To cite this chapter:

Niemeijer, K., Feskens, R., Krempl, G., Koops, J., & Brinkhuis, M. (2020). Constructing and predicting school advice for academic achievement: A comparison of item response theory and machine learning techniques. In *Proceedings of the Tenth International Conference on Learning Analytics & Knowledge* (pp. 462-471). Association for Computing Machinery. https://doi.org/10.1145/3375462.3375486



Constructing and Predicting School Advice for Academic Achievement

A Comparison of Item Response Theory and Machine Learning Techniques

Koen Niemeijer
koen.niemeijer@kuleuven.be
KU Leuven
Utrecht University
Leuven, Belgium

Remco Feskens
remco.feskens@cito.nl
Cito Institute for Educational Measurement
Arnhem, The Netherlands

Georg Krempl
g.m.krempl@uu.nl
Utrecht University
Utrecht, The Netherlands

Jesse Koops
jesse.koops@cito.nl
Cito Institute for Educational Measurement
Arnhem, The Netherlands

Matthieu J. S. Brinkhuis
m.j.s.brinkhuis@uu.nl
Utrecht University
Utrecht, The Netherlands

ABSTRACT

Educational tests can be used to estimate pupils' abilities and thereby give an indication of whether their school type is suitable for them. However, tests in education are usually conducted for each content area separately which makes it difficult to combine these results into one single school advice. To help with school advice, we provide a comparison between both domain-specific and domain-agnostic methods for predicting school types. Both use data from a pupil monitoring system in the Netherlands, a system that keeps track of pupils' educational progress over several years by a series of tests measuring multiple skills.

A domain-specific item response theory (IRT) model is calibrated from which an ability score is extracted and is subsequently plugged into a multinomial log-linear regression model. Second, we train domain-agnostic machine learning (ML) models. These are a random forest (RF) and a shallow neural network (NN). Furthermore, we apply case weighting to give extra attention to pupils who switched between school types.

When considering the performance of all pupils, RFs provided the most accurate predictions followed by NNs and IRT respectively. When only looking at the performance of pupils who switched school type, IRT performed best followed by NNs and RFs. Case weighting proved to provide a major improvement for this group. Lastly, IRT was found to be much easier to explain in comparison to the other models. Thus, while ML provided more accurate results, this comes at the cost of a lower explainability in comparison to IRT.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

LAK '20, March 23–27, 2020, Frankfurt, Germany

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-7712-6/20/03…$15.00
https://doi.org/10.1145/3375462.3375486

CCS CONCEPTS

• **Applied computing** → **E-learning**; • **Computing methodologies** → *Machine learning*.

KEYWORDS

E-Learning, Item Response Theory, Machine Learning, Neural Networks, Random Forests, Explainable AI

ACM Reference Format:
Koen Niemeijer, Remco Feskens, Georg Krempl, Jesse Koops, and Matthieu J. S. Brinkhuis. 2020. Constructing and Predicting School Advice for Academic Achievement: A Comparison of Item Response Theory and Machine Learning Techniques. In *Proceedings of the 10th International Conference on Learning Analytics and Knowledge (LAK '20), March 23–27, 2020, Frankfurt, Germany*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3375462.3375486

1 INTRODUCTION

Academic achievement has become increasingly important because it largely determines admission to further education and future opportunities on the labour market. In contemporary education systems, grades are an indication of academic achievement. Subject tests gauge an underlying (latent) skill which results in such a grade. In practice, however, important decisions of pupils' prospect are based on the general view that a teacher has of a pupil. While latent skills can be measured accurately through tests, combining different test results is less straightforward. For example, should the performance on mathematics be just as important in making decisions as English? In this paper, we want to combine test results to form an overall assessment of a pupil to help them in making important educational decisions. After all, one of the most important decisions a pupil has to make in the final year of primary school is which type of secondary education they want to pursue, especially in countries which have a tracked educational system such as the Netherlands.

The Netherlands' secondary education system is characterised by division — a layered education system ranging from vocational to academic school types. Grades determine what type of secondary education a pupil is eligible for. Subsequently, the type of secondary

education determines what type of tertiary education a pupil can go to and consequently which jobs they can apply for. To adequately guide pupils through the education system, school advice plays an important part in helping pupils decide on which school type to pursue. For this reason, this study aims to establish a method for determining school advice (i.e. advice on which school type to go to) in secondary education[1]. We are particularly interested in finding those pupils who switch between school types in their first three years since they are usually the ones relying most on this advice.

1.1 Problem statement

In the Netherlands, teachers in secondary education already give school advice to pupils. They do so based on the results of a series of progress tests but generally also rely on their own observations. Pupils take tests on different content areas (e.g. English or mathematics) and based on the score of these tests, the teacher determines which school type is most suitable for a pupil.

However, there are three problems with this approach. First of all, latent scores obtained from an item response theory analysis are not intuitive; a score of 150 does not have intrinsic value until you compare it to the scores of classmates. Secondly, latent scores from tests on different content areas are incomparable. That is, a score of 150 for mathematics and a score of 140 for Dutch does not imply this pupil is better at mathematics than at Dutch because the scores have different scales. Thus, scores only have value if you compare them to other scores from the same test. Thirdly, for teachers it is difficult to combine scores from different tests when trying to establish school advice because of the previous problem. How should scores with different scales be counted?

Based on these findings, we look at both domain-specific and domain-agnostic approaches such as item response theory (IRT) and machine learning (ML). Using these techniques, we attempt to produce high-quality predictions to estimate pupils' achievement in a few years based on their current performance including the implicit (and some explicit) weights that are put on these areas. To determine the validity of school advice, we compare the predictive validity of different methods.

1.2 Contributions

We make several scientific contributions: (1) a way to combine test scores on a unidimensional scale on a large scale test administration, (2) a proof-of-concept showing how domain-specific techniques compare to domain-agnostic techniques, and (3) an analysis of the trade-off between predictive power and explainability.

Concerning its practical implications, this work provides some guidance for the selection of school advice prediction models. It also identifies the potential risk that such models might trigger a positive feedback loop, and the need for explainability of such models. These challenges should be addressed in the following years. This paper is structured as follows: In Section 2, we provide the background from which the data stems. Then, Section 3 briefly describes the data and outlines a procedure for calibrating an IRT model (Section 3.1) and shows how to train a variety of ML models (Section 3.2). Subsequently, Section 4 compares the models in terms of predictive accuracy on which conclusions are drawn in Section 5. Finally, Section 6 extensively discusses the implications and the limitations of the proposed solution, including its issues with explainability.

[1]While we aim to establish school *advice* for pupils, we actually predict their school *type* as an indication of where they will be in a few years. This is referred to as predicting pupils' school advice.

