

IRT Modeling of Tutor Performance To Predict End-of-year Exam Scores

Elizabeth Ayers and Brian Junker
Carnegie Mellon University, Pittsburgh, Pennsylvania, United States

Abstract

Interest in end-of-year accountability exams has increased dramatically since the passing of the NCLB law in 2001. With this increased interest comes a desire to use student data collected throughout the year to estimate student proficiency and predict how well they will perform on end-of-year exams. In this paper we use student performance on the Assistment System, an on-line mathematics tutor, to show that replacing percent correct with an Item Response Theory (IRT) estimate of student proficiency leads to better fitting prediction models. In addition, other tutor performance metrics are used to further increase prediction accuracy. Finally we calculate prediction error bounds to attain an absolute measure to which our models can be compared.

Keywords: Cognitive modeling, Bayesian inference, intelligent tutoring systems, item response theory, reliability

1 Introduction

With the recent push in standardized testing in the United States, there has been an increased interest in predicting student performance on end-of-year exams from work done throughout the year (Olson, 2005). This has led to an increase in formative assessment and a growth of companies that provide assessment and prediction services¹. When predicting end-of-year exam performance, one of the most commonly used sources of student work is benchmark exams. Benchmark exams are typically paper and pencil exams given periodically throughout the year to give teachers a snapshot of student knowledge at that time. A popular measure of student understanding for many

¹For example, Pearson (<http://www.pearson.com/>) and 4Sight (www.cddre.org/Services/4Sight.cfm).

researchers is percent or number correct (e.g., Nuthall and Alton-Lee, 1995; Maccini and Hughes, 2000). Many popular prediction methods use a simple percent correct or number of correct problems on the exams as a factor in prediction models (Bishop, 1998; Haist et al., 2003). This leads to linear prediction models of the form

$$Z_i = \lambda_0 + \lambda_1 \cdot \bar{X}_i + \sum_{m=2}^M \lambda_m \cdot Y_{im} + \epsilon_i, \quad (1)$$

where Z_i is student i 's score on the end-of-year exam, \bar{X}_i is the percent (or fraction) correct on the benchmark exam, and Y_{im} are other variables used in the regression such as subject or school level background variables and other measures of performance. However, one drawback of this method is that it does not take into account the difficulty of the problems. For example, if two students see 10 different problems and both correctly answer seven, we should be cautious about using percent (or number) correct to compare the students. If one set of problems is much harder than the other, then there is an obvious difference of abilities.

As a solution to this problem, one can use Item Response Theory (IRT; e.g. van der Linden and Hambleton, 1997) which relates student and problem characteristics to item responses. By separating the problem difficulty from student ability, we can estimate the student's true underlying ability no matter what set of problems they may see. One of the simplest IRT models is the Rasch Model (Fischer and Molenaar, 1995), which models student i 's dichotomous response (0 = wrong, 1 = correct) to problem j , X_{ij} , in terms of student proficiency (θ_i) and problem difficulty (β_j) as

$$P_j(\theta_i) = P(X_{ij} = 1 | \theta_i, \beta_j) = \frac{1}{1 + e^{-(\theta_i - \beta_j)}}. \quad (2)$$

When two students take different benchmark tests, the test characteristic functions (the average of the probabilities in Equation 2, $\bar{P}(\theta) = \frac{1}{J} \sum_{j=1}^J P_j(\theta_i)$) will be different, depending on the difficulty of

the items in the two tests. Then the MLE $\hat{\theta} = \bar{P}^{-1}(\bar{X})$ will automatically adjust estimated proficiency for the differing difficulty of the items on the two benchmark tests, even if \bar{X} is the same for both students. Thus, the IRT estimate of student proficiency is scaled according to the difficulty of the problems that the student saw.

One could then replace percent correct in Equation 1 with the estimated student proficiency to obtain

$$Z_i = \lambda_0 + \lambda_1 \cdot \theta_i + \sum_{m=2}^M \lambda_m \cdot Y_{im} + \epsilon_i, \quad (3)$$

where Z_i and Y_{im} are the same as in Equation 1 and θ_i is student i 's estimated IRT proficiency. This approach is similar to the IRT-based errors-in-variables regression model used by Schofield, Taylor, and Junker (2006) in public policy.

A potentially major source of prediction error in Equation 3 is the measurement error in estimating θ_i . Finding the IRT model that best estimates θ_i is a matter of finding the trade-off between better fit (which tends to reduce statistical bias) and the complexity (which tends to increase statistical uncertainty) of the model. For example, the many individual problem difficulty parameters in the Rasch Model tend to enhance model fit while adding to the complexity of the model. On the other hand, if we know what skills are involved in the problems we can model problem difficulty in terms of the skills, as in the Linear Logistic Test Model (LLTM; Fischer, 1974). This typically reduces the number of parameters (the complexity) at the expense of decreasing the fit of the model. By improving the fit-complexity trade-off we can make more accurate predictions (e.g. lower mean squared error) of end-of-year (accountability) exam scores from benchmark testing data.

In this paper we illustrate the steps described above using data from an on-line Mathematics

Tutor know as the Assistent System (Heffernan et al., 2001; Junker, 2006). During the 2004–2005 school year, over 900 eighth-grade students in Massachusetts used the tutor to prepare for the Massachusetts Comprehensive Assessment System (MCAS) Exam. The MCAS exam is part of the accountability system that Massachusetts uses to evaluate schools and satisfy the requirements of the 2001 NCLB law². In this analysis the benchmark exams are the unique set of tutor problems that each student received and the other variables, Y_{im} , in Equation 3 are other manifest measures of student performance such as number of hints asked for and time spent answering problems.

We will also compare the prediction models we construct to one another. To compare models we computed the 10-fold cross-validation mean absolute prediction error or the mean absolute deviation,

$$MAD = \text{mean } |Z_i - \text{predicted } Z_i| = \frac{1}{N} \sum_{i=1}^N |Z_i - \text{predicted } Z_i|. \quad (4)$$

MAD is used because it is considered to be more interpretable by the Assistent developers. We also report the cross-validation mean squared error (MSE).

