Construction and Analysis of Educational Tests Using Abductive Machine Learning

El-Sayed M. El-Alfy[1] and Radwan E. Abdel-Aal[2]

[1]Information and Computer Science Department and [2]Computer Engineering Department
College of Computer Sciences and Engineering
King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia
{alfy, radwan}@kfupm.edu.sa

Address for corresponding author:

Dr. El-Sayed M. El-Alfy
P. O. Box 371
KFUPM
Dhahran 31261
Saudi Arabia

Email: alfy@kfupm.edu.sa
Phone: +(966) 3-860-1930
Fax: +(966) 3-860-2174

Abstract

Recent advances in educational technologies and the wide-spread use of computers in schools have fueled innovations in test construction and analysis. As the measurement accuracy of a test depends on the quality of the items it includes, item selection procedures play a central role in this process. Mathematical programming and the item response theory (IRT) are often used in automating this task. However, when the item bank is very large, the number of item combinations increases exponentially and item selection becomes more tedious. To alleviate this problem, several attempts were made to utilize heuristic search and machine learning approaches, including neural networks. This paper proposes a novel approach that uses abductive network modeling to automatically identify the most-informative subset of test items that can be used to effectively assess the examinees without seriously degrading accuracy. Abductive machine learning automatically selects only effective model inputs and builds an optimal network model of polynomial functional nodes that minimizes a predicted squared error criterion. Using a training dataset of 1500 cases (examinees) and 45 test items, the proposed approach automatically selected only 12 items which classified an evaluation population of 500 cases with 91% accuracy. Performance is examined for various levels of model complexity and compared with that of statistical IRT-based techniques. Results indicate that the proposed approach significantly reduces the number of test items required while maintaining acceptable test quality.

Keywords: Abductive machine learning, Abductive networks, Neural networks, Optimal test design, Educational measurements, Item response theory, Test analysis, Test construction.

1. Introduction

There is a growing interest in the use of computers in automating test construction and analysis, especially for large-scale testing (Buyske, 2005; Stocking, Swanson & Pearlman, 1991). A primary goal of administering an educational test is to locate examinees on the ability scale and to classify them into categories with acceptable accuracy. This is usually achieved by observing their response to items included in the test, which are selected from a larger collection of items in the form of an item bank or pool. One of the earlier findings on educational measurements is that classification accuracy is improved when the test consists of a large number of discriminating items which are neither too easy nor too difficult for the test takers (Berger, 1997). However, increasing the number of items is not cost effective, as it requires more physical resources, e.g. paper, and consumes longer times from both the examiners and examinees. While test analysis is concerned with item characteristics and how accurate a test is in classifying examinees, test construction is concerned with selecting items to be included in the test that ensure accurate assessment using relatively few items. Unfortunately, the process of test construction and analysis could be quite labor-intensive. As a result, several methods have been proposed for automating the process based on the item response theory (IRT) (Lord, 1980; Hambleton & Swaminathan, 1985; Stocking, Swanson & Pearlman, 1991; van der Linden & Hambleton, 1997). Within the framework of IRT, examinees are described by a single latent variable and each item is described by the Fisher's information function. The item information function (IIF) provides test developer with an indication of the measurement precision for the test item. Accordingly, a test can be formed by selecting items on the basis of their information function. Lord (Lord, 1980) outlined a procedure for selecting items such that the information function of the constructed test (sum of the information functions for the individual items it includes) approximates a target information function to a satisfactory degree. The smaller the distance between the target information function and the constructed test information function, the more precise the test is in measuring ability. Although this procedure is conceptually simple, it becomes impractical to apply as the item bank grows in size. Mathematical programming provides a more systematic approach for optimal test design. A great amount of research has been conducted in this area; see for example, (Lord, 1980; Hambleton & Swaminathan, 1985; Theunissen, 1985; Baker, 1988; van der Linden, 1987; van der Linden & Boekkooi-Timminga, 1989; Adema, 1990a; Adema, 1990b; Adema, Boekkooi-Timminga & van der Linden, 1991; Fletcher, 2000; van der Linden, 2005). With these approaches, the test construction problem is modeled as an optimization problem to maximize (or minimize) some objective function while meeting a number of constraints in the form of test specifications. However, the application of such approaches is often hindered by the need for a prior estimation of item characteristics. Moreover, the search for optimal solutions becomes computationally intensive as the size of the item bank increases. To overcome these limitations, a number of heuristic approaches have been proposed to facilitate finding solutions as close to optimal as possible in a reasonable computation time using Tabu search, simulated annealing, *etc*. (Adema & van der Linden, 1989; Adema 1990b; Swanson & Stocking, 1993; Jeng & Shih, 1997; Luecht, 1998; Hwang, Yin & Yeh, 2006). Recently, artificial neural networks have been successfully used to solve many complex modeling and optimization problems in several areas of science, engineering, and the social sciences, and some attempts have been made in the area of educational measurements. Sun and Chen (Sun & Chen, 1999) used neural networks for constructing educational tests. With their approach, the test information function is transformed into an energy function which is minimized using a neural network model. When the energy function stabilizes, the state of the network represents a solution. Although this approach can be used to effectively solve the problem, the computing processes are complex. To achieve good results at a faster pace, a greedy approach similar to the neural network method was later proposed (Sun, 2001).

This paper proposes an alternative approach based on abductive machine learning for identifying the most informative subset of items that can be used to effectively assess the examinees without severely degrading measurement accuracy. Abductive machine learning has emerged as a powerful technique in artificial intelligence for solving diagnostic problems (Montgomery & Drake, 1991). It builds an optimal network model composed of non-linear functional elements (nodes) organized in layers in a manner that minimize a predicted squared error (PSE) criterion (Barron, 1984). Thus, it can represent complex and uncertain relationships between dependent (output) and independent (input) variables. There are several advantages for using abductive networks for discovering complex relationships between input and output variables. Unlike most approaches such as regression and neural networks, the self-organizing abductive modeling technique automatically synthesizes an optimal network architecture to fit the training data without requiring the user to specify the network architecture in advance. It has also been shown that the prediction accuracy of abductive networks can be higher compared to that of neural networks (Montgomery & Drake, 1991). Furthermore, abductive networks were found to be faster, easier to use, and involved fewer parameters (Agarwal, 1999). The iterative tuning process necessary with regression and neural network approaches is largely reduced with the abductive approach. Accordingly, an abductive network model can be used effectively as an estimator for predicting the output of a complex system, a classifier for handling difficult pattern recognition problems, or a system identifier for determining which inputs are important for modeling the system (Agarwal, 1999). The approach selects only relevant model inputs and synthesizes more transparent models that provide greater insights and give better explanations for the modeled phenomena compared to neural networks, which is an important advantage in human-related disciplines, e.g. education, medicine, and the environment. The abductive network approach has been previously used to model and forecast the educational score in school health surveys (Abdel-Aal & Mangoud, 1996) and in a variety of other areas including weather forecasting (Abdel-Aal & Elhadidy, 1995), financial modeling (Agarwal, 1999), electric load forecasting (Abdel-Aal, 2004), drilling tool life prediction (Lee, Liu & Tarng, 1999), electronic combat (Montgomery, Hess & Hwang, 1990), and fault diagnosis in electrical power transmission networks (Sidhu, Cruder & Huff, 1997).

The rest of this paper is organized as follows: Section 2 briefly describes abductive network machine learning, highlighting similarities and differences with neural networks. Section 3 gives an outline of the dataset used in our experiments together with the results of some exploratory analysis. Section 4 presents the results of using abductive networks to model examinees' ability in terms of their response to the test items at various levels of specified model complexity. Section 5 describes corresponding results obtained using statistical and IRT-based techniques. Section 6 compares the results obtained from the two approaches and conclusions are made in Section 7.