2 BACKGROUND

This research revolves around a pupil monitoring system (PMS) in the Netherlands; that is an information system widely used in Dutch secondary education to keep track of pupils' educational progress over multiple years [29]. The PMS consists of multiple tests and test versions in which multiple skills of the pupils are measured. Pupils can take tests at four moments in time: at the start of year one (T_0) and at the end of year one (T_1), two (T_2), and three (T_3). The further the pupils progress in their education, the more difficult the tests become relative to their starting level at T_0. Moreover, there are different test versions which differ in difficulty, depending on the school type. Finally, tests aim to measure pupils' skills in various content areas, such as English reading skills or mathematics.

The results from these tests are used by teachers to determine the progress of pupils in comparison to their peers. For each content area, teachers receive a score that is consistent with other test scores for that content area. In other words, two tests taken by the *same* pupil at the *same* moment in time but on *different* content areas have independent scores. In this research, we attempt to establish school advice for pupils by combining these scores into an (implicit) composite score.

There are approaches such as Elo-ratings [9] that dynamically track the developing skills of pupils in real-time [6]. However, when one desires a fine-grained model for a high-stakes, these methods might be less suitable as they lack several properties for measurement [5]. Hence, usually an IRT approach is preferred in this context. The extend to which ML is suitable is evaluated in this paper.

3 METHODS

We outline two different approaches for predicting school advice, namely by using item response theory and by using various machine learning techniques. We built these models using data from the PMS. The data set contains 17 891 pupils and 1 173 questions at test moment T_0. Of those 17 891 pupils, only approximately 19% switched school type between T_0 and T_3. As stated before, there are multiple test versions per test moment. More fundamentally, in the Dutch education system, the school types (ranging from vocational to academic) VMBO-BB, VMBO-KB, VMBO-GL, HAVO, and VWO exist which we aggregate as VMBO, HAVO, and VWO. The four test versions in the PMS 515–518 correspond to VMBO(-BB/KB), VMBO(-GL), HAVO, and VWO respectively. Furthermore, the data comes from tests administered between 2015 and 2018.

The main focus of this article is a comparison between the results obtained by more common frequently used item response theory (IRT) models and (machine learning) models which are not regularly used in educational assessment. Within the IRT framework we have also attempted to include multidimensional IRT models, but due to the complex, incomplete data collection design, these models

could not be computed. From the wide range of available machine learning models, we have chosen to include a random forest and a neural network model because these models differ substantially in terms of explainability and flexibility.

3.1 Item Response Theory

A popular method in psychometrics is *item response theory* (IRT) [12, 21, 22]. In contrast to *classical test theory (CTT)* [31], IRT focusses on the theory underlying the item and so does not have a test-level focus like CTT. Therefore, IRT is often more justifiable and more practical to solve measurement problems with than CTT [10]. In the context of test theory, an *item* is usually a question on a test which can have either one or multiple correct answers. Furthermore, a pupil's answer to an item is called a *response*. An item can be either dichotomous (i.e. correct or incorrect) or polytomous (more than two scoring points). A scored item is the representation of a pupil's response with respect to the true answer of the item.

The idea behind IRT is that a person's score is derived from the interaction of at least these two elements: (1) a person's innate skill at some subject called the *ability* and (2) the difficulty of an item referred to as *item difficulty*. A latent trait, or a latent variable, is a hypothetical construct that is not directly observable from the data [2]. In the context of IRT, the latent trait of a person is their *ability*. Thus, the goal of IRT is essentially to make an ordering of the relative ability of respondents [25].

To estimate the difficulty parameter, a Rasch model [28] is used. Given the ability parameter θ and the item parameter β, we can model a person's likelihood $P(x_{ij} = 1|\beta_i)$ of endorsing an item as a sigmoid (logistic) function which is more aptly called the *item response function (IRF)*:

$$P(x_{ij} = 1|\theta_j, \beta_i) = \frac{\exp(\theta_j - \beta_i)}{1 + \exp(\theta_j - \beta_i)},$$

where i denotes the question and j references a person. This model is also known as the one-parameter logistic model (1PL) since only one parameter (β) is used to estimate θ.

To calibrate items on a unidimensional scale, we use the *Dexter* package in R [23]. Items are calibrated using a Rasch model and by following the conditional maximum likelihood (CML) procedure where the difficulty and ability are estimated in two separate iterations. Subsequently, ability parameters are estimated using expected a posteriori (EAP).

3.1.1 Item and Ability Parameters. The modelling of response data is a process that is sometimes referred to as *item calibration*. Items at T_0 have been calibrated on a unidimensional scale by using a unidimensional model. To this end, we have estimated a Rasch model for test moment T_0 using the `fit_enorm` function from the *Dexter* package.

Estimating person parameters in *Dexter* is straightforward. Using the function `ability`, we estimate the corresponding person ability at T_0 which is visualised together with the observed score in Figure 1. This is an increasing monotone function since the observed (sum) score is a *sufficient statistic* for the model (and thus its outcome; the ability). That is, a higher sum score is associated with a higher ability estimate. Additionally, the plot has four lines which correspond to the number of test versions for test moment

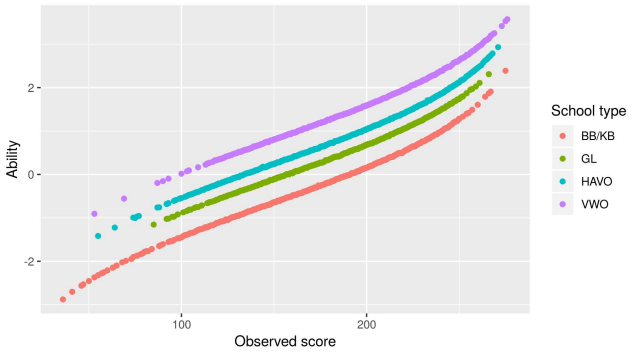


Figure 1: The relationship between the observed test score and estimated ability is an increasing monotone function.

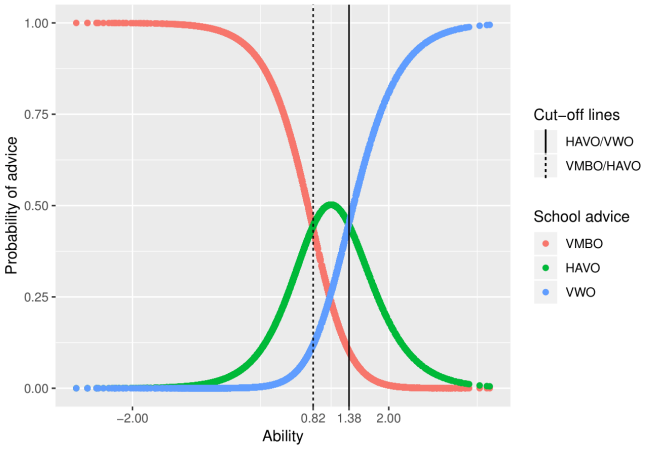


Figure 2: This plot shows the fitted values of the multinomial log-linear model. Just like before, being in the HAVO class was predicted with lower accuracy.

T_0. We can thus show how each test version has been designed for a specific school type (although they can be administered to pupils of other school types as well).