In Equation 3, there are many different variables Y_{im} that can be used and many different choices of IRT models to estimate student proficiency. By comparing prediction error of these models we can tell when one model is doing better than another, but we can not tell whether any one model is doing well or poorly in an absolute sense. We will use classical test theory (Lord and Novick, 1968) to obtain approximate best case bounds on the prediction error in terms of the reliabilities of the individual benchmark tests taken by the students. This gives us an absolute criterion against which to compare the prediction error of various models. If the prediction error of a model is larger

²See more at <http://www.doe.mass.edu/mcas>.

than the upper bound we know to throw out the model and search for a better one.

The MCAS exam score that we predict is the 0 to 54 raw number-right score on the exam, instead of the 200 to 280 scaled reporting score. Each year the reporting scale is recalculated by first using an standard-setting procedure to set the achievement levels in terms of raw number right and then using piece-wise linear transformations to turn the number-right scores into values within the 200 to 280 range. This second step is done such that the reporting scale achievement level cut-points remain the same from year to year (Rothman, 2001).

The study and data which this paper uses are described in Section 2. In Section 3 we describe the statistical methods used to model student proficiency and summarize the results. We then present a random effects model in Section 3.4 in an attempt to find a better fitting model. In Section 4 we use classical test theory to discuss how well we expect to do with predictions. In Section 5 we look at several MCAS exam score prediction models and compare results. Finally we offer some overall conclusions in Section 6.

2 The Assistentment Project

2.1 Design

During the 2004–2005 school year, over 900 eighth-grade students in Massachusetts used the Assistentment System. Eight teachers from two different middle schools participated, with students using the System for 20–40 minutes every two weeks. There were almost 400 main questions in the As-

sistment System which were randomly given to students. The pool of main questions was restricted in various ways, for example by the rate at which questions in different topic areas were developed for the tutor by the Assistments Project team and by teachers' needs to restrict the pool to topics aligned with current instruction. Thus, coverage of topics was not uniform, and students might see the same Assistment tasks more than once.

2.2 Data

Students using the Assistment System are presented with problems that are either previously released MCAS exam questions or that are *prima facie* equivalent “morphs”³ of released MCAS exam questions; these are called “main questions”. If students correctly answer a main question, they move onto another main question. If students incorrectly answer the main question, they are required to complete scaffolding questions which break the problem down into simpler steps. Students may make only one attempt on the main question each time that it is presented, but may take as many attempts as needed for each of the scaffolds. Students may also ask for hints if they get stuck in answering a question.

The analysis in this paper includes only those students who have MCAS exam scores recorded in the database. This narrows the sample size to a total of 683 students. Previously Farooque and Junker (2005) found evidence that skills behave differently in Assistment main questions and scaffolds. Since we want to make comparisons to the MCAS exam, the only Assistment data that is used in the IRT models is performance (correct/incorrect) on Assistment main questions. There

³In other contexts, e.g. Embretson (1999), item morphs are called “item clones”.

are a total of 354 different main questions seen by the above students.

To model problem difficulty, we need a measure of what skills problems do and do not depend upon. We can break the problems down into individual mathematics skills and record the dependencies between problems and skills. These dependencies can then be assembled into a transfer model. The transfer model, also referred to as a Q -matrix (Embretson, 1984; Tatsuoka, 1995; cf. Barnes, 2003 for a recent, more-elaborate application in intelligent tutoring) or skill coding, is a matrix

$$Q = \begin{bmatrix} q_{1,1} & q_{1,2} & \cdots & q_{1,K} \\ \vdots & \ddots & & \vdots \\ q_{J,1} & q_{J,2} & \cdots & q_{J,K} \end{bmatrix},$$

where $q_{jk} = 1$ if problem j contains skill k and 0 if it does not. Thus, the transfer model simply indicates which skills each problem depends on.

We focus our attention on a single transfer model developed in 2005 at Worcester Polytechnic Institute specifically for this project based on analysis of publicly released MCAS questions over the previous eight years. This transfer model has been designed to be a refinement of the 5 strands and 39 learning standards on which the MCAS design is based. The current version of this model contains a total of 106 skills (Pardos et al., 2006), 77 of which appear on the Assistentment main questions included in this analysis.

3 Comparing IRT Models

Since performance on any particular problem depends on both student proficiency and problem difficulty, we will use IRT models to factor out student proficiency and directly model problem difficulty. MCAS multiple choice questions are scaled (Massachusetts Dept of Education, 2004) for operational use with the 3-Parameter Logistic (3PL) model and short answer questions are scaled using the 2-Parameter Logistic (2PL) model from IRT (van der Linden and Hambleton, 1997). We know that Assistentment main questions are built to parallel MCAS exam questions and so it might be reasonable to model Assistentment main questions using the same IRT models. However, for simplicity the Rasch model, Equation 2, was used. There is evidence that student proficiencies and problem difficulties have similar estimates under the 3PL and the Rasch model (Wright, 1995) and so we are not losing much information by starting with the Rasch model. Note that in the Rasch model, the difficulty parameters β_j are not constrained in any way.

3.1 Fitting the IRT Models

As briefly mentioned in Section 1, the many individual problem difficulty parameters in the Rasch model tend to enhance model fit while increasing the complexity the model. At the expense of decreased model fit, we can reduce the number of parameters in estimating θ_i by using the Linear Logistic Test Model (LLTM; Fischer, 1974),

$$P_j(\theta_i) = P(X_{ij} = 1|\theta_i, \alpha_k) = \frac{1}{1 + e^{-(\theta_i - \sum_{k=1}^K q_{jk}\alpha_k)}}, \quad (5)$$

which constrains the Rasch problem difficulty parameters, accounting for skills in the the transfer model in modeling problem difficulty. In the LLTM, it is assumed that skill requirements for each problem combine additively to influence Rasch model problem difficulty. The reader may note that we have not included the normalization constant c in our representation of the LLTM. This decision will be explained at the end of this section when we discuss the priors used for estimation. In Equation 5, θ_i is again the proficiency of student i . Here K is the total number of skills in the transfer model being used and the q_{jk} are the entries of that transfer model. Thus, β_j , in Equation 2, is now a linear combination of the skills that appear in problem j . The α_k represents the difficulty of skill k and, similar to problem difficulties, higher values of α indicate harder skills. When there are fewer skills K than test problems J , the LLTM is a restricted form of the Rasch model: for example, if the transfer model Q is the $J \times J$ identity matrix, we obtain the unrestricted Rasch model again. The LLTM has been successful in other works such as van de Vijver (1988) and De Boeck and Wilson (2004). In both of these cases, the correlation between the Rasch model and LLTM problem difficulties was greater than 0.90.