2. Abductive Machine Learning

Abductory inductive mechanism (AIM) (AbTech, 1990) is a supervised inductive machine-learning tool for automatically synthesizing abductive network models from a database of inputs and outputs representing a training set of solved examples. As a self-organizing group method of data handling (GMDH) (Farlow, 1984), the tool can automatically synthesize adequate models that embody the inherent structure of complex and highly nonlinear systems. The automation of model synthesis not only lessens the burden on the analyst but also safeguards the model generated from being influenced by human biases and misjudgments. The GMDH approach is a formalized paradigm for iterated (multi-phase) polynomial regression capable of producing a high-degree polynomial model in effective predictors. The process is 'evolutionary' in nature, using initially simple (myopic) regression relationships to derive more accurate representations in the next iteration. To prevent exponential growth and limit model complexity, the algorithm selects only relationships having good predicting powers within each phase. Iteration is stopped when the new generation regression equations start to have poorer prediction performance than those of the previous generation, at which point the model starts to become overspecialized and therefore unlikely to perform well with new data. The algorithm has three main elements: representation, selection, and stopping. It applies abduction heuristics for making decisions concerning some or all of these three aspects.

To illustrate these steps for the classical GMDH approach, consider an estimation database of n_e observations (rows) and $m+1$ columns for m independent variables ($x_1, x_2, \ldots, x_m$) and one dependent variable y. In the first iteration we assume that our predictors are the actual input variables. The initial rough prediction equations are derived by taking each pair of input variables ($x_i, x_j; i, j = 1, 2, \ldots, m$) together with the output y and computing the quadratic regression polynomial (Farlow, 1984):

$$y = A + B x_i + C x_j + D x_i^2 + E x_j^2 + F x_i x_j. \qquad (1)$$

Each of the resulting $m(m-1)/2$ polynomials is evaluated using data for the pair of x variables used to generate it, thus producing new estimation variables ($z_1, z_2, \ldots, z_{m(m-1)/2}$) which would be expected to describe y better than the original variables. The resulting z variables are screened according to some selection criterion and only those having good predicting power are kept. The original GMDH algorithm employs an additional and independent selection set of n_s observations for this purpose and uses the regularity selection criterion based on the root mean squared error r_k over that data set, where

$$r_k{}^2 = \sum_{\ell=1}^{n_s}(y_\ell - z_{k\ell})^2 \Bigg/ \sum_{\ell=1}^{n_s} y_\ell{}^2 \quad ;k = 1,2,...,m(m-1)/2\,. \tag{2}$$

Only those polynomials (and associated z variables) that have r_k below a prescribed limit are kept and the minimum value, r_{min}, obtained for r_k is also saved. The selected z variables represent a new database for repeating the estimation and selection steps in the next iteration to derive a set of higher-level variables. At each iteration, r_{min} is compared with its previous value and the process is continued as long as r_{min} decreases or until a given complexity is reached. An increasing r_{min} is an indication of the model becoming overly complex, thus over-fitting the estimation data and performing poorly in predicting the new selection data. Keeping model complexity checked is an important aspect of GMDH-based algorithms, which keep an eye on the final objective of constructing the model, *i.e.*, using it with new data previously unseen during training. The best model for this purpose is that providing the shortest description for the data available (Barron, 1984). Computationally, the resulting GMDH model can be seen as a layered network of partial quadratic descriptor polynomials, each layer representing the results of an iteration.

A number of GMDH methods have been proposed which operate on the whole training data set thus avoiding the use of a dedicated selection set. The adaptive learning network (ALN) approach, AIM being an example, uses the predicted squared error (PSE) criterion (Barron, 1984) for selection and stopping to avoid model overfitting, thus eliminating the problem of determining when to stop training in neural networks. The criterion minimizes the expected squared error that would be obtained when the network is used for predicting new data. AIM expresses the PSE error as:

$$PSE = FSE + CPM(2K/n)\sigma_p{}^2 \ , \qquad (3)$$

where *FSE* is the fitting squared error on the training data, *CPM* is a complexity penalty multiplier selected by the user, *K* is the number of model coefficients, *n* is the number of samples in the training set, and $\sigma_p{}^2$ is a prior estimate for the variance of the error obtained with the unknown model. This estimate does not depend on the model being evaluated and is usually taken as half the variance of the dependent variable *y* (Barron, 1984). As the model becomes more complex relative to the size of the training set, the second term increases linearly while the first term decreases. *PSE* goes through a minimum at the optimum model size that strikes a balance between accuracy and simplicity (exactness and generality). The user may optionally control this trade-off using the *CPM* parameter. Larger values than the default value of 1 lead to simpler models that are less accurate but may generalize well with previously unseen data, while lower values produce more complex networks that may overfit the training data and degrade actual prediction performance.

AIM builds networks consisting of various types of polynomial functional elements. The network size, element types, connectivity, and coefficients for the optimum model are automatically determined using well-proven optimization criteria, thus reducing the need for user intervention compared to neural networks. This simplifies model development and reduces the learning/development time and effort. The models take the form of layered feed-forward abductive networks of functional elements (nodes) (AbTech, 1990), see Fig. 1. Elements in the first layer operate on various combinations of the independent input variables (*x*'s) and the element in the final layer produces the predicted output for the dependent variable *y*. In addition to the main layers of the network, an input layer of normalizers convert the input variables into an internal representation as *Z* scores with zero mean and unity variance, and an output unitizer unit restores the results to the original problem space.

The used version of AIM supports the following main functional elements:

(i) A white element which consists of a constant plus the linear weighted sum of all outputs of the previous layer, i.e.:

$$\text{"White" Output} = w_0 + w_1x_1 + w_2x_2 + ... + w_nx_n, \tag{4}$$

where $x_1, x_2,..., x_n$ are the inputs to the element and $w_0, w_1, ..., w_n$ are the element weights.

(ii) Single, double, and triple elements which implement a third-degree polynomial expression with all possible cross-terms for one, two, and three inputs respectively; for example,

$$\text{"Double" Output} = w_0 + w_1x_1 + w_2x_2 + w_3x_1^2 + w_4x_2^2 + w_5x_1x_2 + w_6x_1^3 + w_7x_2^3, \tag{5}$$

3. The Dataset

In order to evaluate the performance of the proposed approach, we used a dataset from (Rudner, 2005) which consists of a sample of 2000 cases (examinees) and a 45-item test. It is assumed that examinees are classified based on a single-ability parameter, θ. Hence, each case in the dataset gives the response vector and the true ability level for an individual test taker. Table 1 lists the information for the first twenty cases of the dataset, showing the response vector to the test items and the corresponding true ability parameter for each case. The test items are numbered as 1, 2, 3, …, 45 according to the column they occupy in the dataset. The column number is used as an item identification (IID) throughout this paper. Test items are dichotomously scored, *i.e.* when the test is taken, the examinee's response to each item is encoded as 1 (*i.e.* correct) or 0 (*i.e.* incorrect). It is also assumed that the examinee can skip some items which are marked x (*i.e.* missing) in Table 1. Out of the 2000 cases, only two ability values

(0.1%) fall outside the range {-4 to +4}, so practically the ability scale ranges from -4 to +4. The distribution of examinees in this sample approximately follows a normal distribution over the ability range -4 to +4, as indicated by the histogram plot in Fig. 2. For the purpose of experiments reported later in this paper, the total sample population is divided into two categories (fail and pass) and each category is further divided into two groups (G1 and G2 for the fail category and G3 and G4 for the pass category), as marked on Fig. 2. Details of the size of these categories and groups and their boundaries on the ability scale are listed in Table 2.