3.1.2 Predicting School Advice. However, the goal of this research is not to estimate ability but to determine school advice. To predict school advice at test moment T_3 from scored item responses at test moment T_0, we can fit a multinomial log-linear regression (MLR) with the ability at T_0 as a predictor and school advice at T_3 as an outcome variable. We do so by using the function `multinom` from the *nnet* package [30] in R. In contrary to ML procedures, this model does not follow a validation-set approach because the ability distribution has already been fixed by *Dexter* when the IRT model was applied to the data and therefore cannot be overfit.

Figure 2 visualises the distributions for the various school types. We can draw cut-off borders by visual inspection although the model also automatically calculates the class probabilities for each data instance (i.e. pupil). From this figure, we can expect that the model is worse at predicting pupils in the HAVO class because it is being 'squashed' in between the other two classes.

Table 1: Sensitivity, specificity, and AUC for the multinomial log-linear model for classifying pupils from ability scores.

School advice	Sensitivity	Specificity	AUC
VMBO	0.781	0.891	0.744
HAVO	0.494	0.747	0.679
VWO	0.679	0.839	0.719
Overall	0.661	0.830	0.826

Table 2: Sensitivity, specificity, and AUC for the multinomial log-linear model for classifying pupils who switched school type.

School advice	Sensitivity	Specificity	AUC
VMBO	0.403	0.701	0.601
HAVO	0.471	0.735	0.802
VWO	0.471	0.735	0.603
Overall	0.455	0.727	0.660

ROC analysis is performed to give insight regarding the sensitivity/specificity trade-off. The AUC is considered by Pelanek [26] to be appropriate for evaluation of natural classification problems, but since it regards performance over all possible thresholds, sensitivity and specificity is also provided for a single threshold. The specific numbers per class are given in Table 1. Being in the VMBO class was predicted the best followed closely by VWO with HAVO lagging much behind which can be explained by the fact that the HAVO class gets cut-off at both sides of the curve. Overall, the area under the curve (AUC) is 0.83. Note how the specificity is much higher than the sensitivity, although this can be changed by adjusting the (probability) thresholds.

Evaluating Switching of School Type. At the beginning of this research, we stated how there is not only a need to accurately predict those who stay in the same school type throughout test moments but also to provide special attention to predicting those who do switch. To this end, we provide a brief overview of how well IRT has performed in terms of predicting switchers.

Table 2 provides the sensitivity, specificity, and AUC for pupils in the three school types but this time only for pupils who actually switched. The classes have high specificity but low sensitivity, indicating that not many pupils are classified as switching from this class but if it does predict so, the model is fairly certain this is actually the case. Interestingly, the AUC of predictions for pupils in the HAVO class is much higher than for the other classes. One explanation may be the result of the 'squashing' that occurs in Figure 2. That is, only a few pupils are classified as being in the HAVO class (while there are much more) which culminates into correctly predicting many HAVO pupils who switched school type.

Overall, the models predicts pupils who do not switch more reliably than those who do switch; compare the sensitivity (switchers 0.45; all pupils 0.661), specificity (switchers 0.73; all pupils 0.830), and AUC (switchers 0.66; all pupils 0.826). Notwithstanding this relatively poor performance, a binomial test points out the predictions are better than the no information rate, $p = 0.04$ (1-sided).

3.2 Machine Learning

In this section, we outline the usage of two very different types of models: (i) a relatively transparent model and (ii) a *'black-box'* model. By doing this, we appraise what the trade-off is between the complexity and performance of a model. We hypothesise that a more complex (i.e. black-box) model performs better, while a transparent model, despite given less accurate results, may be more valuable thanks to a higher degree of explainability.

3.2.1 Select Modelling Technique. As mentioned before, we select two models at either end of the complexity spectrum. Concretely, we look at a random forest (RF) and a single hidden layer (i.e. shallow) neural network (NN). RFs [3] are fairly transparent since they are built on decision trees which are easy to interpret. In contrast, a NN [30] is much more complex due to the hidden layer which essentially forms a model of other models. While there are many other models available, we limit ourselves to only these two since the goal is not to try every model available, but rather to examine two well-known models that are very different from each other.

3.2.2 Generate Test Design. One of the key challenges in training ML models is choosing a suitable test design. To make a fair comparison between IRT and ML, we opt to limit the ML models to the item responses. Thus, the predictions for the school types are based on a matrix with only 0's (incorrect), 1's (correct), 9's (not filled in), and missing values (item was not administered). The latter option has to do with the fact that the tests follow a design where the difficulty of the items has been adapted to the ability of pupils. In essence, the data used for prediction is a sparse matrix with values *not* missing at random (NMAR) [20].

Since most classifiers cannot deal with large amounts of missing values (especially if they are NMAR), we consequently train one classifier per test version. Because our goal is to predict school advice at T_3 from item responses at T_0, there are four test versions to deal with: 515, 516, 517, and 518. The procedure for generating the test design is as follows:

(1) Select the items from T_0;
(2) Split the data into a training (70%) and test (30%) set;
(3) Split the training data into four parts corresponding to their test version;
(4) Train the models on each of the parts.

In this research, there is a special interest in those who switch school type. However, we have already established that there are many more people who do not switch than those who do switch. A classifier is therefore prone to predicting *none* (or very few) pupils as having switched to a different school type. To alleviate this, we follow a cost-sensitive learning approach [8] and assign *case weights* to observations. According to this approach, a case weight determines the *misclassification cost* that occurs when an observation is incorrectly classified; the higher the weight, the more costly the misclassification. To gauge the impact of these weights, we train two variations of a model: one with case weights and one without case weights. The weights are chosen such that the group of switchers has the same sum of weights as the group who did not switch.

Table 3: Performance metrics for 12 models. For models estimated for test version 515, some metrics could not be computed due to division by zero and is therefore not shown in this table.

Test	Model	Weighted	Sensitivity	Specificity	AUC
516	NN	Unweighted	0.740	0.870	0.691
516	NN	Weighted	0.742	0.871	0.719
516	RF	Unweighted	0.863	0.932	0.698
516	RF	Weighted	0.861	0.931	0.634
517	NN	Unweighted	0.692	0.846	0.529
517	NN	Weighted	0.567	0.784	0.638
517	RF	Unweighted	0.697	0.848	0.685
517	RF	Weighted	0.700	0.850	0.713
518	NN	Unweighted	0.790	0.895	0.715
518	NN	Weighted	0.772	0.886	0.659
518	RF	Unweighted	0.795	0.897	0.788
518	RF	Weighted	0.790	0.895	0.792

3.2.3 Build Model. In the previous subsection, we argued the need to train 16 different models: two techniques, namely a RF and a NN; one model per test version (4 in total); and a variation of every model with weights and no weights. Training these models and tuning their hyperparameters in a loop can be done effortlessly with the `train` function from the *caret* package [17] in R. For the RF the *Ranger* package [33] is used (which is just a faster reimplementation in C++ of the original Fortran code by Breiman [3]) and for the NN we utilise the *nnet* package [30].