In our analysis we consider $N = 683$ students' dichotomous answers to up to $J = 354$ Assistentment main questions. There are many missing values since no student saw all of the problems. We treat these missing values as missing completely at random (MCAR) since problems were assigned to students randomly by the Assistentments software from a "curriculum" of possible questions designed for all students by their teachers in collaboration with project investigators.

The dichotomous responses X_{ij} are modeled as Bernoulli trials,

$$X_{ij} \sim \text{Bern}(P_j(\theta_i)) \quad i = 1, \dots, N; \quad j = 1, \dots, J,$$

where $P_j(\theta_i)$ is given as above by Equation 2 or Equation 5. Under the usual IRT assumption of independence between students and between responses, given the model parameters, the complete data likelihood can be written as

$$P(\underline{X} = \underline{x}) = \prod_{i=1}^N \prod_{j:i \text{ saw } j} P_j(\theta_i)^{x_{ij}} [1 - P_j(\theta_i)]^{1-x_{ij}}. \quad (6)$$

We estimated the student proficiency (θ_i) and problem difficulty (β_j) parameters in the Rasch model and the student proficiency (θ_i) and skill difficulty (α_k) parameters in the LLTM, using Markov Chain Monte Carlo methods with the program WinBUGS⁴ (Bayesian inference Using Gibbs Sampling; Spiegelhalter et al., 2003). The Rasch model, Equations 2 and 6, was estimated using the data with the priors $\theta_i \sim N(\mu_\theta, \sigma_\theta^2)$ and $\beta_j \sim N(\mu_\beta, \sigma_\beta^2)$. We placed a weak Normal hyperprior on μ_β and a weak Inverse-Gamma hyperprior on σ_β^2 . In item response models, the location and scale of the latent variable, and hence of problem difficulty parameters, are not fully identified, which can undermine comparisons between fits on different data sets. We decided to fix the (prior) mean and variance of the student proficiency (θ) to be 0.69 and 0.758. These values were found by preliminary analysis using weak hyperpriors on these parameters. In the LLTM, which constrains the Rasch model as shown by Equation 5, we used the same Normal(0.69,0.758) prior on the student proficiency (θ) and the prior on α was Normal($\mu_\alpha, \sigma_\alpha^2$). Again, a vague Normal hyperprior was placed on μ_α and a vague Inverse-Gamma on σ_α^2 . This method differs slightly from other estimation methods that use a $N(0, 1)$ prior for θ_i . In our case, one can think of the prior on θ taking the place of the normalization constant c . All estimates mentioned below refer to the posterior means of the parameters.

⁴WinBUGS and R code available from the authors on request.

Table 1: Rasch vs LLTM fits for Assisment Main Questions

Model	$-2 \cdot l_M = \text{Deviance}$	Parameters	BIC
LLTM	56090	79	~ 56605
Rasch	47640	356	~ 49963
		Difference in BIC	~ 6600

3.2 Direct Model Comparison

We compared the Rasch model and LLTM using Bayesian Information Criterion (BIC; Raftery, 1995) scores,

$$-2 \cdot l_M + k \cdot \log(n). \quad (7)$$

Here l_M is the log-likelihood of the model, k is the number of free parameters to be estimated, and n is the sample size (here, the number of students). In this version of BIC scores, lower values indicate a better fitting model and a difference of as small as 2 denotes a difference between models and differences larger than 10 denote a very strong significant difference between the models.

As discussed in Section 1, in testing the LLTM restrictions as imposed by the transfer model, the comparison of the LLTM and Rasch model can be construed in the fit vs. complexity framework. BIC scores do this with the fit being measured by the first term and complexity by the second in Equation 7. WinBUGS tracks the deviance, which is defined as $-2 \cdot l_M$ (Spiegelhalter, Thomas, and Best, 2003), of the model during estimation runs. Table 1 shows the BIC scores for both the Rasch model and LLTM. One can see that the difference in BIC scores is ~ 6600 and thus the Rasch model is overwhelmingly favored. Although the Rasch model is more complex (in the number of parameters) than the LLTM, the dramatically better fit of the Rasch model makes up for the complexity and the Rasch model is strongly favored.

3.3 Diagnosing the Misfit

The posterior estimates of the problem difficulties for Assistent main questions can be seen in Figure 1 (a). Rasch model estimates of problem difficulty are on the vertical axis and LLTM estimates are on the horizontal axis. The estimates are number coded by the number of skills in the problem. In general, the problem difficulty increases as the number of skills increases. Note that the majority of the LLTM problem difficulties (horizontal axis) lie between -2 and 2 . However, the Rasch problem difficulties (vertical axis) range fully from -4 to 4 . This illustrates how the LLTM really does constrain the Rasch problem difficulties.

For a closer look at the fit of the IRT models to the data and transfer model the per problem standardized residuals,

$$r_j = \frac{n_j - E(n_j)}{\sqrt{\widehat{Var}(n_j)}}, \quad (8)$$

were analyzed. Here, $n_j = \sum_{i:i \text{ saw } j} X_{ij}$ is the number of correct answers to problem j , $E[n_j]$ is its expected value estimated from fitting the model in Equation 2 or 5, and $\widehat{Var}(n_j)$ is its variance estimated from the same model. Since these residuals are standardized, we expect the majority to fall between -3 and 3 . Figure 1 (b) shows the Rasch versus LLTM residuals, as described in Equation 8. Similar to problem difficulty estimates, the residuals are number coded by the number of skills in the problem. For these Assistent main questions, the Rasch model residuals (vertical axis) range from -0.6 to 1.4 , indicating good fit, and the LLTM residuals (horizontal axis) range from -23 to 11.2 , indicating bad fit.