4. Abductive Network Modeling

Abductive networks were used to model the relationship between the ability level of the examinees and their response to the 45 test items, through training on a subset of the dataset. To account numerically for skipped test items in the response vectors, these input items were assigned 0, while correct responses were represented as +1 and incorrect responses as -1. The objective of abductive modeling is to utilize the property of automatic selection of effective input variables to identify the optimum subset of test items that explain the ability outcome. To verify the adequacy of the resulting model, performance of the model in predicting the ability level was evaluated on an evaluation subset not seen previously during training. Two modeling experiments were performed which are described in the two subsections below.

4.1. Modeling for Pass/Fail Classification

Abductive networks were used to model a two-level outcome for the examinees' ability as a function of relevant input test items. Ability values in the range {-4.1456 to +0.0055} were assigned an output level 0 (fail category) while values in the range {+0.0075 to +4.0583} were assigned an output level 1 (pass category). Referring to Table 2, the first category consists of groups G1 and G2 and the second category consists of groups G3 and G4, with each category comprising 1000 cases. The overall set of 2000 cases was then randomly split into two subsets: 1500 cases used for training and 500 cases for evaluation. Responses for all the 45 test items were enabled as inputs to the model. Table 3 shows abductive model structures synthesized at various levels of model complexity as indicated by the CPM parameter specified prior to training. The variable number indicated at a model input in Table 3, *e.g.* Var_i, correspond to the IID of the test item selected as input to the model during model synthesis. Var_46 is the binary (pass/fail) ability output. Lower CPM values give more complex models. The same model structure and selected model inputs were preserved over the CPM range of 0.2 to 2.0, which is a sign of model robustness. All these models select the same 12 inputs (test items) out of the 45 inputs available, thus achieving about 73.3% dimensionality reduction for the modeled problem. The selected model inputs correspond to test items having IIDs 3, 10, 17, 19, 23, 25, 27, 31, 36, 41, 43, and 45. Preserving the same subset of inputs over a decade of variations in the CPM value indicates the importance of the selected inputs to the modeling process. At CPM = 5, a slightly simpler model is synthesized which uses only 11 inputs, namely 3, 6, 7, 15, 17, 25, 27, 31, 36, 41, and 45. Approximately 73% of these inputs are included in the previous subset of 12 inputs. The table also lists the percentage classification error for each model on both the training and evaluation datasets. As model complexity increases (lower CPM values), the model fits the training data more closely and the classification performance on the training dataset improves. However, the possibility of overfitting increases, which degrades performance on the external evaluation set. Fig. 3 plots the above two model performance indicators versus the CPM value. Best classification performance on the evaluation set is obtained using the optimum model with CPM = 0.5 which gives a classification error of 9.4%. Table 4(a) shows the resulting confusion matrix and Table 4(b) lists the parameters characterizing the classification performance of this optimum model on the evaluation set, including classification accuracy, sensitivity, specificity, positive predictive value, and negative predictive value. Throughout the above analysis, “Pass” is considered as the positive outcome. The results indicate a minimum value of approximately 90% for all performance parameters.

4.2. Modeling for Further Classification within Categories

Abductive models were also developed to further discriminate between examinees in each of the fail and pass categories based on their ability level. Referring to Table 2, the 1000 cases in the fail category were split into two groups G1 (100 cases) and G2 (900 cases) corresponding to two binary levels for the ability. The 1000 cases were then randomly split into two subsets: 750 for training and 250 for evaluation. The G1/G2 model was synthesized using the training subset and evaluated on the evaluation subset. Similarly, the G3/G4 model was developed for the pass category. Table 5 shows the optimum abductive network structures that minimize the classification error on the evaluation subsets for the G1/G2 and the G3/G4 models. The first model selects 10 input items, {2, 4, 5, 8, 9, 12, 29, 34, 39, 43}, while the second model selects 11 items, {8, 12, 15, 18, 19, 20, 21, 27, 32, 33, 38}. Only 2 items are common between the two subsets, which demonstrates the ability of abductive learning to successfully select different subsets of test items that achieve different objectives. Classification accuracy on the evaluation set is 90% and 93% for the G1/G2 and the G3/G4 models, respectively.

5. IRT-Based Analysis

Following the three parameter logistic model (3PL) (Lord, 1980), each dichotomously scored test item is characterized by three parameters; namely discrimination power parameter, *a*, item difficulty parameter, *b*, and guessing parameter, *c*. With this model, the probability that a test taker with ability θ correctly answers an item with parameters (*a*, *b*, *c*) is given by (Lord, 1980):

$$P(\theta) = c + \frac{1-c}{1+e^{-a(\theta-b)}} \,. \tag{6}$$

where $a \in (0, \infty)$, $b \in (0, \infty)$ and $c \in (0,1)$. Using the empirical dataset described in Section 3, individual test items were calibrated using Newton-Raphson maximum likelihood estimation as outlined in (Lord, 1980; Rudner, 2005). Table 6 lists the actual values for the *a*, *b*, and *c* parameters for each test item. We carried out the calibration for test items given the response patterns and true abilities (method a). The estimated values for the three parameters as well as their standard errors, SE_*a*, SE_*b*, SE_*c*, for each test item are shown in columns (a) of Table 6. The table also shows the number of cases, *N*, used for calibrating each item. We have also estimated the examinees' abilities given their response patterns and the item parameters calculated above. Examinees were then classified according to the estimated ability as pass or fail by setting the threshold value to 0. The total percentage classification error (passing a failed examinee or failing a passed examinee) was found to be 6.15%, with the false fail rate (failing a passed examinee) being 2.8% and the false pass rate (passing a failed examinee) being 3.35%. We also estimated the ability parameter, θ, and item parameters given only the response patterns (method b). The item parameters estimated in this way together with their standard errors are shown in columns (b) of Table 6. As a result, the total classification error increased to 6.45% while the false fail rate dropped to 2.6%. Table 7 lists the true and estimated ability parameter, θ, and its standard error, SE, for the first twenty cases in the dataset using item parameters and responses (columns a) and responses only (columns b).

To examine the correlation between the actual and estimated parameters of the test items, we plot the scatter diagram and give the computed correlation coefficients in Fig. 4 and Fig. 5 for methods a and b, respectively. The results show that parameters estimated from response vectors and true abilities (method a) are more correlated to the actual parameters than those estimated from the response vectors alone. Similarly Fig. 6 shows the scatter diagrams and correlation coefficients for estimated abilities using the two methods. Again, we found that the abilities estimated using response vectors and estimated item parameters (method a) are slightly more correlated to the true abilities than those estimated from response vectors alone. The figure also shows the correlation between the two estimated ability parameters. We have also observed that estimating the item parameters and the ability parameter from just the response patterns converged much slower compared to estimating item parameters from θ and response patterns or estimating θ from item parameters and response patterns.

6. Comparison of Results

The purpose of this paper is to investigate the potential use of abductive machine learning in test development as an alternative to conventional methods, *e.g*. IRT-based analysis. This section compares results obtained using the two approaches. As described in Section 5, best results for pass/fail classification using IRT-based analysis using all 45 test items give a classification error of 6.15%. The corresponding optimum abductive model synthesized with CPM = 0.5 gives classification errors of 7.8% and 9.4% on the training and evaluation sets, respectively, and uses only 12 test items, see Table 3. The significant reduction in the number of test items required for the test may justify the slight degradation in classification accuracy. We also examined the properties of test items selected as inputs during the synthesis of the abductive network model for fail/pass classification to verify if they represent an adequate selection according to IRT criteria. Table 8 lists all 45 test items and the results of sorting them in a number of ways. Columns 2, 3, and 4 in the table show the test items sorted according to the actual values of their *a*, *b*, and *c* parameters, respectively, with items having the smallest values listed at the top.