The *caret* package tries different combinations of hyperparameters, automatically performs cross-validation (CV) and then calculates a performance metric. To speed up this process, we perform *futility analysis* [18] to adaptively resample hyperparameters such that it discards clearly sub-optimal combinations. When tuning hyperparameters in the *caret* package, there are two types of performance measures which can be optimised for in a classification setting, that is (i) accuracy and (ii) Cohen's kappa. While the latter is more suitable for class imbalance (as is the case for school type), we instead use accuracy because, as it turns out, kappa cannot always be computed and thus results in a value of 0. For each hyperparameter, we try 20 different combinations (i.e. tune length).

3.2.4 Assess Model. The performance of these trained models can be evaluated on the test set. Similar to Section 3.1, we do this by looking at the metrics sensitivity, specificity, and area under the curve (AUC). Table 3 presents these results but we discuss them in greater detail in this section. Because we trained a separate classifier per test version, we will evaluate them per classifier as well. One characteristic of performance that is true for all models in Table 3 is that the specificity is higher than the sensitivity. Of course, one could adjust the threshold t such that sensitivity and specificity are balanced. Another characteristic of these models we note is that there is a minimal difference in terms of performance between the weighted and unweighted model.

Test version 515. Test version 515 roughly corresponds to VMBO-BB and VMBO-KB which is one of the vocational school subtypes. Consequently, it rarely occurs that people go 'upstream' by multiple levels. In fact, out of 3 085, only 6 pupils switched to the HAVO school type. While case weighing has been applied to accommodate for class imbalance, this situation is too severe: not every fold in the CV loop may now have switchers (i.e. there is only one response class). These results are therefore excluded in Table 3. We find there to be a specificity of 0.999 for all four models (i.e. NN and RF, one with weights and one without) while the sensitivity cannot be computed. Consequently, the AUC cannot be computed either.

Test version 516. As mentioned before, test version 516 corresponds to school advice of VMBO-GL. When looking at the performance, the weighted NN performs a bit better in terms of sensitivity and specificity than an unweighted NN. This discrepancy is likely caused by the random search algorithm used in *caret* which tries random combinations of hyperparameters and then selects the best ones available. Due to this randomness, slightly different optimal values may have been selected.

The unweighted RF performs much better than the NN described previously in terms of sensitivity and specificity but not so much on AUC. This indicates that, at one point, a RF performs much better than a NN but on average a RF is only slightly better. Finally, we note that the sensitivity and specificity of the *weighted* model are similar to that of the *unweighted* model.

Test version 517. In test version 517 (corresponding to HAVO), 30% of the pupils switched to a different school type. Consequently, we expect the models to perform worse for this test version.

As expected, both sensitivity and specificity are much lower than for the other test versions. Because many more pupils who were administered test version 517 switched school type, we expect weighing to play a larger role. Although both the sensitivity and specificity are lower for a weighted than for an unweighted NN, the AUC is *higher*. Thus, while an *unweighted* NN performs better for a specific threshold, a *weighted* NN performs better *on average*. The RF models perform worse for this test version as well. Nevertheless, the AUC has not decreased by much.

In contrast to NNs, a weighted RF model performs slightly better than an unweighted RF. However, it is unclear whether this is actually caused by case weights or by the effects of the random search pattern when tuning hyperparameters.

Test version 518. Since test version 518 roughly corresponds to VWO, we expect most people to stay in VWO and some going downstream to HAVO or even VMBO. For the unweighted NN, not only are the number of switchers approximately equal between test version 516 and 518, but the models also perform approximately the same. A weighted NN slightly underperforms in comparison to an unweighted NN. A RF performs similar to a NN. A weighted RF performs slightly worse than an unweighted RF in terms of sensitivity and specificity but a bit better on average AUC.

Evaluating Switching of School Type. Analogously to Section 3.1, we evaluate the impact of the model on pupils who have *actually* switched school type. The sensitivity, specificity, and AUC of the 12 models for pupils who switched school type are visually displayed in Figure 3. While these results are not discussed in as much detail as in Section 3.2.4, we note several key observations that characterise the model performance of this specific group of switchers.

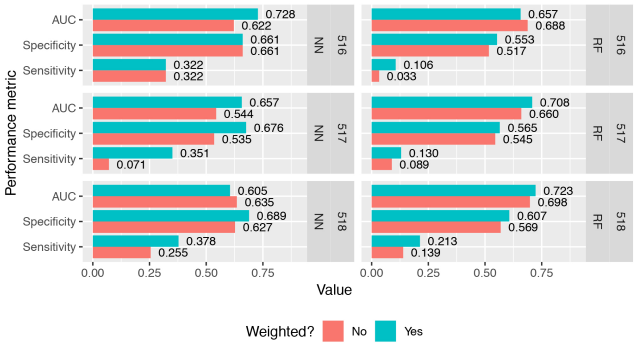


Figure 3: The sensitivity, specificity, and AUC of 12 models (NN and RF; weighted and unweighted; one model per test version) for pupils who switched school types.

The first observation is straightforward; that is pupils who switch are likely harder to predict that those who do not, so it makes sense that the models perform worse on this group. Averaging over all models, the sensitivity has decreased by 0.55, the specificity by 0.28, and the AUC by 0.028 in comparison to the unweighted models. While the decrease in specificity and most notably sensitivity is substantial, the AUC has only gone down slightly in performance. This may indicate that the threshold t along which we can move the ROC curve is chosen sub-optimally. In other words, there are thresholds which are favourable to pupils who switched, but less favourable to those who do not switch.

A second observation is that — in contrast to the performance metrics shown in Table 3 — *weighted* models perform better than *unweighted* models in terms of AUC. This is an important observation as it confirms our intention that we want to weigh pupils who switch school types higher than pupils who do not switch. Thus, Figure 3 effectively shows that the poor model performance on the group of switchers can be mitigated at the cost of the model performance on the group of non-switchers by employing a cost-sensitive learning approach. The question then becomes: *how much could the accuracy of predictions for non-switchers be allowed to decrease in order for the accuracy of the predictions for switchers to increase?*

Finally, we note that — as is the case for the full dataset with data for all pupils — specificity is much higher than sensitivity. While we find that the accuracy of the models on switchers is substantially *worse* than the no information rate, $p < .001$, the practical consequences are limited: it only implies that *if* we know that a pupil will switch, it is better to predict the majority class.

4 RESULTS

In this section, we compare the two methods in terms of predictive performance. More specifically, we compare the two models in two different ways: (i) in terms of predictive accuracy and (ii) in terms of sensitivity, specificity, and AUC. This is because different performance metrics may come to different conclusions. One thing to keep in mind is to view these results in light of their relative *explainability* which is further discussed in Section 6.1.

