Performance Comparison of Item-to-Item Skills Models with the IRT Single Latent Trait Model

Michel C. Desmarais

Polytechnique Montréal
michel.desmarais@polymtl.ca

Abstract. Assessing a learner's mastery of a set of skills is a fundamental issue in intelligent learning environments. We compare the predictive performance of two approaches for training a learner model with domain data. One is based on the principle of building the model solely from observable data items, such as exercises or test items. Skills modelling is not part of the training phase, but instead dealt with at later stage. The other approach incorporates a single latent skill in the model. We compare the capacity of both approaches to accurately predict item outcome (binary success or failure) from a subset of item outcomes. Three types of item-to-item models based on standard Bayesian modeling algorithms are tested: (1) Naive Bayes, (2) Tree-Augmented Naive Bayes (TAN), and (3) a K2 Bayesian Classifier. Their performance is compared to the widely used IRT-2PL approach which incorporates a single latent skill. The results show that the item-to-item approaches perform as well, or better than the IRT-2PL approach over 4 widely different data sets, but the differences vary considerably among the data sets. We discuss the implications of these results and the issues relating to the practical use of item-to-item models.

Key words: IRT, Bayesian Models, TAN, Learner models

1 Introduction

A number of adaptive applications need a learner model to assess the student skills. They will query this model to find out if a given concept is known, or if a skill is mastered, to perform some adaptation of the learning environment to the user's profile. The skill modelled is an abstraction that cannot be measured directly. A skill is often referred to as a learner's *latent trait* that will determine the successes or failures to some test items or exercises. It is often represented as a probabilistic abstraction, to reflect the fact that stochastic factors like slips and guesses influence the success or failure outcome to item trials.

We explore two means to create such abstractions. One is to integrate skills directly along observable items in a domain model. Hierarchies of skills, where observable items are situated at the bottom of this hierarchy, is a typical example of a domain model that is commonly found in the literature of intelligent tutoring systems and most often modeled as a Bayesian Network or some hybrid derivative

(for eg. [22,5,4]). Standard algorithms for probabilistic inference can then be used to infer the probability of mastery of skills given observed items.

Another approach relies on a Q-matrix [20], which defines which skills are linked to each test items. A familiar example that can be considered as a *summative* assessment with a Q-matrix is a standard questionnaire scoring scheme, where each question is given a weight and the weighted sum of successes to each question yields the assessment of the skill that is intended to be measured by the questionnaire. The skills are the columns of the Q-matrix and the items are the row, and the contribution of each item to a set of skills is given by the weights in the matrix. Assuming a matrix of n rows representing items, and m columns representing skills, and assuming that if a value greater than 0 in cell (i, j) represents the weight of item i to skill j, then we can compute the skill profile of a student through the dot product of the student's item response outcomes vector and the Q-matrix. This product is a skills mastery vector which readily can be normalized to obtain the percent mastery of each skill, for example.

The summative assessment approach to skill assessment with a Q-matrix is not probabilistic in itself, but if the student item outcome matrix contains probabilities of mastery, then the resulting skills assessment is probabilistic.

The choice between the item-to-item approach or the latent traits approach (eg. Bayesian Network) is a compromise between a number of factors to consider, such as knowledge engineering efforts, computational complexity, and most importantly reliability and accuracy of predictions. A number of researchers in the learner modeling field have investigated this issue over the last decade or so [22,5,4,6,1,15].

This paper revisits the issue of assessing item-to-item model performance by comparing the predictive accuracy of standard Bayesian classifier algorithms [10] to create item-to-item learner models with that of the IRT approach (see [21]), which contains a single latent skills. These approaches readily lend themselves to a fair comparison to the extent that each of them are solely data driven and require no knowledge engineering effort for the purpose of predicting item outcome. This would not the case if we wanted to predict the mastery of a set of (unobservable) skills, in which case both approaches would require some knowledge engineering effort, such as defining a Q-matrix or defining the topology of a Bayesian Network, as well as independent means to assess the skills for validation purpose.