We also calculated the per problem outfit statistics (van der Linden and Hambleton, 1997, page

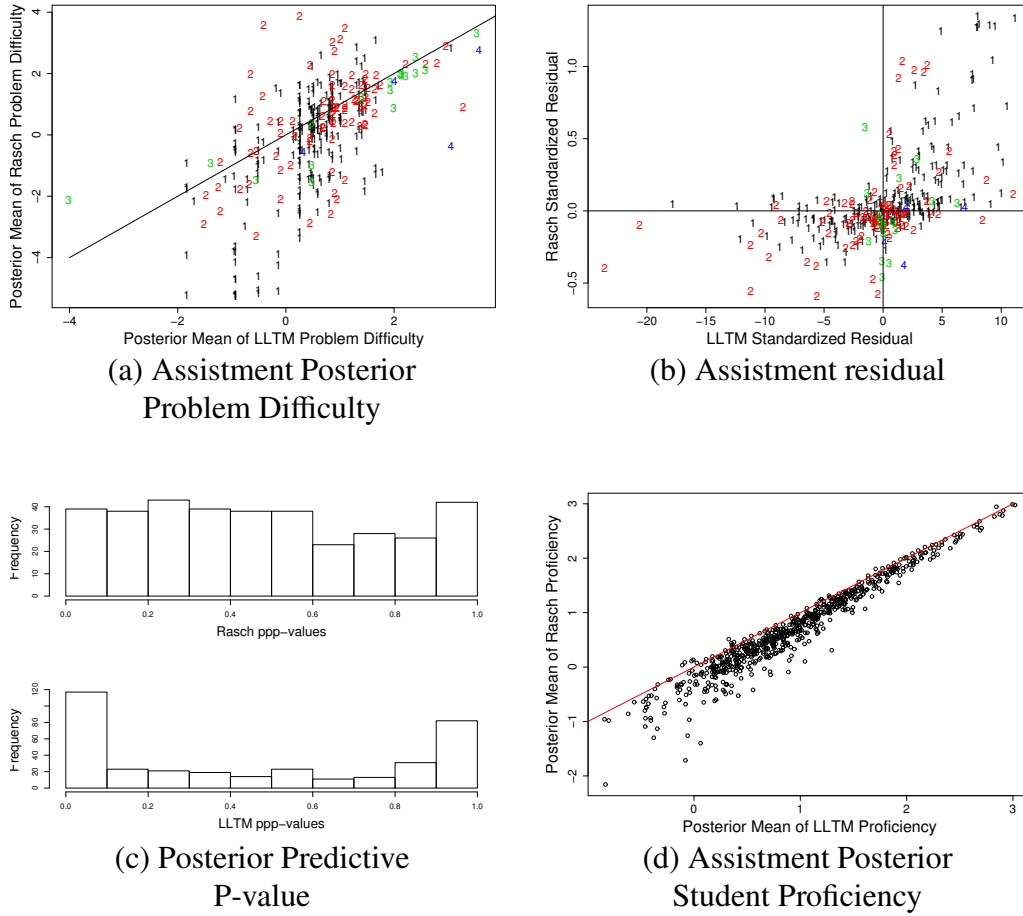


Figure 1: Assistent Main Question Posterior Problem Difficulty Estimates, Residuals, ppp-values, and Posterior Student Proficiency Estimates, Graphs are Number Coded by the Number of Skills in the Problem

113),

$$T_j(x | \phi) = \sum_{i=1}^{N_j} \frac{(x_{ij} - E_{ij})^2}{N_j W_{ij}},$$

where N_j is the number of students that saw problem j , x_{ij} is student i 's response on problem j , E_{ij} is the expected value of X_{ij} conditional on the parameter vector ϕ , and W_{ij} is the variance of X_{ij} also conditional on ϕ . To check the per problem fit of each model, the posterior predictive p-value (ppp-value; Gelman et al., 2004), the expected value of the classical p-value over the posterior

distribution of the parameter vector given the model and the observed data, was estimated using

$$p_i \approx \frac{\#\{s : T_i(x | \phi_x) < T_i(x^* | \phi_x); s = 1, 2, \dots, M\}}{M},$$

which compares the observed values of the test statistic to values of the test statistic for data simulated from the model. For this calculation, the simulated data (x^*) was obtained by using the Markov Chain given by WinBUGS. Similar to classical p-values, there is reason to question the fit of the model to problem i if p_i is small. A weakness of the ppp-value is that it uses the data twice, once to calculate the observed test statistics and again to simulate data to calculate the ppp-value. One consequence of this is that ppp-values are not uniformly distributed and tend to be conservative (Gelman et al, 1996, page 790). However, we can still expect the ppp-values to aggregate around zero if there is serious misfit for some of the problems.

Figure 1 (c) shows the histograms for the Rasch model and LLTM ppp-values. We see that the Rasch model ppp-values are roughly uniform, which we would expect if the model fit is acceptable. For the LLTM, the grouping of ppp-values around 1 shows the weakness of ppp-values to be bias toward accepting the model. However, there are also many ppp-values concentrated at 0 giving the stronger impression of misfit of the model. Finally, Figure 1 (d) displays the estimated student proficiencies. The LLTM (horizontal axis) estimates higher student proficiencies than the Rasch model (vertical axis).

3.4 Considering a Random Effects LLTM

A striking feature of Figure 1 (a) is the pattern of vertical lines evident in the scatter plot. These vertical lines indicate groups of problems tagged with the same skills in the LLTM, and hence

receiving a single, common difficulty estimate. On the other hand the unconstrained Rasch model estimates very different difficulties for these problems.

It may be that the LLTM is essentially the right model, but there is some variation in the difficulty of problems that depend on the same target skills, due to nuisance influences on performance factors that are unique to each problem. One way of modeling this is to add a random effects component, similar to Janssen and De Boeck (2006), to each problem difficulty in the LLTM. The model for problem difficulty then becomes

$$\beta_j = \sum_{k=1}^K q_{jk} \cdot \alpha_k + \epsilon_j$$

where

$$\epsilon_j \sim N(0, \sigma_\beta^2).$$

Our initial attempts to fit this model led to a random effect so large that skills were no longer playing a significant role in modeling problem difficulty. We performed a sensitivity analysis to see the effect of various fixed values of σ_β^2 (the LLTM itself corresponds to $\sigma_\beta^2 = 0$) on our estimates of skill difficulties (α'_k 's). We were interested in finding the point at which the random effects component overtook the skills and accounted for the majority of the difficulty in the problems, as indicated by a reduced number of skills whose difficulty estimates were significantly different from zero. Given values of σ_β^2 equal to 0.01, 0.1, 0.2, 0.5, and 1, we found, respectively, 60, 52, 47, 35, and 16 skills with difficulty estimates significantly different from zero. One can see that σ_β^2 does not need to be very large before there are few significant skills. When we tried the same hyperprior on σ_β^2 as Janssen and de Boeck (2006) did, the posterior estimate of σ_β^2 was 2.37 and, as expected from our sensitivity analysis, there were only 4 skill difficulties significantly different

from zero. The lack of many skills with difficulty estimates significantly different from zero is an indication of a misalignment between the skills in our model and the difficulty of the problems