Table 4: McNemar's chi-squared test for four ML models. The labels from the different test versions have been aggregated. The columns *Model 1 and 2 acc* indicate the number of correct labels alongside the diagonal (i.e. the true positives).

Model 1	Model 2	Model 1 acc	Model 2 acc	$\mathcal{X}^2$	df	p-value
Weighted RF	IRT	0.813	0.657	316.49	1	<.001
Weighted NN	IRT	0.735	0.657	74.73	1	<.001
Unweighted RF	IRT	0.814	0.657	315.76	1	<.001
Unweighted NN	IRT	0.781	0.657	194.38	1	<.001
Weighted RF	Weighted NN	0.813	0.735	180.57	1	<.001
Weighted RF	Unweighted RF	0.813	0.814	0.07	1	.79
Weighted RF	Unweighted NN	0.813	0.781	72.25	1	<.001
Weighted NN	Unweighted RF	0.735	0.814	174.31	1	<.001
Weighted NN	Unweighted NN	0.735	0.781	58.27	1	<.001
Unweighted RF	Unweighted NN	0.814	0.781	79.81	1	<.001

4.1 Model Performance on all Pupils

First, we look at the predictive accuracy on all pupils. To compare IRT and ML in terms of performance, we have computed a McNemar's chi-squared test [1] to compare the *accuracy* of IRT[2] and the four ML models. This is achieved by aggregating the class labels of each model of all test versions (i.e. pretending there is only one test version) and then tabulating a confusion matrix of the two methods from which the diagonals can be compared against each other. The McNemar's chi-squared test is used since it can be applied to a 2x2 contingency matrix where the *McNemar test statistic* is defined as $\mathcal{X}^2 = \frac{(b-c)^2}{b+c}$ [1]. Clearly, this should only be performed on the test set. Results are presented in Table 4. The tests point out that all ML models are *significantly different* from IRT, $p < .001$. However, this does not imply ML models are performing *better* than IRT, but judging from the higher sensitivity, specificity, and occasional higher AUC it is easy to imagine that ML techniques indeed perform better than IRT. Additionally, the higher accuracy of the ML models also indicates they indeed perform better than IRT.

Furthermore, we have compared every ML model with every other ML model in a similar fashion. One way to appreciate these models is to rank them by their accuracy:

(1) Unweighted RF (accuracy of 0.81)
(2) Weighted RF (accuracy of 0.81)
(3) Unweighted NN (accuracy of 0.78)
(4) Weighted NN (accuracy of 0.73)
(5) IRT (accuracy of 0.66)

From Table 4, we see that the predictions of a *weighted* RF and an *unweighted* RF are similar, $\mathcal{X}^2(1, N = 5\,364) = 0.07, p = .79$. This pattern has previously been observed — albeit without a significance value — in Table 3 where the performance metrics of both models were consistently similar.

4.1.1 Comparing Sensitivity, Specificity, and AUC. In addition to looking at the predictive accuracy of the models, we can compare the other performance measures of IRT given in Table 1 against the performance measures of ML given in Table 3. One use of doing this is that it serves as a validation or robustness measure of the previously described results. Note that this is only for data from the test set. Furthermore, this comparison is convoluted because

[2] Although we used MLR to make predictions from the IRT procedure, we use the term IRT to refer to the general process that involves both IRT and MLR.

Table 5: McNemar's chi-squared test for all models of the group of pupils that switched.

Model 1	Model 2	Model 1 acc	Model 2 acc	χ^2	df	p-value
Weighted RF	IRT	0.147	0.693	464.83	1	<.001
Weighted NN	IRT	0.354	0.693	202.21	1	<.001
Unweighted RF	IRT	0.091	0.693	542.82	1	<.001
Unweighted NN	IRT	0.167	0.693	425.53	1	<.001
Weighted RF	Weighted NN	0.147	0.354	154.13	1	<.001
Weighted RF	Unweighted RF	0.147	0.091	49.59	1	<.001
Weighted RF	Unweighted NN	0.147	0.167	3.33	1	.07
Weighted NN	Unweighted RF	0.354	0.091	235.37	1	<.001
Weighted NN	Unweighted NN	0.354	0.167	125.39	1	<.001
Unweighted RF	Unweighted NN	0.091	0.167	51.61	1	<.001

classifiers are trained per test version and can therefore not easily be compared to IRT. Therefore, the general patterns we have seen between the two approaches are described by mapping test versions to school type.

Starting with the VMBO class, we can roughly map this to test version 515 and 516. Test version 515 has been rendered inoperable due to only very few people switching school type and is thus left out of the equation. In terms of sensitivity and specificity, a (un)weighted NN slightly underperforms and a (un)weighted RF slightly outperforms IRT. When it comes to AUC, however, IRT is the better model. This may be relevant if one decides sensitivity should be more favoured instead of specificity (or vice versa); at the current threshold, it means RF performs slightly better and a NN performs slightly worse than IRT.

The HAVO class is interesting because it contains many more switchers than other classes. This is precisely where ML may have an edge; ML models trained on test version 517 all outperform IRT in terms of sensitivity and specificity. The differences are also much greater than in the previous test version. When it comes to AUC, NNs perform much worse while RFs perform slightly better. Thus, we can say that IRT is, on average, a slightly better model than NNs in the HAVO class, but NNs still perform better at the current threshold. RFs outperform IRT on every metric.

Finally, we look at the VWO class which we can roughly map to test version 518. Again, in terms of sensitivity and specificity, ML outperforms IRT. NNs are slightly behind IRT when it comes to AUC, a weighted NN more so than an unweighted NN. On the other hand, RF also outperforms IRT in terms of AUC; consequently, we can say that a RF is a better model if a pupil's school advice at T_0 is VWO or their administered test version is 518.

4.2 Performance on Switchers

Again, we direct some of our attention towards those who switch school type. Similar to before, we have also computed McNemar's chi-squared tests [1] to compare the predictive accuracy of IRT and ML models and displayed the results in Table 5. Looking at the comparisons of each ML model with respect to IRT, we see that all are significantly different. Moreover, the predictive accuracy of IRT is much higher than for any ML model. Being provided with the previously described results and the much higher accuracy, we can conclude IRT performs better at predicting school advice of pupils who switch.

We can also draw some conclusions about ML models. For instance, we can create a ranking based on their predictive accuracy:

(1) IRT (accuracy of 0.69)
(2) Weighted NN (accuracy of 0.35)
(3) Unweighted NN (accuracy of 0.17)
(4) Weighted RF (accuracy of 0.15)
(5) Unweighted RF (accuracy of 0.091)

This ordering is the complete opposite of the one we found when looking at all pupils. Furthermore, from Table 5 we see that an unweighted NN and a weighted RF perform similar, $\chi^2(1, N = 987) = 3.3, p = 0.07$. This reversed ranking is caused by weighted models outperforming unweighted models for pupils who switched school type and by the complexity of the models. While previously we found that a RF performed better than a NN, it is the other way around when we only focus on those who switch school type. Because a NN is generally more flexible than a RF (i.e. a RF is more *robust*), NNs can more easily adapt itself to 'unlikely' data patterns such as pupils who switch school type and therefore deviate from the majority of pupils. This behaviour is tightly connected to the *explainability* of the models which is further discussed in Section 6.1.