Similar studies were conducted by Desmarais et al., [6,7]. These studies respectively compared the performance of a Bayesian Network developed by Vomlel [23] and of the IRT approach with a derivative of a Naive Bayes item-to-item model (POKS). The results showed that for predicting item outcome, POKS performed slightly better than the two other approaches. The current study extends this work by comparing IRT with three standard probabilistic inference techniques: (1) the Tree Augmented Naive Bayes (TAN), (2) a variant of TAN that relies on the K2 search algorithm, and (3) the simple Naive Bayes model. Because the POKS technique used in the work of Desmarais et al. (2005, 2006) integrates a feature selection algorithm in addition to the probabilistic infer-

ence techniques listed, it cannot be directly compared here. However, given that POKS uses a Naive Bayes inference rule, the performance would be expected to be the same as the Naive Bayes technique of this study.

The next two sections describe the IRT model and the item-to-item models. They are followed by the description of the experiments methodology and results.

2 Model with a Single Latent Trait: Item Response Theory

The Item Response Theory (IRT) model [21] is the most widely studied model in psychometrics and routinely used for Computer Adaptive Testing applications. It also gained some adoptions by the intelligent learning community in the last decade or so. This model assumes that the success to all items in a test is determined by a single skill, θ. This skill is referred to as the latent trait. The model can be graphically represented by the network in figure 1.

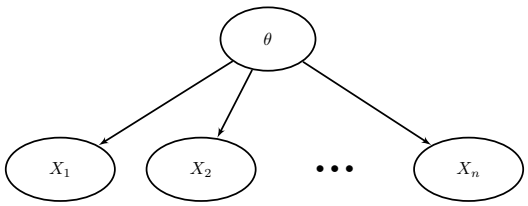

Fig. 1. Generic graphical representation of an IRT model

In the two parameter version of IRT, the probability of success to a single item, X_i, is determined by the logistic function:

$$P(X_i|\theta) = \frac{1}{1 + e^{-a_i(\theta - b_i)}} \tag{1}$$

where parameter a_i is the *discrimination* and b_i is the *difficulty* of item i. A multiplicative factor of 1.7 is often added to a to fit the curve closer to the integration of the normal curve and align it with the so called normal ogive model of the original IRT theory. These parameters are estimated from a training sample and they are specific to each item i (see [3]). This model has a single latent trait (skill) corresponding to θ, which is estimated by maximizing its value according to the observed outcomes to a vector of item nodes $\mathbf{X}$ and under the assumption of independence of the conditional probabilities $P(X_i|\theta)$:

$$\arg\max_{\theta} P(\theta|\mathbf{X}) = P(\theta|X_1, X_2, ..., X_n) = \prod^{n} P(X_i|\theta) \tag{2}$$

3 Bayesian Models Without Latent Skills

To compare the predictive performance of latent models vs. non latent models, we now consider three types of Bayesian classifier models which do not integrate any latent traits (such as θ in IRT):

Naive Bayes (NB) The Naive Bayes model can be represented as figure 1's network, but the (latent) class node θ is replaced by some node X_k for which we aim to predict the most likely binary value, $\{0, 1\}$ (or predict the probability of each value). Computation follows the structure of equation 2, except that instead of maximizing the conditional probability $P(\theta|\mathbf{X})$, we maximize for X_k:

$$\arg \max_{X_k=\{0,1\}} P(X_k|\mathbf{X}) = \prod_{X_i \in \mathbf{X}} P(X_i|X_k) \tag{3}$$

where $\mathbf{X}$ can be any subset of test items excluding X_k.
A distinct equation of the form above is constructed for each of the item in the set. Given that there are no latent trait, the link function of equation (1) is replaced by the conditional probability estimate $P(X_i|X_k)$, which is estimated from the observed frequencies. Akin to the IRT model, independence of the conditional probabilities $P(X_i|X_k)$ is assumed.