Columns 5 to 11 show the items sorted according to the value of the item information function computed at seven ability levels corresponding to θ = -1.5, -1.0, -0.5, 0.0, +0.5, +1.0, and +1.5. Throughout the table, cells containing test items selected by the optimum pass/fail abductive model with CPM = 0.5 are marked by a black background. The table indicates that items selected by the abductive network approach are concentrated around the middle of the difficulty parameter, *b*, thus satisfying the criteria of being not too easy nor too difficult for the test takers. Most of such items also have high values for the information function at θ = 0, which is the threshold for pass/fail classification. As we go away from this ability cutoff in either direction, the selected items become more scattered. This shows that the abductive learning approach selects test items that are effective discriminators with a high information content at the required ability cutoff level.

To examine the effectiveness of the abductive network approach in identifying the most informative subset of test items, we compared the classification performance of three pass/fail tests, one composed of all 45 test items, another composed of the 12 items selected as inputs by the optimum abductive network model with CPM = 0.5, and the third composed of a randomly selected subset of 12 items {4, 8, 10, 12, 13, 22, 23, 27, 28, 31, 37, 43}. Results are plotted in Fig. 7 for the overall classification error, the false pass rate, and the false fail rate. They indicate that the abductive selection is significantly superior to the random selection, particularly for the classification rate and the false fail rate. It is interesting to note that the abductive false fail rate is slightly lower than that achieved using the full set of test items. We have also examined the test information function, defined as the sum of item information functions for items included in the test, for the three tests described above. Fig. 8 plots the results over the full ability range. Sharper peaks for the information function lead to more precise classification. Although the inclusion of all test items results in a higher peak value, this peak is slightly offset away from the zero ability cut-off for the pass/fail test. However, using the abductive item selection, the peak value coincides more accurately on the cutoff point. The test information peak for the randomly selected subset is lower than the peak for the abductive selection, in spite of the fact that the two subsets have the same size. The former peak is also shallower and further offset from the center, which are all signs of poorer classification performance.

In another experiment, we formed two tests using the abductive item selection made for the two abductive models that perform G1/G2 group discrimination within the fail category and between G3/G4 group discrimination within the pass category, see Fig. 2 and Table 2. The model structures, selected inputs, and classification performance for the two abductive models were introduced in Section 4.2 and given in Table 5. The test information functions for the two tests are shown in Fig. 9. The peak for the G3/G4 model is sharper and more centered around the nominal G3/G4 ability cut-off level of 2, compared to the peak for the G1/G2 model which is shallower and is significantly offset from the nominal G3/G4 ability cut-off level of -2. This explains the relatively poorer classification performance by the G1/G2 model as indicated in Table 5, where the percentage classification error for that model is shown to be approximately 1.5 times that of the G3/G4 model.

7. Conclusions

In this paper, we have demonstrated the utility of abductive machine learning as an alternative tool for educational test design and analysis. Performance of the proposed approach was examined and compared to classical statistical IRT-based techniques using a dataset of 2000 cases and 45 test items with various levels of model complexities and at various ability thresholds. Results indicate that abductive network models can classify examinees with a reasonable classification accuracy. The learning algorithm automatically identifies a concise and effective subset of test items with high discriminatory power, which can be used to form the test. Therefore, large-scale assessment systems can benefit from using abductive networks in test development. In general, results show that abductive networks can improve the educational measurement by reducing the number of items included in the test without severely degrading measurement precision. We have also demonstrated that multiple tests for finer grade classification can be constructed by controlling the ability threshold value. Several areas could benefit from the proposed approach including college placement testing, medical licensing, job applicant screening, and academic achievement testing. This paper lays a new research direction in educational measurement. Future work will attempt to further improve the predication accuracy, e.g. using network ensembles, and extend the modeling approach to multidimensional assessment and polytomous items.

Acknowledgment

The authors are grateful to King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia for continuous support during this work.

References

Abdel-Aal, R. E., and Elhadidy, M. A. (1995). Modeling and Forecasting the Maximum Temperature using Abductive Machine Learning. *Weather and Forecasting*, 10:310-25.

Abdel-Aal, R. E., and Mangoud A. M., (1996). Abductive Machine Learning for Modeling and Predicting the Educational Score in School Health Surveys. *Methods of Information in Medicine*, 35(3):265-71.

Abdel-Aal, R. E. (2004). Short Term Hourly Load Forecasting using Abductive Networks. *IEEE Trans. Power Systems*, 19:164-73.

AbTech Corporation (1990). *AIM User's Manual*, Charlottesville, VA.

Adema, J. J. (1989). *Implementations of the Branch-and-Bound Method for Test Construction Problems*. Research Report 89-6. Enschede: Department of Education, University of Twente, Netherlands.

Adema, J. J., van der Linden, W. J. (1989). Algorithms for Computerized Test Construction Using Classical Item Parameters. *Journal of Educational Statistics*, 14(3): 279-290.

Adema, J. J. (1990a). *Models and Algorithms for the Construction of Achievement Tests*. Ph.D. Thesis, Enschede: University of Twente.

Adema, J. J. (1990b). *A Revised Simplex Method for Test Construction Problems*. Research Report 90-5. Enschede: Department of Education, University of Twente, Netherlands.

Adema, J. J., Boekkooi-Timminga, E., and van der Linden, W. J. (1991). Achievement Test Construction Using 0-1 Linear Programming. *European Journal of Operational Research*, 55, 103-111.

Agarwal, A. (1999). Abductive Networks for Two-Group Classification: A Comparison with Neural Networks. *The Journal of Applied Business Research*, 15(2):1:12.

Armstrong, R. D., Jones, D. H., Wang, Z. (1998). Optimization of Classical Reliability in Test Construction. *Journal of Educational and Behavioral Statistics*, 23(1):1-17.

Baker, F. B., Chen, A. S., and Barmish, B. R. (1988). Item Characteristics of Tests Constructed by Linear Programming. *Applied Psychological Measurement*, 12:189-199.

Barron, A. R. (1984). Predicted Squared Error: A Criterion for Automatic Model Selection. In Farlow, S. J. (Ed.), *Self-Organizing Methods in Modeling: GMDH Type Algorithms*, (pp. 87-103). Marcel-Dekker, New York.

Berger, M. P. F. (1997). Optimal Designs for Latent Variable Models: A Review. In Rost, J. and Langeheine, R. (Eds.), *Applications of Latent Trait and Latent Class Models in the Social Sciences* (pp. 71-79). Münster: Waxmann.

Boekkooi-Timminga, E. (1987). Simultaneous Test Construction by Zero-One Programming. *Methodika*, 1:101-112.

Boekkooi-Timminga, E. (1989). *Models for Computerized Test Construction*. The Haag, Netherlands: Academisch Boeken Centrum.

Boekkooi-Timminga, E. (1990). The Construction of Parallel Tests from IRT-Based Item Banks. *Journal of Educational Statistics*, 15(2):129-145.

Buyske, S. (2005). Optimal Design in Educational Testing. In Berger, M. P. F., and Wong, W. K. (Eds.), *Applied Optimal Designs*. John Wiley & Sons.

Farlow, S. J. (1984). The GMDH Algorithm. In Farlow, S. J. (Ed.), *Self-Organizing Methods in Modeling: GMDH Type Algorithms*, (pp. 1-24). Marcel-Dekker, New York.

Fletcher, R. B. (2000). A Review of Linear Programming and its Application to the Assessment Tools for Teaching and Learning (asTTle) Projects. Technical Report 5, Project asTTle, University of Auckland.

Hambleton, R. K., and Swaminathan, H. (1985). *Item Response Theory: Principles and Applications*. Kluwer Academic Publishers Group, Netherlands.

Hwang, G.-J., Yin, P.-Y., and Yeh, S. H. (2006). A Tabu Search Approach to Generating Test Sheets for Multiple Assessment Criteria. *IEEE Transactions on Education*, 49(1): 88-97.