4.2.1 Comparing Sensitivity, Specificity, and AUC. We also compare the sensitivity, specificity, and AUC for the group of pupils that switched school type. Thus, we compare Table 2 (IRT) to Figure 3 (ML). Again, the mapping between school types and test version is unclear, but we assume the structure 515/516–VMBO, 517–HAVO, and 518–VWO to provide a reasonable comparison.

From Table 2 we can see that for the VMBO school type the sensitivity is 0.4, the specificity 0.7, and the AUC 0.6. Then, from Figure 3 we learn that the sensitivity and specificity of every ML model trained on test version 516 is lower than for IRT (again, we leave test version 515 out of the equation). That said, the AUC for every ML model is higher than for IRT which indicates that ML models are better on average, while at the current threshold IRT performs better.

For the HAVO class, IRT achieved a sensitivity of 0.47, a specificity of 0.74, and an AUC of 0.8. None of the ML models performed better on any measure. Hence, we can conclude IRT is clearly the better model for the HAVO school type.

Finally, we compare the performance of model on the VWO school type. IRT has a sensitivity of 0.47, a specificity of 0.7, and an AUC of 0.6. These results are comparable to that of the VMBO school type; the same is true for ML models. That is, all ML models performed worse in terms of sensitivity and specificity, but better in terms of AUC.

5 CONCLUSION

School advice is used for determining which school type is most suitable for a pupil. To establish school advice, we used two disparate ways to predict school type, namely both IRT and ML techniques. For IRT, we calibrated a Rasch model on all test versions and extracted expected a posteriori (EAP) ability estimates, estimates that are then plugged into a multinomial log-linear regression model to calculate pupils' probabilities for every distinct school type. For ML we trained one classifier (a RF and a NN) per test version due to a sparse matrix. Furthermore, we followed a cost-sensitive learning

approach and applied higher case weights to pupils who switched between school type.

In terms of overall predictive accuracy, we found that ML outperformed IRT with a RF giving more accurate results than a NN. However, when zooming in on the group of pupils who switched school type, IRT severely outperformed all other models with NNs being more accurate than RFs. Moreover, using case weights substantially improved the ML models for the group of switchers.

6 DISCUSSION

This section discusses a number of choices made in this research. Concretely, it expands on both the philosophical and psychological aspect that underlies this research as well as discusses some practical and ethical implications, limitations, and finally provides pointers for future research.

6.1 Theoretical Analysis of Explainability

In addition to comparing the predictive quality of models, another important aspect is the difference in explainability of domain-specific and domain-agnostic models. In other words, if a pupil were to ask why they were given certain school advice, to what extent can we explain why this advice was given?

One of the properties of interpretable models is that a model is transparent. In a sense, a *transparent* model is the antithesis of a *black-box* model. Lipton [19] shows how transparency can be subdivided into three distinct viewpoints: (i) simulatability, (ii) decomposability, and (iii) algorithmic transparency. We visit each of these three notions, explain what they entail, and to what extent this applies to IRT, RFs, and NNs.

6.1.1 Simulatability. One way to evaluate whether a model is transparent is to see if a person can grasp the workings of a model in its entirety. That is, *simulatability* [19] refers to what extent a model can be fully understood by a human in a *reasonable* amount of time. This may include time needed to perform the calculations of the model by hand, but may also refer to making inferences based on the model. However, simulatability is not necessarily intrinsic to a model.

While an IRT model with a few items can easily be calculated by hand, this process becomes cumbersome for many items. Moreover, calculating a RF for even a small number of features is time-consuming but walking through a RF from root to leaf tree-by-tree is viable. Finally, it should be self-evident that computing a single hidden layer NN with many input features is infeasible for a human. While both RFs and certainly NNs are difficult to conceptually grasp for a non-expert, this is not the case for IRT. In this model, the difficulty parameter is the difficulty relative to the other items on the test and is therefore very easy to understand. Moreover, the ability parameter is less intuitive but can also be described by looking at the sum score of the scored item responses since this is a *sufficient statistic* for the Rasch model.

6.1.2 Decomposability. The concept of *decomposability* refers to the extent that each input, parameter, and calculation can be intuitively explained. For ML models, we evaluate the decomposability based on their hyperparameters and their feature importance (FI). Although it is in the current setup not possible to do this for IRT

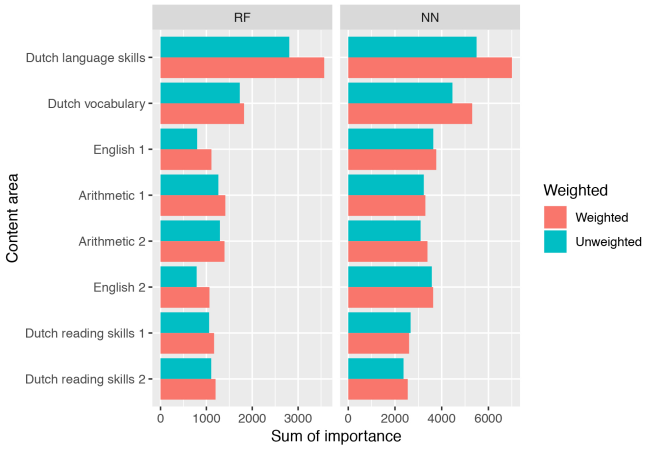


Figure 4: Relative feature importance grouped by subject. This figure shows that Dutch language skills is the most important subject followed by Dutch vocabulary. Note that in this figure, the models over different test versions have been aggregated.

(since it does not have hyperparameters or FI), this model can be explained via its difficulty and ability parameter. The difficulty is easy to interpret and understand since it has real-world value in that it can be linked to items itself. The ability is either useful when comparing it to peers or when looking at sum score. The decomposability is even greater in a 2PL model since this also estimates a *discrimination parameter* which can be interpreted as the importance of an item.

Hyperparameters. For NNs, we found that the number of nodes in the hidden layers is higher in a weighted model than in an unweighted model. From this observation, it appears that weighed models are more complex than their unweighted counterparts. Secondly, the weight decay of the unweighted models is higher than the weighted models. Models with a *higher* weight decay are *less* complex because they are more restricted by the higher penalisation term. Because unweighted models are less complex and more restricted, this may imply that they generalise better to other, previously unseen data [16].

For the hyperparameters of RFs, there seems to be no obvious pattern as to why they were selected. While there is no clear explanation for this, one hint may be given by the fact that there is only little variance in the CV accuracy for different hyperparameters set-ups. For instance, the difference between the best and the worst performing selection (i.e. the range) of hyperparameters for an unweighted RF forest model is only as little as 0.0033. Thus, it hardly seems to matter what value of the hyperparameters is selected since all attempted combinations perform about the same.