Tree Augmented Bayesian Network (TAN) To address the issue that some items may be highly correlated, and therefore that the independence assumption between conditional probabilities does not hold, an alternative class of network topologies was proposed by Friedman et al. [10]: the Tree Augmented Bayesian network (TAN). This topology retains the Naive Bayes topology but it adds a tree structure of links among the leaf nodes. Except for the class root node, each node can have two parents, the class and another node among $\mathbf{X}$. The resulting network creates a tree among the children of the class node X_k (see figure 2). As with the Naive Bayes approach, a different model is created for each item. This structures retains much of the simplicity of Naive Bayes while allowing for efficient network topology induction and inference.

Bayesian Network Classifier (BNC) BNC is a variant of the TAN model that uses the K2 algorithm (see [24]) to search for the tree structure among children nodes. We will name this model a Bayesian Network Classifier in accordance with [24], but the reader should keep in mind that it follows the same topological constraints as the TAN.

4 Experiments

The respective performances of the IRT latent trait model and of the non latent Bayesian models are compared by assessing their predictive power in a simulation

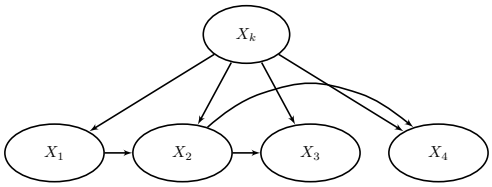

Fig. 2. TAN network example with four predictor items for X_k. In addition to the usual Naive Bayes structure, a tree structure is defined among the leaf nodes, X_1 to X_4.

study. A fixed number of observed item outcomes (success or failure) from a test is fed to the model and we measure the model's ability to correctly classify the outcome of all other observable items from the same individual. These remaining observable items are kept unobserved from the model's perspective, but the real outcome has in fact been recorded, which allows a comparison of the prediction to the reality.

Our choice is to compare predictions over observed data only. Even for the latent model IRT, we do not attempt to derive an independent measure of the skill θ to assess how accurately its estimate/prediction matches. Such procedure was used by Vomlel in an experiment where he asked experts to independently assess concept mastery from test data and compared the assessment of a Bayesian Network over this independent data [23]. Instead, we presume that if θ is correctly estimated, then it will show in the model's ability to predict the observed outcome to items. This approach allows for better experimental replication quality as it is less prone to biases and errors introduced by a few number of experts in assessing skill mastery.

4.1 Data sets

The experiments are conducted over four sets of real data:

College mathematics: Data from a mathematics test administered in 2005 to freshman engineering students that covers their general knowledge of college mathematics. It spans many topics from algebra to analytical geometry, calculus, trigonometry, and exponentials.

Fraction algebra: This data set is from Vomlel's Bayesian Network study [23]. It was administered to 10-12 year old and covers the basics of fraction algebra. Only the data from the 20 question items tests was used and the concept expert assessment was ignored.

LSAT: This data set is available from the `ltm` package written R [17] which can be obtained through the usual CRAN repository[1]. It corresponds to data from the Law School Admission Test.

[1] `http://cran.r-project.org/`

UNIX: A questionnaire developed by the author to assess knowledge of the Unix shell commands. It contains question items that cover basic knowledge to advanced topics and was administered to respondents having a very large array of expertise. This distribution of item difficulty and respondent expertise allows for strong classification performance.

All these sets are composed of binary success/failure data with few missing values, from 0% to 5%, which were recoded as failure answers.

Table 1 reports a number of statistics and informations about the four data sets:

Nb. items: Size of the questionnaire.
Nb. respondents: Number of questionnaires answered by respondents.
Training size: Number of respondents used in a cross validation (the remaining being used for testing).
Avg. respondent score: Average of respondent success rates.
Stdev. score: Standard deviation of success rates.
Nb. folds: Number of folds in the cross-validation experiments for the results reported in the next section.
Nb. features: Number of items fed to the models as observations. These items are selected based on a simple feature selection, namely the degree of correlation with the class variable. Each item has a different set of "feature" items selected for its prediction by the models. The training of the models is done only on the features selected.
Avg. cor. among features: As a measure of the degree to which the independence assumption of the BN and IRT models is violated, we report the average correlation among the features selected.