Jeng, H. L., and Shih, S. G. (1997). A Comparison of Pair-wise and Group Selections of Items using Simulated Annealing in Automated Construction of Parallel Tests. *Psychological Testing*, 44(2):195-210.

Lee, B. Y., Liu, H. S. and Tarng, Y. S. (1999). An Abductive Network for Predicting Tool Life in Drilling, *IEEE Transactions on Industry Applications*, 35(1): 190-195.

Lord, F. M. (1980). *Applications of Item Response Theory to Practical Testing Problems*. New Jersey: Lawrence Erlbaum.

Luecht, R. M. (1998). Computer-Assisted Test Assembly Using Optimization Heuristics. *Applied Psychological Measurement*, 22(3):224-236

Montgomery, G. J., Hess, P., and Hwang, J. S. (1990). Abductive Networks Applied to Electronic Combat. *Proceedings of SPIE - The International Society for Optical Engineering*, 1294:454-465.

Montgomery, G. J. and Drake, K. C. (1991). Abductive Reasoning Networks. *Neurocomputing*, 2(3):97-104.

Rudner, L. M. (2005). *PARAM-3PL Calibration Software for the 3 Parameter Logistic IRT Model*. Available: http://edres.org/irt/param

Sidhu, T. S., Cruder, O., and Huff, G. J. (1997). An Abductive Inference Technique for Fault Diagnosis in Electrical Power Transmission Networks, *IEEE Transactions on Power Delivery,* 12(1):515-522

Stocking, M. L., Swanson, L., and Pearlman, M. (1991). *Automated Item Selection Using Item Response Theory*. Research Report 91-9. Princeton, NJ: Educational Testing Service.

Sun, K. T., and Chen, S. F. (1999). A Study of Applying the Artificial Intelligent Technique to Select Test Items. *Psychological Testing*, 46(1):75-88.

Sun, K.-T. (2001). A Greedy Approach to Test Construction Problems. *Proceedings of the National Science Council*, ROC-Part D, 11(2):78-87.

Swanson, L., and Stocking, M. L. (1993). A Model and Heuristic for Solving Very Large Item Selection Problems. *Applied Psychological Measurement*, 17(2):151-166.

Theunissen, T. J. J. M. (1985). Binary Programming and Test Design. *Psychometrika*, 50:411-420.

van der Linden, W. J. (1987). Automated Test Construction using Minimax Programming. In van der Linden, W. J. (Ed.), *IRT-Based Test Construction* (pp. 1-16). Enschede, Netherlands: Department of Education, University of Twente.

van der Linden, W. J., and Boekkooi-Timminga, E. (1989). A Maximum Model for Test Design with Practical Constraints. *Psychometrika*, 54:237-247.

van der Linden, W. J. and Hambleton, R. K. (Eds.), (1997). *Handbook of Modern Item Response Theory*, Springer-Verlag.

van der Linden, W. J. (2005). *Linear Models for Optimal Test Design*. Springer.

Table 1. First twenty cases in the dataset. Shown are the examinee’s identification (EID), response pattern and true ability level for each case. (1 = correct, 0 = incorrect, x = skipped).

EID	Response Pattern	True Ability
1	10001101000000 1000x0001000000100010010010 11101	-1.5841
2	011001000100000010000101101100100100011000001	-1.689
3	111111111101101111100111111010011100111 1100	0.494
4	100000011000x100000x0010000000101111111 1x0100	-0.981
5	111111011111111111110111111111101111011111111	1.4221
6	01101100111000001101 00x101101010010001110 1111	0.0353
7	111011101110011111011010111110101111101011101	0.7333
8	01x011101010100110000000111110100111011111100	-0.2385
9	100011100110010111001100001000000001010 1000000	-0.5911
10	101000101110111101000001000110000110100111000	-0.6697
11	110011101010110111100001010110100111111101101	0.0707
12	0111111011101000110000111001100011101 01x00111	-0.3552
13	011110101110111111110x11111111111111111111101	1.0341
14	011110001000010001111000010011100110011101100	-0.7209
15	110010101010100001000001010110000010101101010	-1.1992
16	111111001011101111111001011001110011111101111	0.2684
17	01111110111011111110000101110101011101111111 1x	0.6881
18	010100101110100011010001001010100011001101001	-0.62
19	011000101000100001001010010000100101101100100	-0.2659
20	01111110101011111111100111111 1x100111001111110	0.9098

Table 2. Details of examinee categories and groups in the dataset.
*: The only two values outside the ability range -4 to +4. See Fig. 2.

Category	Group	Nominal Ability Range	Actual Ability Range		Number of Examinees
			Start	End	
Fail	G1	-4 to -2	-4.1456*	-1.9797	100
	G2	-2 to 0	-1.9796	+0.0055	900
Pass	G3	0 to +2	+0.0075	+1.9971	900
	G4	+2 to +4	+2.0006	+4.0583*	100
Total					2000

Table 3. Structure and performance of abductive network models synthesized for pass/fail classification at various levels of model complexity. Training on 1500 cases and evaluation on 500 cases.

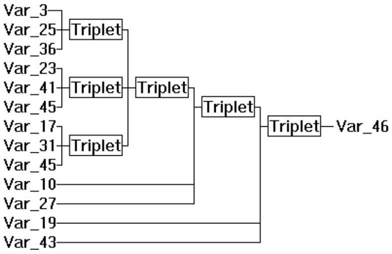
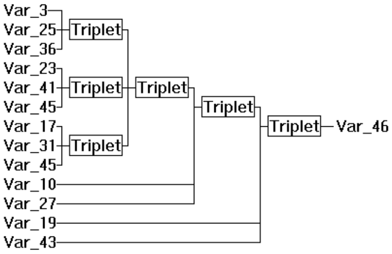
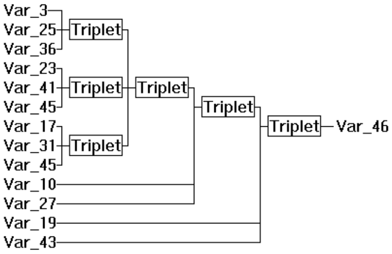
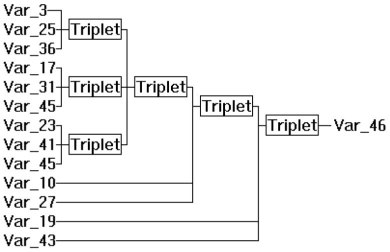
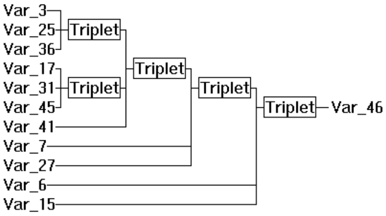
CPM	Model	Selected Test Items	% Classification Error on	
			Training Set	Evaluation Set
0.2	 Var_3 Var_25 Triplet Var_36 Var_23 Var_41 Triplet Triplet Var_45 Triplet Var_17 Triplet Var_46 Var_31 Triplet Var_45 Var_10 Var_27 Var_19 Var_43	3, 10, 17, 19, 23, 25, 27, 31, 36, 41, 43, 45	7.8	9.6
0.5	 Var_3 Var_25 Triplet Var_36 Var_23 Var_41 Triplet Triplet Var_45 Triplet Var_17 Triplet Var_46 Var_31 Triplet Var_45 Var_10 Var_27 Var_19 Var_43	3, 10, 17, 19, 23, 25, 27, 31, 36, 41, 43, 45	7.8	9.4
1	 Var_3 Var_25 Triplet Var_36 Var_23 Var_41 Triplet Triplet Var_45 Triplet Var_17 Triplet Var_46 Var_31 Triplet Var_45 Var_10 Var_27 Var_19 Var_43	3, 10, 17, 19, 23, 25, 27, 31, 36, 41, 43, 45	7.9	9.6
2	 Var_3 Var_25 Triplet Var_36 Var_17 Var_31 Triplet Triplet Var_45 Triplet Var_23 Triplet Var_46 Var_41 Triplet Var_45 Var_10 Var_27 Var_19 Var_43	3, 10, 17, 19, 23, 25, 27, 31, 36, 41, 43, 45	7.9	9.6
5	 Var_3 Var_25 Triplet Var_36 Var_17 Triplet Var_31 Triplet Triplet Var_45 Triplet Var_46 Var_41 Var_7 Var_27 Var_6 Var_15	3, 6, 7, 15, 17, 25, 27, 31, 36, 41, 45	8.1	12.2

Table 4. Confusion matrix (a) and parameters characterizing classification performance (b) for the optimum pass/fail abductive model synthesized with CPM = 0.5 on the evaluation dataset of 500 cases.