4 Reliability and Predictive Accuracy

Before exploring the predictive accuracy of our models using the mean absolute deviation (MAD) measure defined in Equation 4, it is important to ask how well Assistentment scores could predict MCAS scores under ideal circumstances. Let us begin by assuming the MCAS exam and the Assistentment System to be two parallel tests of the same underlying construct. Following classical test theory (Lord and Novick, 1968) we have

$$X_{i1} = T_i + \epsilon_{i1}$$

$$X_{i2} = T_i + \epsilon_{i2}$$

where the true score of student i is T_i , X_{it} is student i 's observed score on test t , and ϵ_{it} is the error on test t . We have followed the usual assumptions that the expected value of the error terms are zero, the error terms are uncorrelated, and that the error terms and the true score are uncorrelated.

The expected mean square error (MSE) between the tests is then

$$E[(X_{i1} - X_{i2})^2] = E[(\epsilon_{i1} - \epsilon_{i2})^2] = \sigma_{\epsilon_1}^2 + \sigma_{\epsilon_2}^2.$$

Since the reliability of test t ($t = 1$ or 2) is defined as

$$r_t = \frac{\sigma_T^2}{\sigma_{X_t}^2} = \frac{\sigma_T^2}{\sigma_T^2 + \sigma_{\epsilon_t}^2}, \quad (9)$$

some algebra then shows that the root mean square error (RMSE) is

$$RMSE = \sqrt{E[(X_{i1} - X_{i2})^2]} = \sigma_T \sqrt{\left(\frac{r_1 + r_2}{r_1 \cdot r_2} - 2\right)}.$$

This can be converted into lower and upper bounds on the MAD score as follows. Using the Cauchy-Schwarz Inequality for Euclidean spaces (Protter and Morrey, 1991, page 130) with $x_i = |\text{MCAS}_i - \text{predicted MCAS}_i|$ and $y_i = 1$,

$$\sum_{i=1}^N |\text{MCAS}_i - \text{predicted MCAS}_i| \leq \sqrt{N} \cdot \sqrt{\sum_{i=1}^N (\text{MCAS}_i - \text{predicted MCAS}_i)^2}.$$

We can then scale both sides by $\frac{1}{N}$ to achieve

$$MAD \leq RMSE.$$

We can also bound the MAD from below. First, let $x_i = \text{MCAS}_i - \text{predicted MCAS}_i$ and $|x_{\max}|$ denote the absolute maximum deviation between the true and predicted MCAS scores. Then,

$$RMSE^2 = \frac{1}{n} \sum_{i=1}^N x_i^2 \leq \frac{1}{n} \sum_{i=1}^N |x_i| \cdot |x_{\max}| = |x_{\max}| \frac{1}{n} \sum_{i=1}^N |x_i| = |x_{\max}| MAD,$$

so we have that

$$\frac{1}{|x_{\max}|} \cdot RMSE^2 \leq MAD.$$

Thus, our lower and upper bounds for the MAD score are

$$\frac{1}{|x_{\max}|} \cdot RMSE^2 \leq MAD \leq RMSE. \quad (10)$$

From Equation 9, we have that $\sigma_T^2 = r_t \cdot \sigma_X^2$. In the most recent technical report published (Massachusetts Dept of Education, 2004) the MCAS has listed $r_{t=1} = 0.9190$ and $\sigma_X^2 = 142.39$, so that in predicting MCAS exam scores from Assistent scores we have

$$RMSE = \sqrt{130.86 \cdot \left(\frac{0.9190 + r_2}{0.9190 \cdot r_2} - 2\right)}, \quad (11)$$

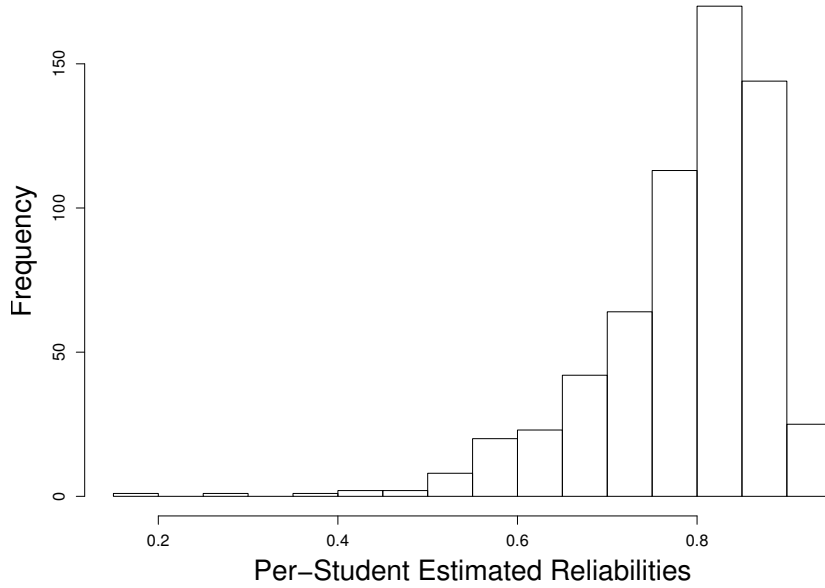


Figure 2: Histogram of per-student Assistent reliabilities as given by Equation 12

where r_2 is the reliability of the Assistent score.

However, since each student completes a unique set of Assistent main questions, we could not calculate r_2 directly. Instead, we calculated reliability separately for each student. For this purpose we considered a reduced dataset of 616 students who had 10 or more problems completed for which all pairs of correlations were available. To estimate the per-student reliability, we used Cronbach's alpha coefficient (Cronbach, 1951),

$$\alpha_i = \frac{n_i \bar{r}_i}{1 + (n_i - 1) \bar{r}_i}. \quad (12)$$

In Equation 12, n_i is the number of problems that student i saw and \bar{r}_i is the average inter-item correlation for problems seen by student i . Once the per-student reliabilities were calculated, the per-student estimated RMSE values were computed using Equation 11. Figure 2 shows the estimated reliabilities for the students who met the criteria explained above. It is interesting to note

that the estimated RMSE is never lower than 4.44.