We can judge from table 1 that the data sets differ widely among them. LSAT is only a 5 items set but, it contains a large number of respondents, whereas UNIX has larger number of items. With only 48 respondents, the UNIX training is limited to 38 cases and the testing to 10, such that the number of folds was increased to obtain more reliable results from the simulations. The correlation among features is also widespread, ranging from 0.08 for LSAT to 0.62 for UNIX. These differences may explain to some extent the large differences in performance reported in the next section.

4.2 Simulation Methodology

Model performance assessment is done through cross-validations. Each model is trained on a portion of the data and tested on the other. For a single fold, the same training and testing sets are used across models to reduce variance. The IRT 2PL model is based on the `ltm` package implemented in R [17]. All three other models are taken from the Weka data mining package [24] and used within R through `RWeka`[2] [14].

[2] RWeka version 0.4-3 and RWeka.jar dated 27 Sept 2010. These packages are available under the CRAN repository. The scripts for the simulations and the data sets are

Table 1. Data sets

	Coll. math	Frac. algebra	LSAT	UNIX
Nb. of items	60	20	5	34
Nb. of respondents	246	149	1000	48
Training size	171	100	900	38
Testing size	75	49	100	10
Avg. respondent score	0.60	0.61	0.76	0.53
Stdev. score	0.15	0.25	0.21	0.29
Nb. of folds	10	10	10	20
Nb. of features	5	5	4	5
Avg. cor. among features	0.17	0.47	0.08	0.62

In accordance with the approach described in section 3, a different model is trained for each item. Although this is not required for the IRT model, for which a single model could be derived for the prediction of all item outcomes, we chose to apply the same methodology throughout all models[3].

Following the usual terminology for classification tasks, we also refer to the predicted item as the target class and to the observed items as features. For each model, 5 features are selected, except for the LSAT data which has only 5 items in total and therefore only 4 other features can be defined. The respective item models are trained only over the selected feature subset. The selection of features for each item is based on the correlation with the target. For a subset of size 5, the top 5 features most correlated with the target nodes are selected. Note that a more sophisticated feature selection algorithm which would take into account intra-features dependencies would likely yield slightly better results from the current experiment for the item-to-item models. However, it remains unclear whether it would favor one item-to-item model over another.

Once a model is trained, the simulation procedure consists in feeding the model with observed items (features). All four models output a probability that the target item will be 0 or 1 and this prediction is compared with the actual respondent's score. Using this probability allows us to derive a ROC curve (Receiver Operating Characteristic), from which the AUC (The Area Under the ROC Curve) score is computed and which serves as one of the performance measure[4]. The other measure reported is the accuracy: if a target node has a probability above 0.5, it is considered true, or false otherwise. Accuracy is reported as percent correct of predictions matched with reality.

---

available from this url: `http://www.professeurs.polymtl.ca/michel.desmarais/Papers/UMAP2011/`.

[3] IRT can predict all items from the same model because θ is the single predictor to all item nodes, whereas for the item-to-item models, a different network is derived for each node.

[4] ROC and AUC analysis are computed with the ROCR package (Sing et al., 2005; available at `http://cran.r-project.org/`)

5 Results

The simulation methodology described above is run over the 4 data sets and the average AUC and accuracy scores are computed. Table 2 reports the different results of the AUC scores for each model and each data set. Each number represents the mean across AUC values of each run, where each AUC value is the average AUC of all question items for a given data set. The number in parenthesis is the standard error across simulation runs.

Table 2 also reports significance levels for three hypothesis tests based on an analysis of variance (AoV)[5]:

All:	all 4 conditions (models)
TAN-IRT:	TAN and IRT conditions alone
w/o IRT:	without IRT (i.e. TAN+BNC+NB)

The AoV test is performed on the AUC score averaged over students and over items.