Feature Importance. We define the concept of FI as the relative importance of the most important feature in comparison to all the other features. Concretely, the most important feature is defined as having a FI of 100(%) while the others have a score relative to this number. In a RF, the FI is computed as the total decrease in node impurities from splitting on the variable, averaged over all trees.

The FI in NNs is computed using Garson weights [11]. Because each test version has 290–360 input features (i.e. items), we have increased readability by summing over the subjects the features belong to and is displayed in Figure 4. From this figure, it is easy to see that Dutch language skills is viewed as most important in both models. In the RF model, arithmetic 1 and 2 seem to be a little bit more important than the other content areas, but in general, the other content areas are similarly influential. In a NN, only Dutch reading skills 1 and 2 appear to be less important than the other content areas.

6.1.3 Algorithmic Transparency. The final property of a transparent model concerns the algorithm's transparency. In contrast to simulatability, this is intrinsic to the learning algorithm itself. This involves being able to give mathematical guarantees the model will perform well even on unseen data so that it does not need retraining. For both RFs and NNs, we cannot a priori guarantee whether they will work well on previously unseen data and their algorithmic transparency is therefore low. However, for IRT we can guarantee that the sum score of items is directly related to the ability itself; the higher the sum score, the higher the ability since this function is monotone and the sum score is a *sufficient statistic* of the model.

6.1.4 Post-hoc Interpretability. As Lipton [19] notes, post-hoc interpretability involves explaining models *after the fact*. While this has not been implemented in this research, we provide some directions on how post-hoc interpretability can contribute to understanding the results.

By giving verbal explanations of the model's strategy, *Text explanations* can inform a pupil why they were given certain school advice. This can be achieved by (textually) listing some of the items they answered incorrectly but that proved to be important for the model or by training a second model to generate this explanation [e.g. 15]. Secondly, *visualisation* is a popular technique of qualitatively showcasing what a model has learnt. In the context of educational tests, however, it is not clear what type of visualisation can help explain pupils' school advice. Thirdly, *local explanations* explain only a single decision if explaining the whole model is unattainable. However, in this context, this may not be applicable since a single changed value is unlikely to impact the model significantly. Finally, *explanation by example* can be used to provide pupils with response patterns similar to their own to show why their school advice is different.

6.2 Practical Implications

One important aspect of discussing the use of the proposed solutions is to see how it can be applied in practice. First of all, *concept drift* is likely to occur. If the model advises a pupil to go to a different school type than they are currently, they might do so. This means the model then becomes a *positive feedback loop* [4]. That is, a pupil will switch school type because the model said so; the model is correct in that the pupil switched because the model *recommended* the pupil to do so. To combat this problem, the use of *trackers* has been proposed to deal with such dynamically changing data [e.g. 5].

Another practical implication is related to the explainability of the models. With this high performance also comes the cost of complexity; to what extent are we capable to explain how an algorithm comes to a decision? Especially because of the 'right to an explanation' from the General Data Protection Regulation (GDPR) [7], we pose the question whether it is ethical (or even legal) to use algorithms to influence an important choice in the lives of teenagers.

6.3 Limitations

There are also a few other limitations that need to be taken into account. First of all, this is a study aimed at showcasing differences between IRT and ML which means the created models can by no means be readily implemented into practice. We evaluate the validity by using the validity framework laid out by Wohlin et al. [32]. *Internal validity* refers to the extent the models influence the outcome. *Selection* is only partly mitigated since schools can choose whether they can participate in the PMS or not. *Resentful demoralisation* may be present because some pupils do not *want* to participate in the tests. As a consequence, they may not put in as much effort and thereby get different school advice. Internal validity may also be at risk due to the *black-boxness* of ML which leads to a decreased understanding of how the models came to a decision. Another internal validity threat for ML is posed by the seemingly contradictory results found in Table 4 and Table 5 where the ranking of best performing models based on accuracy reverses. *External validity* is concerned with conditions that limit our ability to generalise the results outside the context of this research. Most notably, the *subject population* is a threat because pupils who switched schools, skipped or repeated a year, or just any pupil who had an anomaly in their data has been left out. *Construction validity* is at risk because *confounding constructs and levels of constructs* were not accounted for but can be evaluated by future research. *Conclusion validity* is concerned with the relationship between the models and the outcome. Although *low statistical power* has been mitigated because the data set consists of 17 891 pupils, ML could likely profit from having even more data. Furthermore, drawing conclusions was convoluted because of the mismatch in the number of test versions and the number of outcome classes. This led to a comparison of aggregated predictive accuracy with only a loose comparison of sensitivity, specificity, and AUC.

6.4 Future Directions

Based on these results and limitations we also provide some directions for future research.

Decision support. While there have been some successes to provide decision support for teachers [13], the models proposed in this research have not yet reached that level of maturity. Moreover, Holstein et al. [14] shows how teachers are more valuable for providing support for pupils than educational AI systems.

Different Models, Metrics, and Classes. Other models for either IRT (2PL, 3PL, or multidimensional) or ML (support vector machines, deep NNs) may be attempted to evaluate its results. Similarly, comparing other metrics or choosing different thresholds along the ROC curve also affect which model performs 'best'. Additionally,

choosing different class aggregations (i.e. more fine-grained classes) will highly influence the outcome.

Confounding Variables. For the sake of fairness, only the item responses were used to keep bias at a minimum. However, it may still be the case that there are *confounding variables* which led to a bias in the model. This can be inspected by either reviewing the predictions by group to see if there are any differences or by adding those variables we think are confounding to the model to see if they are of any importance.

Integrating IRT and ML. To even further improve the performance of the models, one might look into integrating IRT and ML into a new framework. For example, Pliakos et al. [27] have already integrated these methods for the prediction of personalised items' recommendation in e-learning. They found their method to perform better than either IRT or ML by itself. Furthermore, multidimensional IRT models could be integrated with ML to provide even further improvement.

ACKNOWLEDGMENTS

This work has previously been published as a Master's thesis [24].

REFERENCES

[1] Alan Agresti. 1990. *Categorical data analysis* (3rd ed.). Wiley-Interscience, New York.

[2] Kenneth A Bollen. 2002. Latent variables in psychology and the social sciences. *Annual review of psychology* 53, 1 (2002), 605–634.

[3] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.

