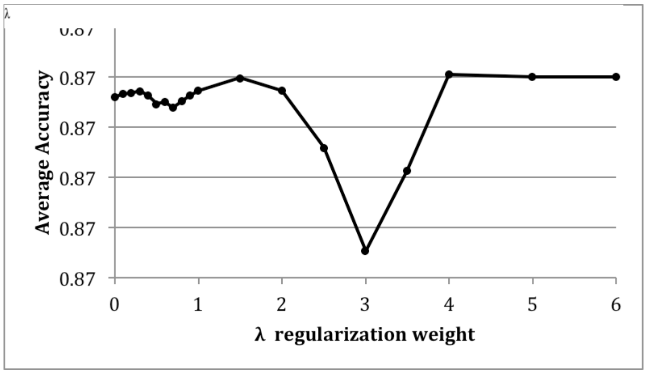

Figure 2. λ regularization weights vs. average Accuracy.

Although RMSE and Accuracy plots are not entirely in sync: the best (lowest) RMSE value corresponds to λ=2.0, and the best (highest) Accuracy value corresponds to λ=4.0, it is, arguably, appropriate to choose λ=2.0, thus trusting RMSE metric.

To underline the usefulness of the regularizing student parameters in iBKT models, we have performed the model ranking F-test described in [4] by making pairwise comparisons between multiple student-stratified 2-fold cross-validated models. We have used a Majority Class model that always predicted correct outcome, a shipped model that used parameters from the deployed Cognitive Tutor, a standard BKT model fit by using Expectation-Maximization (EM) method implemented in `hmm-scalable`, and iBKT non-regularized, and regularized models fit using gradient-based Lagrangian updates solver. The results of the pairwise comparisons are given in Table 1.

Table 1. Comparing multiple cross-validated models.

			p-value of the difference			
Rank	**Mean Acc.**		**Ship**	**EM**	**iBKT**	**iBKT λ=2.0**
4	0.86167	MC	0.000	0.000	0.000	0.000
5	0.85206	Ship		0.000	0.000	0.000
3	0.87010	EM			0.000	0.000
2	0.87130	iBKT				0.327
1	0.87136	iBKT λ=2.0				

			p-value of the difference			
Rank	**Mean RMSE**		**Ship**	**EM**	**iBKT**	**iBKT λ=2.0**
5	0.37192	MC	0.000	0.000	0.000	0.000
4	0.34095	Ship		0.000	0.000	0.000
3	0.32061	EM			0.327	0.128
2	0.32039	iBKT				0.001
1	0.32027	iBKT λ=2.0				

Here we can see that BKT models reliably (with a p-value less than 0.001) beat Majority Class and Shipped models both in terms of Accuracy and RMSE. In terms of accuracy, iBKT models beat standard BKT, while the difference between the best-regularized iBKR and non-regularized iBKT is not significant.

If we look at RMSE, BKT models, again, prevail. There is now the statistical difference between iBKT models: regularized iBKT model has an edge. Standard BKT model fit using EM is not statistically different from either iBKT model in terms of RMSE.

4. REFERENCES

[1] Alpaydin, E. (1999) Combined 5 × 2 cv F test for comparing supervised classification learning algorithms. Neural Computation, 11 (8), 1885–1892.

[2] Baayen, R. H., Davidson, D. J., & Bates D. M. (2008) Mixed-effects modeling with crossed random effects for subjects and items. Journal of Memory and Language, 59(4), 390-412.

[3] Cen, H., Koedinger, K. R., & Junker, B. (2008) Comparing Two IRT Models for Conjunctive Skills. In Woolf, B.P, A¨ımeur, E., Nkambou, R., and Lajoie, S. (Eds.), Proceedings of the 9th international conference on Intelligent Tutoring Systems (ITS '08), Springer-Verlag, Berlin/Heidelberg, (pp. 796-798).

[4] Clark, H. H. (1973). The language-as-fixed-effect fallacy: A critique of language statistics in psychological research. Journal of Verbal Learning and Verbal Behavior, 12, 335-359.

[5] 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.

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

[7] 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).