Table 2. mean (AUC) results of Models for the four data sets

	TAN	BNC	NB	IRT	AoV significance level		
					All	TAN-IRT	w/o IRT
Coll. math	0.77(.012)	0.76(.012)	0.75(.014)	0.74(.013)	***	***	**
Frac. algebra	0.90(.018)	0.90(.018)	0.88(.018)	0.85(.015)	***	***	**
LSAT	0.59(.038)	0.59(.038)	0.58(.039)	0.57(.041)	-	-	-
UNIX	0.96(.021)	0.96(.023)	0.95(.023)	0.91(.036)	***	***	-[(a)]

*** p<0.001, ** p<0.01, * p<0.05 - p>0.05

[(a)] Close to significant: p=0.052

The results show that, for AUC scores, apart from the LSAT data set, almost all the hypothesis tests are positive at the level of $p<0.01$ or $p<0.001$. The TAN-IRT condition indicates that TAN performs significantly better than IRT, with differences in AUC ranging from 3% to 5% for Coll. math, Frac. algebra and UNIX data sets, and 1% for LSAT. TAN and BNC have almost exactly the same performance up to the second decimal, so all conclusions regarding TAN applies to BNC.

Note that even if these differences are small, they must be taken into the context that a random prediction would perform at 0.5 for AUC, and that the relative error reduction from 95% to 97.5% is equivalent to the reduction from 80% to 90% (reducing the remaining error rate by half). Considering this, the 3% AUC error reduction for the UNIX data set is in fact more substantial in relative terms than the 5% fraction algebra. This would be reflected when computing

[5] An analysis of variance is preferred over a Student-t test here to avoid conducting multiple t-tests. Furthermore, the choice of reporting only the TAN-IRT condition over all 6 possible pairs is because TAN seems to yield the best results.

confidence intervals in the prediction of test scores, for example, which entails important implications when a tutoring system needs to gauge the certainty of its assessment. In other words, even if the differences are small in absolute terms, they can have a substantial impact in practice.

The "w/o IRT" condition shows that the thee different latent free Bayesian methods do perform at significantly different levels. The NB condition is systematically lower than the other, suggesting that the added value of the more complex topology of TAN and BNC does yield improvement by accounting for internal correlation among predictor items.

Large differences in the AUC scores are found across data sets. Even if large training samples are available for the Coll. math and LSAT experiments, performance over these tests is the lowest. However, the LSAT relative performance differences is by far the lowest. A possible explanation is that large data sets reduces the predictive advantage of the three other techniques.

Table 3. Accuracy results of Models for the four data sets

	TAN	BNC	NB	IRT	AoV significance level		
					All	TAN-IRT	w/o IRT
Coll. math	0.64(.044)	0.64(.043)	0.63(.044)	0.65(.036)	-	-	-
Frac. algebra	0.70(.069)	0.70(.068)	0.68(.064)	0.71(.047)	-	-	-
LSAT	0.83(.009)	0.83(.010)	0.83(.012)	0.83(.010)	-	-	-
UNIX	0.93(.016)	0.94(.013)	0.91(.021)	0.86(.029)	***	***	***

*** $p<0.001$, ** $p<0.01$, * $p<0.05$ - $p>0.05$

Table 3 reports the accuracy scores with a cutoff of 0.5 (an item is considered succeeded if the estimated probability is above 0.5). The scores are obtained according to the procedure described in section 5, and the significance levels reported are for the same conditions as the AUC score.

Accuracy scores show no significant differences among models, except for the UNIX data set, and indicate that accuracy is a much less sensitive measure than the AUC results reported in table 2. However, the results concur with the explanation that the size of the data sets has an effect on model performance, since the only significant diffference between the models is for the smallest data set, UNIX, which is composed of only 38 respondants.

6 Discussion

The results of the experiments clearly suggest that the predictive performance of item-to-item models is generally as good, or superior to the well known IRT model that contains a single latent skill to predict performance. Even the simplest of the item-to-item model (NB) performs as well or better than IRT on the AUC scores. However, the accuracy scores show smaller differences than the AUC scores across models.

The improvement over IRT varies considerably between data sets and appear sensitive to sample size, with small samples favoring the Bayesian approaches over IRT. The gain over IRT also coincides with the strength of inter-item correlations reported in table 1, which is to be expected since the item-to-item approach exploits these very correlations in the estimates.