[4] Matthieu J. S. Brinkhuis, Marjan Bakker, and Gunter Maris. 2015. Filtering data for detecting differential development. *Journal of Educational Measurement* 52, 3 (2015), 319–338. https://doi.org/10.1111/jedm.12078

[5] Matthieu J. S. Brinkhuis and Gunter Maris. 2019. Tracking Ability: Defining Trackers for Measuring Educational Progress. In *Theoretical and Practical Advances in Computer-based Educational Measurement*, Bernard P. Veldkamp and Cor Sluijter (Eds.). Springer International Publishing, Cham, 161–173. https://doi.org/10.1007/978-3-030-18480-3_8

[6] Matthieu J. S. Brinkhuis, Alexander Savi, Abe Hofman, Frederik Coomans, Han van der Maas, and Gunter Maris. 2018. Learning As It Happens: A Decade of Analyzing and Shaping a Large-Scale Online Learning System. *Journal of Learning Analytics* 5, 2 (Aug. 2018), 29–46. https://doi.org/10.18608/jla.2018.52.3

[7] Council of the European Union and European Parliament. 2016. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). *Official Journal of the European Union* 119, 1 (May 2016), 1–88.

[8] Charles Elkan. 2001. The Foundations of Cost-Sensitive Learning. *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence* 17, 1 (2001), 973–978.

[9] Arpad E. Elo. 1978. *The rating of chessplayers, past and present.* B.T. Batsford, Ltd., London.

[10] Susan E. Embretson and Steven P. Reise. 2000. *Item response theory for psychologists.* Lawrence Erlbaum Associates Publishers, Mahwah, NJ, US.

[11] G. David Garson. 1991. Interpreting Neural-network Connection Weights. *AI Expert* 6, 4 (April 1991), 46–51. http://dl.acm.org/citation.cfm?id=129449.129452

[12] Ronald K Hambleton, Hariharan Swaminathan, and H Jane Rogers. 1991. *Fundamentals of item response theory.* Vol. 2. Sage, Amherst, USA.

[13] Kenneth Holstein, Bruce M. McLaren, and Vincent Aleven. 2019. Co-Designing a Real-Time Classroom Orchestration Tool to Support Teacher–AI Complementarity. *Journal of Learning Analytics* 6, 2 (Aug. 2019), 27–52. https://doi.org/10.18608/jla.2019.62.3

[14] Kenneth Holstein, Bruce M. McLaren, and Vincent Aleven. 2019. Designing for Complementarity: Teacher and Student Needs for Orchestration Support in AI-Enhanced Classrooms. In *Proceedings, 20th International Conference on Artificial Intelligence in Education*, Seiji Isotani, Eva Millán, Amy Ogan, Peter Hastings, Bruce McLaren, and Rose Luckin (Eds.), Vol. 11625. Springer International Publishing, Cham, 157–171. https://doi.org/10.1007/978-3-030-23204-7_14

[15] Samantha Krening, Brent Harrison, Karen M. Feigh, Charles Lee Isbell, Mark Riedl, and Andrea Thomaz. 2017. Learning From Explanations Using Sentiment and Advice in RL. *IEEE Transactions on Cognitive and Developmental Systems* 9, 1 (March 2017), 44–55. https://doi.org/10.1109/TCDS.2016.2628365

[16] Anders Krogh and John A. Hertz. 1992. A Simple Weight Decay Can Improve Generalization. In *NIPS'91 Proceedings of the 4th International Conference on Neural Information Processing Systems*, J. E. Moody, S. J. Hanson, and R. P. Lippmann (Eds.). Morgan-Kaufmann, Denver, CO, USA, 950–957. http://papers.nips.cc/paper/563-a-simple-weight-decay-can-improve-generalization.pdf

[17] Max Kuhn. 2008. Building Predictive Models in R Using the caret Package. *Journal of Statistical Software* 28, 5 (Nov. 2008), 1–26. https://doi.org/10.18637/jss.v028.i05

[18] Max Kuhn. 2014. *Futility Analysis in the Cross-Validation of Machine Learning Models.* Technical Report. Pfizer Global R&D, Groton, USA. http://arxiv.org/abs/1405.6974 arXiv: 1405.6974.

[19] Zachary C. Lipton. 2016. The Mythos of Model Interpretability. In *Proceedings of the 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016)*, Kim Been, Dmitry M. Malioutov, and Kush R. Varshney (Eds.). arXiv.org[stat.ml], New York, NY, 96–100. http://arxiv.org/abs/1606.03490 arXiv: 1606.03490.

[20] Roderick J. A. Little and Donald B. Rubin. 2002. *Statistical Analysis with Missing Data* (2 ed.). John Wiley & Sons, Inc., Hoboken, New Jersey.

[21] Frederic M. Lord. 1980. *Applications of item response theory to practical testing problems.* Lawrence Erlbaum Associates, Mahwah, NJ, US.

[22] Frederic M. Lord, M.R. Novick, and Allan Birnbaum. 1968. *Statistical theories of mental test scores.* Addison-Wesley, Oxford, England.

[23] Gunter Maris, Timo Bechger, Jesse Koops, and Ivailo Partchev. 2019. dexter: Data Management and Analysis of Tests. (2019). https://CRAN.R-project.org/package=dexter

[24] Koen J. Niemeijer. 2019. *Constructing and Predicting School Advice for Academic Achievement: A Comparison of Item Response Theory and Machine Learning Techniques.* Master's thesis. Utrecht University, Utrecht.

[25] Ivailo Partchev. 2004. *A visual guide to item response theory.* Friedrich-Schiller-Universitat Jena.

[26] Radek Pelanek. 2015. Metrics for Evaluation of Student Models. *Journal of Educational Data Mining* 7, 2 (2015), 1–19. https://doi.org/10.5281/zenodo.3554665

[27] Konstantinos Pliakos, Seang-Hwane Joo, Jung Yeon Park, Frederik Cornillie, Celine Vens, and Wim Van den Noortgate. 2019. Integrating machine learning into item response theory for addressing the cold start problem in adaptive learning systems. *Computers & Education* 137 (Aug. 2019), 91–103. https://doi.org/10.1016/j.compedu.2019.04.009

[28] Georg Rasch. 1960. *Studies in mathematical psychology: I. Probabilistic models for some intelligence and attainment tests.* Nielsen & Lydiche, Oxford, England.

[29] Alma Van Til and Herman Van Boxtel. 2015. *Wetenschappelijke verantwoording Toets 0 t/m 3, tweede generatie.* Cito, Arnhem. https://www.cito.nl/-/media/Files/kennisbank/cito-bv/96_wetenschappelijke-verantwoording-volgsyteemvo-gen2.pdf?la=nl-NL

[30] W. N. Venables and B. D. Ripley. 2002. *Modern Applied Statistics with S* (fourth ed.). Springer, New York. http://www.stats.ox.ac.uk/pub/MASS4

[31] N.D. Verhelst and H.H.F.M. Verstralen. 1994. *The one parameter logistic model: computer program and manual.* Universiteit Twente, CITO, Arnhem. https://research.utwente.nl/en/publications/the-one-parameter-logistic-model-computer-program-and-manual

[32] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, and Anders Wesslén. 2012. *Experimentation in software engineering.* Springer, Berlin, Heidelberg.

[33] Marvin N. Wright and Andreas Ziegler. 2017. ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R. *Journal of Statistical Software* 77, 1 (2017), 1–17. https://doi.org/10.18637/jss.v077.i01