In order to have a single approximate set of approximate bounds for the MAD score in Equation 10, we found the median Assistent reliability, 0.8080, and the corresponding RMSE of 6.529 from Equation 11. The largest deviation, $|x_{max}|$, between the true and predicted MCAS scores among the models in Table 2 below was 40.5. Substituting these values for RMSE and $|x_{max}|$ into Equation 10 we find the approximate bounds,

$$1.053 \leq MAD \leq 6.529.$$

5 MCAS Exam Score Prediction

Student proficiencies estimated from a successful IRT model are combined with other Assistent performance metrics to produce an effective prediction function, following the work of Anozie and Junker (2006), using an errors-in-variables regression approach similar to that of Schofield et al. (2005). The linear model is

$$MCAS_i = \lambda_0 + \lambda_1 \cdot \theta_i + \sum_{m=2}^M \lambda_m \cdot Y_{im} + \epsilon_i,$$

where θ_i is the proficiency of student i as estimated by the IRT model and Y_{im} is performance of student i on manifest measure m . WinBUGS was again used to find Bayesian estimates of the linear regression coefficients. When estimating each of the following models, the IRT item parameters were fixed at their estimates from Section 3, but student proficiency was re-estimated. It is logical to not re-estimate item parameters since this is how MCAS prediction would occur in practice: problems are fixed but student proficiencies are changing throughout the year and year-to-year.

To compare the prediction models we calculated the 10-fold cross-validation (CV) MAD score (Equation 4). In K-fold CV, the dataset is randomly divided into K subsets of approximately equal size. One subset is omitted (referred to as the testing set) and the remaining K-1 subsets (referred to as the training set) are used to fit the model. The fitted model is then used to predict the MCAS exam scores for the testing set. The desired statistic, in this case the MAD score, is then calculated for the testing set. The process is repeated K times (the folds) with each of the K subsets being used exactly once as the testing set. The K results from the folds are then averaged to produce a single estimation of the MAD score. By using cross-validation we avoid using the data to both fit the model and give an estimate of the fit. In addition we also report the 10-fold cross-validation root sample mean square error,

$$RMSE = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^N (\text{MCAS}_i - \text{predicted MCAS}_i)^2},$$

of the models.

Table 2 shows results from several prediction models. In view of the results of Section 3 and Section 3.4 above, the IRT based prediction models below are based on the Rasch model. Column 2 lists which variables are in the model (for a full list and description of the variables used in each model see Table 3). Column 3 simply states the number of variables in the model. Columns 4 and 5 give the CV MAD score and the CV RMSE respectively. Column 6 offers any important notes about the model. Historically, and in particular within the Assistent Project, percent correct on main questions has been used as a proxy for student ability. To see if any information was gained by simply using the Rasch estimate of student proficiency, we compared the two models with only these variables. Model 1 is the simple linear regression using only percent correct on main questions and has a MAD score of 7.18. Model 2 uses only the Rasch student proficiency and gives

Table 2: Prediction Models

Model	Variables	# of Vars	CV MAD	CV RMSE	Notes
Model 1	Percent Correct on main questions	1	7.18	8.65	
Model 2	Rasch student proficiency	1	5.90	7.18	
Model 3 (Anozie & Junker, 2006)	Percent Correct on main questions and 4 other manifest performance metrics	35	5.46	7.00	uses multiple monthly summaries
Model 4	Rasch student proficiency and same 4 manifest performance measures as Model 3	5	5.39	6.56	uses only year-end aggregates optimized for student proficiency
Model 5	Rasch student proficiency and 5 manifest performance measures (one overlap with models 3 & 4)	6	5.24	6.46	

a MAD score of 5.90. By simply using IRT to account for problem difficulty in estimating student proficiency, we can drop the MAD score a full point. Accounting for problem difficulty gives a more efficient estimate of how well a student is doing and leads to better predictions. Model 3, from Anozie and Junker (2006), uses as predictors monthly summaries from October to April for percent correct on main questions and four other manifest measures of student performance. Model 4 uses the year-end aggregates of the same variables and substitutes Rasch student proficiency for percent correct on main questions. We see that Model 4 gives a slightly lower MAD score. Thus by using Rasch student proficiency (in place of percent correct on main questions) we can use fewer measures of student performance on Assisments.

Model 5 was optimized (for MAD score) for Rasch student proficiency and year end aggregates of student performance measures using backwards variable selection implemented in WinBUGS

Table 3: Definitions of Variables used in Prediction Models

Variable Name	Model	Definition
Student Proficiency	2, 4, 5	IRT estimate of student Proficiency
PctCorMain	1, 3	Percent of correctly answered main questions
PctCorScaf	3, 4	Percent of correctly answered scaffolds
SecIncScaf	3, 4	Number of seconds spent answering all incorrect scaffolds
NumPmAllScaf	3, 4, 5	Number of scaffolds completed per minute
NumHintsIncMainPerMain	3, 4	$\frac{\text{number hints} + \text{number incorrect main questions}}{\text{Number of main questions attempted}}$
SecCorScaff	5	Number of seconds spent answering all correct scaffolds
SecIncMain	5	Number of seconds spent on incorrect main questions
MedSecIncMain	5	Median number of seconds per incorrect main question
PctSecIncMain	5	Percent of time on main q's spend on incorrect main q's

and R⁵ (R Development Core Team, 2004). To start we used the same 12 variables as Anozie and Junker (2006), excluding percent correct on main questions and adding Rasch student proficiency. We ran the full model and all models excluding one variable, with the caveat that student proficiency was always kept in the model. For each model, MCAS exam scores were predicted and MAD scores calculated. The model with the lowest MAD score was then used as the new “full” model. This process was repeated until removing variables from the “full” model no longer reduced the MAD score. The final model, which contained student proficiency and five manifest measures of student performance, gives a MAD score of 5.24, a slight improvement from Model 4. Overall, the ability to use fewer variables makes the effort expended in estimating the IRT models worth it.