The item-to-item approaches outlined in this paper can therefore offer a valid alternative to an IRT approach, especially for small samples where the item-to-item models appear to outperform IRT. Under this approach, estimating the chances of success to a single item requires building a classifier from a chosen subset of a few observed items. Assessing overall mastery involves estimating the chances of success to each item that is yet to be administred in the test. This overall assessment process can take close to one second, according to the setup we used for this experiment (a combination of non optimized code written over R and Weka and running on an single threaded process on an AMD Phenom II 2.6 GHz processor). On a multicore machine, and granted that the processing time of the simulation code can be improved, we can expect that a signle server can support testing of an averaged size class of around 50 students and more from a single server.

6.1 From Item-to-Item Models to Skills Assessment

The assessment outcome of the item-to-item approach is a set of probabilities, the probabilities that a given student will succeed each test item. Now, item outcome estimates do not constitute, in themselves, a skills assessment. Recall from the introduction that the student's assessment of skills is based on the weighted sum of all item responses, one weighted sum for each skill. This can be conceived as the dot product of the response matrix by the Q-matrix. Implicit to this approach is that the skill domain is covered by the set of items, of which only a subset is actually administered as part of the actual student assessment, and the mastery of the rest of the items is estimated based on the item-to-item model. The assumption is that the estimated probabilities of success to untested items allow for a more accurate assessment of skills. Such framework has been extensively studied by Falmagne, Doignon, and a number of colleagues [8] under the theory of Knowledge Spaces and it has given rise to a widely used commercial intelligent learning environment named ALEKS[6] and to a few academic systems [11,13]. Moreover, Heller and his colleagues [12] have devised a formal framework to define prerequisite relations between items and skills that allows a more sophisticated means of assessing skills with item mastery estimates.

6.2 Q-Matrix vs. Skills as Latent Traits

A Q-matrix is an intuitive concept that is readily understood as a weighted sum of items. Therefore we can assume any teacher or domain expert would be able to construct one without exceptional effort. However, the single latent concept

[6] `www.aleks.com`. See also [9].

IRT approach is even more simple to the extent that no other artifact like a Q-matrix is necessary to assess the single concept. The discrimination and difficulty parameters of an item indirectly determines its weight to the assessment of this concept, and yet no expert intervention is required given sufficient data. Of course, it is limited to a single concept, but multidimensional IRT models allow for a few skills to be assessed simultaneously, albeit with the aid of an expert that does an item classification that approaches the task of building a Q-matrix. So, in the end, the two approaches must involve a minimum knowledge engineering effort to handle multiple skills. Recent work by Pavlik et al. [16], by Stamper et al. [19], and by Liu [15], among others, offer some avenues to automatize the induction of Q-matrices from data, but this work is still in early stage.

However, a difference arises in the fact that the item-to-item approach offers no means to validate the Q-matrix, since item mastery prediction is entirely detached from the skills assessment. With IRT, the item fit method and the procedures used in our experiment allows some assessment of the validity of the skill assessment to predict item outcome, even if we had used a multidimensional (multi-skill) model. This is not possible with the item-to-item approach, as defined here, and it leaves open the question of how to validate the Q-matrix. However, research on automating the construction of a Q-matrix may offer interesting solutions in the future (see for eg. [2]).

References

1. Amershi, S., Conati, C.: Unsupervised and supervised machine learning in user modeling for intelligent learning environments. In: IUI '07: Proceedings of the 12th international conference on Intelligent user interfaces. pp. 72–81. ACM, New York, NY, USA (2007)
2. Ayers, E., Nugent, R., Dean, N.: A comparison of student skill knowledge estimates. In: 2nd International Conference on Educational Data mining, Cordoba, Spain. pp. 1–10 (2009)
3. Baker, F.B.: Item Response Theory Parameter Estimation Techniques. Marcel Dekker Inc., New York, NY (1992)
4. Carmona, C., Millán, E., de-la Cruz, J.L.P., Trella, M., Conejo, R.: Introducing prerequisite relations in a multi-layered bayesian student model. In: Ardissono, L., Brna, P., Mitrovic, A. (eds.) User Modeling 2005, 10th International Conference, UM 2005. pp. 347–356 (Edinburgh, Scotland, UK, July 24-29, 2005 2005)
5. Conati, C., Gertner, A., VanLehn, K.: Using Bayesian networks to manage uncertainty in student modeling. User Modeling and User-Adapted Interaction 12(4), 371–417 (2002)
6. Desmarais, M.C., Meshkinfam, P., Gagnon, M.: Learned student models with item to item knowledge structures. User Modeling and User-Adapted Interaction 16(5), 403–434 (2006)
7. Desmarais, M.C., Pu, X.: A bayesian inference adaptive testing framework and its comparison with Item Response Theory. International Journal of Artificial Intelligence in Education 15, 291–323 (2005)
8. Doignon, J.P., Falmagne, J.C.: Knowledge Spaces. Springer-Verlag, Berlin (1999)