(a)

		Predicted	
		1 (261)	0 (239)
Actual	1 (254)	234	20
	0 (246)	27	219

(b)

Classification Accuracy, %	Sensitivity, %	Specificity, %	Positive Predictive Value, %	Negative Predictive Value, %
90.6	92.1	89.0	89.7	91.6

Table 5. Structure and performance for the two abductive models performing further classification of the fail and pass examinees' categories into two groups each: {G1, G2} and {G3, G4}, respectively.

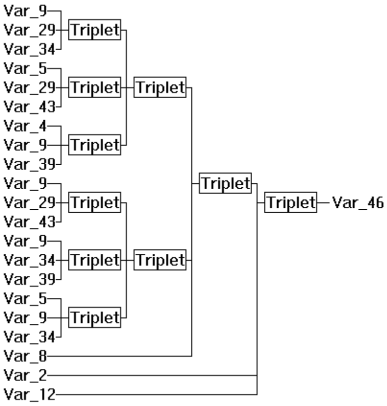
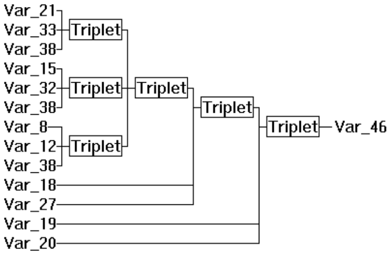
Model	Function	CPM	Structure	Selected Test Items	% Classification Error on:	
					Training set	Evaluation set
G1/G2	Classify the Fail category into groups G1 and G2	0.5	 Var_9 Var_29 Triplet Var_34 Var_5 Var_29 Triplet Triplet Var_43 Var_4 Var_9 Triplet Var_39 Var_9 Triplet Var_29 Triplet Triplet Var_46 Var_43 Var_9 Var_34 Triplet Triplet Var_39 Var_5 Var_9 Triplet Var_34 Var_8 Var_2 Var_12	2, 4, 5, 8, 9, 12, 29, 34, 39, 43	5.3	10
G3/G4	Classify the Pass category into groups G3 and G4	1	 Var_21 Var_33 Triplet Var_38 Var_15 Var_32 Triplet Triplet Var_38 Triplet Triplet Var_46 Var_8 Var_12 Triplet Var_38 Var_18 Var_27 Var_19 Var_20	8, 12, 15, 18, 19, 20, 21, 27, 32, 33, 38	5.1	7

Table 6. Actual and estimated parameters for each test item. Estimated parameters are calculated by two methods: (a) using abilities and responses, and (b) using responses only. IID is the item identification number. *N* is the number of cases used in item estimation and SE is the standard error.

IID	Actual Parameters			Method (a)							Method (b)						
	a	*b*	*C*	*N*	*a*	*b*	*c*	SE_*a*	SE_*b*	SE_*c*	*N*	*a*	*b*	*c*	SE_*a*	SE_*b*	SE_*c*
1	0.967	0.826	0.201	1983	0.9	0.796	0.197	0.065	0.052	0.014	1983	1.027	0.805	0.205	0.075	0.048	0.013
2	1.148	-0.51	0.169	1981	1.066	-0.603	0.142	0.055	0.04	0.021	1981	1.048	-0.722	0	0.047	0.035	0.017
3	1.494	-0.336	0.217	1984	1.571	-0.36	0.221	0.094	0.033	0.019	1984	1.875	-0.262	0.23	0.116	0.029	0.018
4	0.894	0.05	0.205	1977	0.84	0.044	0.206	0.054	0.049	0.018	1977	0.89	0.05	0.188	0.056	0.045	0.018
5	1.039	-0.843	0.221	1981	0.98	-0.878	0.205	0.051	0.046	0.025	1981	0.946	-1.073	0	0.041	0.04	0.025
6	1.272	-0.123	0.219	1971	1.296	-0.184	0.192	0.077	0.036	0.017	1971	1.523	-0.121	0.19	0.091	0.032	0.017
7	1.149	0.025	0.208	1984	1.164	-0.026	0.204	0.071	0.039	0.017	1984	1.312	-0.017	0.184	0.079	0.035	0.017
8	1.023	2.124	0.145	1976	1.032	2.005	0.135	0.088	0.07	0.009	1976	1.279	1.916	0.142	0.12	0.063	0.009
9	1.366	-1.342	0.195	1983	1.327	-1.409	0.169	0.07	0.043	0.033	1983	1.504	-1.358	0	0.072	0.034	0.02
10	1.079	0.17	0.296	1983	1.053	0.213	0.318	0.077	0.048	0.017	1983	1.104	0.205	0.303	0.079	0.045	0.017
11	1.326	-0.657	0.154	1975	1.427	-0.664	0.148	0.076	0.034	0.02	1975	1.818	-0.54	0.145	0.1	0.028	0.02
12	1.372	1.346	0.143	1978	1.631	1.249	0.15	0.137	0.04	0.01	1978	2	1.2	0.15	0.176	0.035	0.01
13	0.707	-1.199	0.207	1982	0.709	-1.327	0.181	0.035	0.059	0.032	1982	0.741	-1.447	0	0.031	0.051	0.035
14	1.232	-0.008	0.164	1980	1.299	-0.036	0.147	0.075	0.034	0.015	1980	1.548	0.028	0.154	0.092	0.03	0.015
15	1.204	0.618	0.214	1979	1.077	0.632	0.203	0.077	0.045	0.014	1979	1.202	0.655	0.211	0.086	0.042	0.014
16	0.688	0.043	0.228	1981	0.713	0.099	0.259	0.051	0.059	0.018	1981	0.722	0.011	0.21	0.048	0.054	0.019
17	1.148	-0.497	0.162	1975	1.248	-0.465	0.211	0.071	0.038	0.02	1975	1.263	-0.497	0.142	0.066	0.034	0.021
18	1.281	-0.811	0.176	1985	1.218	-0.899	0.132	0.061	0.038	0.024	1985	1.31	-0.92	0	0.059	0.031	0.015
19	1.633	0.531	0.233	1970	1.76	0.553	0.237	0.133	0.034	0.013	1970	2	0.543	0.234	0.151	0.031	0.013
20	1.354	0.665	0.194	1978	1.486	0.642	0.209	0.109	0.037	0.013	1978	1.75	0.651	0.215	0.131	0.033	0.013
21	0.978	1.231	0.156	1979	0.913	1.125	0.133	0.062	0.052	0.011	1979	1.033	1.097	0.137	0.071	0.048	0.011
22	1.142	1.015	0.193	1974	1.256	1.031	0.171	0.096	0.044	0.011	1974	1.389	1.005	0.171	0.106	0.041	0.011
23	1.592	-0.476	0.224	1982	1.718	-0.484	0.181	0.099	0.031	0.019	1982	1.945	-0.421	0.162	0.111	0.027	0.018
24	1.671	0.643	0.158	1983	1.681	0.643	0.165	0.119	0.033	0.012	1983	2	0.622	0.161	0.142	0.029	0.012
25	1.504	0.226	0.266	1978	1.344	0.114	0.229	0.088	0.037	0.016	1978	1.443	0.115	0.212	0.092	0.034	0.016
26	1.334	0.063	0.22	1976	1.295	0.014	0.202	0.08	0.037	0.016	1976	1.541	0.063	0.204	0.097	0.032	0.016
27	1.289	-0.208	0.224	1977	1.261	-0.252	0.18	0.073	0.036	0.018	1977	1.398	-0.199	0.17	0.08	0.033	0.018
28	1.28	0.868	0.198	1972	1.276	0.87	0.2	0.097	0.043	0.012	1972	1.453	0.849	0.2	0.111	0.039	0.012
29	1.435	-1.252	0.151	1984	1.523	-1.177	0.174	0.083	0.037	0.028	1984	1.683	-1.173	0	0.081	0.03	0.013
30	1.272	1.084	0.194	1984	1.36	1.03	0.183	0.107	0.043	0.011	1984	1.502	0.994	0.179	0.117	0.039	0.011
31	1.683	-0.301	0.211	1979	1.717	-0.354	0.176	0.1	0.03	0.017	1979	2	-0.298	0.161	0.117	0.026	0.017
32	1.453	1.428	0.091	1984	1.453	1.385	0.082	0.106	0.041	0.008	1984	1.67	1.338	0.083	0.127	0.037	0.008
33	1.471	1.219	0.2	1978	1.883	1.192	0.22	0.18	0.039	0.011	1978	2	1.148	0.214	0.19	0.037	0.011
34	1.358	-0.781	0.231	1986	1.3	-0.805	0.213	0.071	0.039	0.024	1986	1.824	-0.557	0.272	0.111	0.032	0.022
35	1.202	-0.789	0.186	1977	1.189	-0.792	0.229	0.065	0.042	0.024	1977	1.322	-0.72	0.193	0.071	0.036	0.024
36	1.179	-0.597	0.17	1981	1.324	-0.598	0.167	0.071	0.036	0.021	1981	1.439	-0.587	0.103	0.073	0.031	0.02
37	1.178	-0.229	0.233	1978	1.17	-0.19	0.264	0.074	0.042	0.019	1978	1.36	-0.117	0.267	0.087	0.037	0.019
38	1.62	1.628	0.229	1980	1.89	1.579	0.221	0.213	0.048	0.01	1980	2	1.523	0.218	0.229	0.044	0.01
39	1.544	-1.25	0.233	1984	1.857	-1.18	0.293	0.116	0.038	0.029	1984	2	-1.081	0.236	0.119	0.033	0.031
40	1.07	-0.502	0.193	1979	1.117	-0.472	0.19	0.062	0.04	0.02	1979	1.172	-0.493	0.125	0.061	0.036	0.021
41	1.467	0.345	0.179	1985	1.423	0.39	0.167	0.092	0.035	0.013	1985	1.616	0.405	0.168	0.105	0.031	0.013
42	1.052	-0.629	0.207	1980	0.998	-0.592	0.198	0.054	0.044	0.022	1980	0.926	-0.832	0	0.041	0.038	0.021
43	1.289	-0.638	0.161	1978	1.227	-0.737	0.115	0.062	0.036	0.021	1978	1.516	-0.583	0.127	0.079	0.031	0.021
44	1.283	0.363	0.174	1979	1.178	0.315	0.16	0.074	0.038	0.014	1979	1.288	0.31	0.148	0.079	0.035	0.014
45	1.426	0.097	0.182	1976	1.441	-0.076	0.147	0.084	0.032	0.015	1976	1.564	-0.065	0.122	0.088	0.029	0.015