The regression equation for Model 2 is

$$MCAS_i = 18.289 + 10.425 \cdot (\text{Rasch student proficiency}). \quad (13)$$

From this we see that there is a baseline MCAS exam score prediction of 18 points and for each

⁵WinBUGS and R code available from the authors on request.

additional unit of estimated Rasch student proficiency we add 10.425 to to the exam score prediction. As a student's proficiency increases, so does their exam score prediction. The regression equation for Model 5 is

$$\begin{aligned}
 MCAS_i = & 8.514 + 10.336 \cdot (\text{Rasch student proficiency}) + 8.928 \cdot (\text{NumPmAllScaf}) \\
 & + 0.004 \cdot (\text{SecCorScaff}) + 0.032 \cdot (\text{MedSecIncMain}) - 0.001 \cdot (\text{SecIncMain}) \\
 & - 2.696 \cdot (\text{PctSecIncMain}).
 \end{aligned} \tag{14}$$

In Equation 14 the increase in MCAS score for each unit of increase in Rasch proficiency is about the same as in Equation 13. However, the baseline of 18.289 has been decomposed into a new baseline of about 8.5 points, incremented or decremented according to various measurements of response efficiency. The largest increment, 8.928, comes from the rate at which scaffolding questions are completed and the largest decrement, 2.696, comes from time spent on answering main questions incorrectly.

Now that we have compared models to one another, we need to compare the models to the bounds calculated in Section 4. Recall from Section 4 that we have a bound of

$$1.053 \leq MAD \leq 6.529.$$

From Table 2, one can see that Model 5 has a MAD score of 5.24, which is well below the upper bound.

Moreover, the RMSE reported for Model 5, 6.46, is similar to our estimated optimal RMSE of 6.53. It should also be noted, that with a perfect Assistent reliability in Equation 11, the estimated

RMSE would be 5.576 and the bound would be

$$0.768 \leq MAD \leq 5.576.$$

Again, the MAD score of Model 5 is below this upper bound. Using a split-half reliability calculation on the MCAS exam itself, Feng et al. (2006a) found an average MAD score of 5.94. Since we are achieving MAD scores less than this and the two previously mentioned upper bounds, we do not expect to do much better without an increase in the reliability of the MCAS exam.

6 Discussion

In this article, we have developed a framework to create prediction functions for end-of-year exam scores using an IRT estimate of student ability based on work done throughout the school year. Although this framework was illustrated using data from an online mathematics tutor, other benchmark work, such as homework or paper and pencil exams, could be used to predict end-of-year exam scores as well.

In addition to developing this general framework, our research generated two additional findings. First, prediction using IRT scores is more effective than prediction using number correct scores. For example, our Rasch model based predictions always produced lower MAD and RMSE prediction errors than the corresponding predictions based on number correct. Moreover, the IRT-based predictions were essentially as good as one could do with parallel tests, even though our Assessment System was not constructed to be parallel (in the classical test theory sense) to the MCAS exam.

Second, in our application, the Rasch model outperformed the LLTM. This is contrary to some previous experience with the LLTM (van de Vijver, 1988; De Boeck and Wilson, 2004) and this deserves further exploration. An obvious place to look is possible improvements in the transfer model (Q-matrix) and we are beginning to examine prediction-based methods for improving the LLTM in this situation. Of course, another possible difficulty may be that the LLTM is inappropriate for this data. Alternative models involving Bayes net style cognitive diagnosis models (Anozie, 2006; Pardos et al., 2006) and multidimensional IRT models (based on the 5 MCAS strands: Number Sense and Operations; Patterns, Relations, and Algebra; Geometry; Measurement; and Data Analysis, Statistics, and Probability) are also being pursued.

Acknowledgments

This Assistentment Project was made possible by the US Dept of Education, Institute of Education Science, “Effective Mathematics Education Research” program grant #R305K03140, the Office of Naval Research grant #N00014-03-1-0221, the Spencer Foundation, and NSF CAREER award to Neil Heffernan. Additional support for Ayers was provided by The Carnegie Mellon Program for Interdisciplinary Educational Research under the IES grant #R305K3140. All the opinions in this article are those of the authors, and not those of any of the funders.

This work would not have been possible without the assistance of the 2004-2005 WPI/CMU ASSISTment Team including Nathaniel O. Anozie, Andrea Knight, Ken Koedinger, Meghan Myers, Carolyn Rose all at CMU, Steven Ritter at Carnegie Learning, Mingyu Feng, Neil Heffernan,

Tom Livak, Abraao Lourenco, Michael Macasek, Goss Nuzzo-Jones, Kai Rasmussen, Leena Razaq, Terrence Turner, Ruta Upalekar, and Jason Walonoski all at WPI.

References

- Anozie, N.O. (2006). *Investigating the utility of a conjunctive model in Q-matrix assessment using monthly student records in an online tutoring system*. Proposal submitted to the National Council on Measurement in Education 2007 Annual Meeting.
- Anozie, N.O. and Junker, B.W. (2006). Predicting end-of-year accountability assessment scores from monthly student records in an online tutoring system. *Proceedings of the American Association for Artificial Intelligence Workshop on Educational Data Mining (AAAI-06)*, July 17, 2006, Boston, MA. Menlo Park, CA: AAAI Press. pp. 1-6. Technical Report WS-06-05.
- Ayers, E. and Junker, B.W. (2006). Do skills combine additively to predict task difficulty in eighth grade mathematics? *Proceedings of the American Association for Artificial Intelligence Workshop on Educational Data Mining (AAAI-06)*, July 17, 2006, Boston, MA. Menlo Park, CA: AAAI Press. pp. 14-20. Technical Report WS-06-05.
- Barnes, T.M. (2003). *The Q-matrix Method of Fault-tolerant Teaching in Knowledge Assessment and Data Mining*. Ph.D. Dissertation, Department of Computer Science, North Carolina State University.
- Bishop, J.H. (1998). The Effect of Curriculum-Based External Exit Exam Systems on Student Achievement. *The Journal of Economic Education*, Vol 29, Issue 2, 171-182.