9. Falmagne, J.C., Cosyn, E., Doignon, J.P., Thiéry, N.: The assessment of knowledge, in theory and in practice. In: Missaoui, R., Schmid, J. (eds.) ICFCA. Lecture Notes in Computer Science, vol. 3874, pp. 61–79. Springer (2006)
10. Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian network classifiers. Machine Learning 29(2-3), 131–163 (1997)
11. Heller, J., Hockemeyer, C., Albert, D.: Applying competence structures for peer tutor recommendations in CSCL environments. In: Kinshuk, Looi, C., Sutinen, E., Sampson, D., Aedo, I., Uden, L., Kähkönen, E. (eds.) The 4th IEEE International Conference on Advanced Learning Technologies. pp. 1050–1051. IEEE Computer Society, Los Alamitos, CA (2004)
12. Heller, J., Steiner, C., Hockemeyer, C., Albert, D.: Competence–based knowledge structures for personalised learning. International Journal on E–Learning 5(1), 75–88 (2006)
13. Hockemeyer, C., Held, T., Albert, D.: Rath - a relational adaptive tutoring hypertext www-environment based on knowledge space theory (1997)
14. Hornik, K., Buchta, C., Hothorn, T., Meyer, D., Zeileis, A.: The RWeka package (2006)
15. Liu, C.L.: A simulation-based experience in learning structures of bayesian networks to represent how students learn composite concepts. I. J. Artificial Intelligence in Education 18(3), 237–285 (2008)
16. Pavlik, P.I., Cen, H., Koedinger, K.R.: Learning factors transfer analysis: Using learning curve analysis to automatically generate domain models. In: Barnes, T., Desmarais, M.C., Romero, C., Ventura, S. (eds.) Educational Data Mining - EDM 2009, Cordoba, Spain, July 1-3, 2009. Proceedings of the 2nd International Conference on Educational Data Mining. pp. 121–130. www.educationaldatamining.org (2009)
17. Rizopoulos, D.: ltm: An r package for latent variable modelling and item response theory analyses. Journal of Statistical Software 17(5), 1–25 (2006)
18. Sing, T., Sander, O., Beerenwinkel, N., Lengauer, T.: Rocr: visualizing classifier performance in r. Bioinformatics 21(20), 3940–3941 (2005), `http://bioinformatics.oxfordjournals.org/content/21/20/3940.abstract`
19. Stamper, J.C., Barnes, T., Croy, M.J.: Extracting student models for intelligent tutoring systems. In: AAAI 2007. pp. 1900–1901. AAAI Press (2007)
20. Tatsuoka, K.K.: Rule space: An approach for dealing with misconceptions based on item response theory. Journal of Educational Measurement 20, 345–354 (1983)
21. van der Linden, W.J., Hambleton, R.K. (eds.): Handbook of Modern Item Response Theory. Springer-Verlag (1997)
22. VanLehn, K., Niu, Z., Siler, S., Gertner, A.S.: Student modeling from conventional test data: A Bayesian approach without priors. In: ITS'98: Proceedings of the 4th International Conference on Intelligent Tutoring Systems. pp. 434–443. Springer-Verlag, London, UK (1998)
23. Vomlel, J.: Bayesian networks in educational testing. International Journal of Uncertainty, Fuzziness and Knowledge Based Systems 12, 83–100 (2004)
24. Witten, I.H., Frank, E.: Data mining. Morgan Kaufmann, Los Altos, US (2000)