.

Table 7. True abilities and estimated abilities, θ, for the first twenty cases in the dataset calculated by two methods: (a) using item parameters and responses, (b) using responses only. *N* is the number of items used in estimating the ability and SE is the standard error in θ.

EID	True	Method (a)			Method (b)		
	Ability	*N*	θ	SE	*N*	θ	SE
1	-1.5841	44	-1.316	0.3579	44	-1.0437	0.2572
2	-1.689	45	-3	2.3198	45	-1.6617	0.3968
3	0.494	45	0.6745	0.2315	45	0.6486	0.2095
4	-0.981	42	-0.9938	0.2936	42	-0.8626	0.2368
5	1.4221	45	1.5675	0.299	45	1.4053	0.2484
6	0.0353	44	-0.1185	0.2288	44	-0.0955	0.2004
7	0.7333	45	0.6075	0.2291	45	0.573	0.2078
8	-0.2385	44	-0.1241	0.2289	44	-0.0956	0.2002
9	-0.5911	45	-1.1117	0.3106	45	-0.979	0.2485
10	-0.6697	45	-0.7136	0.2545	45	-0.6713	0.2148
11	0.0707	45	0.0787	0.2213	45	0.0791	0.1975
12	-0.3552	44	-0.0415	0.2248	44	-0.0388	0.1975
13	1.0341	44	1.2545	0.2731	44	1.1669	0.2349
14	-0.7209	45	-0.5093	0.2377	45	-0.4737	0.2005
15	-1.1992	45	-0.7214	0.2553	45	-0.6441	0.2124
16	0.2684	45	0.5005	0.2259	45	0.4622	0.2056
17	0.6881	44	0.5347	0.2321	44	0.4761	0.2101
18	-0.62	45	-0.4738	0.2355	45	-0.4125	0.1977
19	-0.2659	45	-0.7143	0.2546	45	-0.5908	0.2081
20	0.9098	44	0.7154	0.2369	44	0.6811	0.2142

Table 8. Test items sorted in ascending order by item parameters and item information function (IIF) at several ability levels at and around the pass/fail cut-off of 0. Cells with black background indicate items selected by the optimum pass/fail abductive network model with CPM = 0.5, see Table 3.

IID	Items Sorted By			Items Sorted By IIF at θ =						
	a	*b*	*c*	-1.5	-1	-0.5	0	0.5	1	1.5
1	16	9	32	38	38	38	38	13	39	39
2	13	29	12	33	8	8	8	8	13	9
3	4	39	8	8	33	33	16	16	9	29
4	1	13	29	32	32	12	13	38	29	13
5	21	5	11	12	12	32	21	9	16	34
6	8	18	21	30	30	30	33	39	5	5
7	5	35	24	24	22	21	12	29	34	18
8	42	34	43	19	21	22	32	5	35	35
9	40	11	17	28	28	16	1	4	18	16
10	10	43	14	22	19	1	30	21	42	42
11	22	42	2	21	24	28	22	1	8	11
12	2	36	36	20	1	13	4	42	4	23
13	17	2	44	15	20	15	10	35	40	43
14	7	40	18	1	15	20	28	10	36	36
15	37	17	41	25	16	10	5	40	11	40
16	36	23	45	41	10	4	15	34	43	2
17	35	3	35	10	25	19	9	18	2	17
18	15	31	40	44	41	24	42	22	17	31
19	14	37	22	16	13	44	40	36	10	3
20	6	27	20	26	4	7	39	2	23	4
21	30	6	30	45	44	25	20	17	37	37
22	28	14	9	4	7	5	29	12	1	27
23	18	7	28	7	26	42	7	30	21	10
24	44	16	33	6	45	41	35	37	27	6
25	27	4	1	14	37	40	37	43	3	7
26	43	26	4	37	6	37	2	11	7	14
27	11	45	13	13	14	26	17	33	6	26
28	26	10	42	27	27	6	36	7	31	1
29	20	25	7	31	42	14	18	15	14	8
30	34	41	31	3	40	2	34	32	22	45
31	9	44	15	40	5	27	44	28	26	21
32	12	19	3	42	2	17	6	27	15	25
33	45	15	6	23	17	45	27	6	38	44
34	29	24	26	5	36	36	14	14	30	15
35	32	20	5	2	3	35	19	23	44	22
36	41	1	23	17	35	9	43	26	45	41
37	33	28	27	36	31	18	26	3	25	20
38	3	22	16	35	43	34	11	20	28	28
39	25	30	38	43	34	43	25	44	12	30
40	39	33	34	34	18	39	24	25	20	19
41	23	21	39	11	23	29	41	31	33	24
42	38	12	37	18	11	11	45	45	41	12
43	19	32	19	9	9	3	3	41	32	33
44	24	38	25	39	39	23	23	19	19	38
45	31	8	10	29	29	31	31	24	24	32