- Cronbach, L.J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika*, 16, 297-334.
- De Boeck, P. and Wilson, M. (2004). *Statistics for Social Science and Public Policy*. New York: Springer.
- Embretson, S.E. (1984). A General Latent Trait Model for Response Processes. *Psychometrika*, 49, 175-186.
- Embretson, S.E. (1999). Generating items during testing: psychometric issues and models. *Psychometrika*, 64, 407-433.
- Embretson, S.E. (1985). Multicomponent latent trait models for test design. In S.E. Embretson (Eds). *Test design: developments in psychology and psychometrics* (1985). New York: Academic Press.
- Farooque, P. and Junker, B.W. (2005). Behavior of Skills within MCAS and Assisment Main Problems, Final Project Poster, Dept. of Statistics, Carnegie Mellon University.
- Feng, M., Heffernan, N.T, and Koedinger, K.R. (2006). Predicting State Test Scores Better with Intelligent Tutoring Systems: Developing Metrics to Measure Assistance Required. Accepted by the 8th International Conference on Intelligent Tutoring Systems, Taiwan. <http://web.cs.wpi.edu/~mfeng/pub/its06.pdf>
- Fischer, G.H. (1974). *Einführung in die Theorie psychologischer Tests: Grundlagen und Anwendungen*. (Introduction to the Theory of Psychological Tests: Foundations and Applications) Switzerland: Verlag Hans Huber.
- Fischer, G.H. and Molenaar, I.W. (1995). *Rasch Models: Foundations, Recent Developments,*

- and Applications*. New York: Springer Verlag.
- Gelman, A., Carlin, J., Stern, H., and Rubin, Donald B. (2004). *Bayesian Data Analysis*, 2nd edn. New York: Chapman and Hall.
- Gelman, A., Meng, X.-L., and Stern, H. (1996). Posterior Predictive Assessment of model fitness via realized discrepancies [with discussion]. *Statistica Sinica*, 6, 733-807.
- Haist, S.A., Witzke, D.B., Quinlivan, S., Murphy-Spencer, A., and Wilson, J.F. (2003). Clinical Skills as Demonstrated by a Comprehensive Clinical Performance Examination: Who Performs Better - Men or Women? *Advances in Health Sciences Education*, 8: 189-199.
- Hartel, E.H. (1989). Using restricted latent class models to map the skill structure of achievement items. *Journal of Educational Measurement*, 26, 301-321.
- Heffernan, N.T., Koedinger, K.R. and Junker, B.W. (2001). Using Web-Based Cognitive Assessment Systems for Predicting Student Performance on State Exams. Research proposal to the Institute of Educational Statistics, US Department of Education. Department of Computer Science at Worcester Polytechnic Institute, Worcester County, Massachusetts. http://nth.wpi.edu/pubs_and_grants/Grant_to_IES_with_WPS.pdf
- Janssen, R. and De Boeck, P. (2006). A random-effects version of the LLTM. Technical report, Department of Psychology, University of Leuven, Belgium.
- Junker, B. W. (2006). Using on-line tutoring records to predict end-of-year exam scores: experience with the ASSISTments Project and MCAS 8th grade mathematics. To appear in Lissitz, R. W. (Ed.), *Assessing and modeling cognitive development in school: intellectual growth and standard setting*. Maple Grove, MN: JAM Press.

- Lord, F.M. and Novick, M.R. (1968). *Statistical Theories of Mental Test Scores*. Reading, MA: Addison-Wesley.
- Maccini, P. and Hughes, C.A. (2000). Effects of a problem-solving strategy on the introductory algebra performance of secondary students with learning disabilities. *Learning Disabilities Research and Practice*, 15, 10-21.
- Massachusetts Department of Education. (2004). *2004 MCAS Technical Report*. Downloaded December 2005 from <http://www.doe.mass.edu/mcas/2005/news/04techrpt.pdf>.
- Massachusetts Department of Education. (2006). School and District Accountability. Retrieved April 2006 from <http://www.doe.mass.edu/sda/>.
- Nuthall, G. and Alton-lee, A. Assessing Classroom Learning: How Students Use Their Knowledge and Experience to Answer Classroom Achievement test Questions in Science and Social Studies. *American Educational Research Journal*, Vol. 31, No. 1, 185-223.
- Olson, L. (2005). State Test Programs Mushroom as NCLB Mandate Kicks In. *Education Week*, Nov. 30: 10-14.
- Pardos, Z.A., Heffernan, N.T., Anderson, B. and Heffernan, C.L. (2006). Using Fine-Grained Skill Models to Fit Student Performance with Bayesian Networks. Workshop in Educational Data Mining held at the Eighth International Conference on Intelligent Tutoring Systems. Taiwan. 2006. <http://web.cs.wpi.edu/Research/trg/public/project/papers/its06/zpardos-its-final22.pdf>
- Protter, M.H. and Morrey, C.B. Jr. (1991). *A First Course in Real Analysis. 2nd Edition*. New York: Springer.

- R Development Core Team (2004). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org>.
- Raftery, A.E. (1995). Bayesian Model Selection in Social Research. *Sociological Methodology*, Vol. 25, 111-163.
- Rothman, S. (2001). *2001 MCAS Reporting Workshop: The second generation of MCAS results*. Massachusetts Department of Education. Downloaded January 2007 from http://www.doe.mass.edu/mcas/2001/news/reproting_wkshp.pps.
- Schofield, L., Taylor, L. and Junker, B.W. (2006). *The use of cognitive test scores in evaluating black-white wage disparity*. Working paper.
- Spiegelhalter, D.J., Thomas, A. and Best, N.G. (2003). *WinBUGS: Bayesian Inference Using Gibbs Sampling, Manual Version 1.4*. Cambridge: Medical Research Council Biostatistics Unit.
- Tatsuoka, Kikumi K. (1995). Architecture of Knowledge Structures and Cognitive Diagnosis: A Statistical Pattern Recognition and Classification Approach. Chapter 14 in P.D. Nichols, S.F. Chipman, and R.L. Brennan. (Eds). *Cognitively Diagnostic Assessment* (1995). Hillsdale, NJ: Lawrence Erlbaum Associates.
- van de Vijver, Fons J.R. (1988). Systematizing the Item Content in Test Design. Chapter 13 in R. Langeheine and J. Rost. (Eds). *Latent Trait and Latent Class Models* (1988). New York: Plenum Press.
- van der Linden, W.J. and Hambleton, R.K. (1997). *Handbook of Modern Item Response Theory*.

New York: Springer-Verlag.

Wright, B.D. (1995). 3PL or Rasch? *Rasch Measurement Transactions*, 9(1), 408-409.