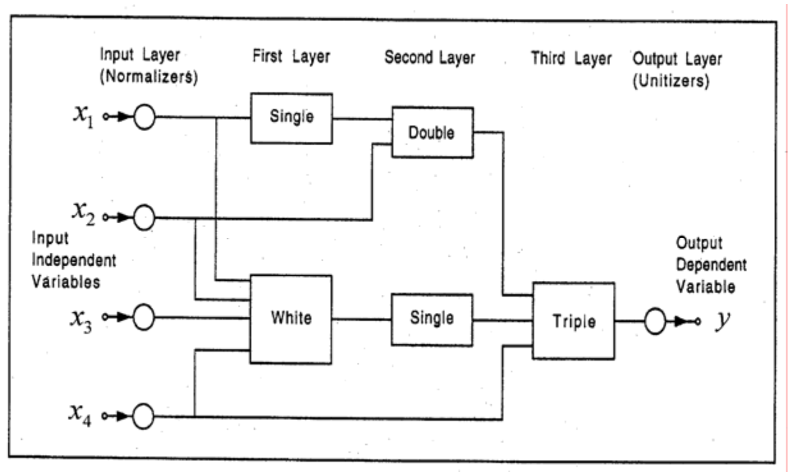


Fig. 1. A typical AIM abductive network model showing various types of functional elements.

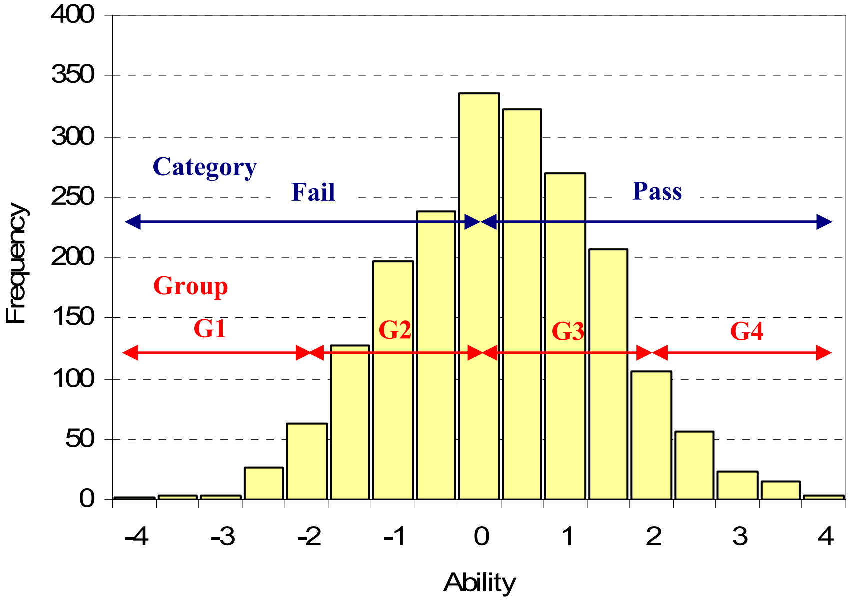


Fig. 2. Frequency distribution of examinees in the dataset over the ability continuum. See Table 2 for details of the categories and groups.

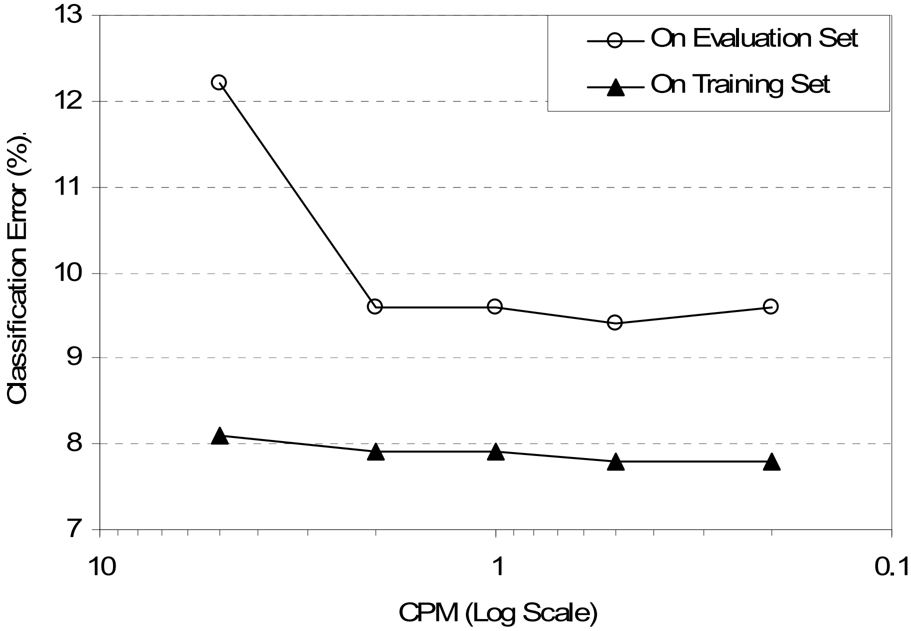


Fig. 3. Classification error versus the CPM parameter for the abductive models in Table 3 on both the training and evaluation sets. Lower CPM values correspond to greater model complexity.

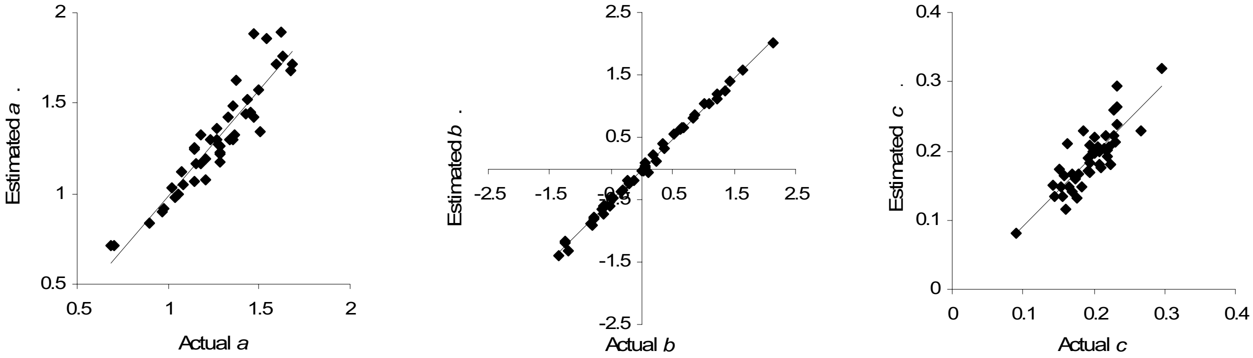


Fig. 4. Correlation between actual and estimated values for the parameters *a*, *b* and *c*. Estimation is carried out using true abilities and response patterns (method a). Correlation coefficients are 0.930, 0.998 and 0.831 respectively.

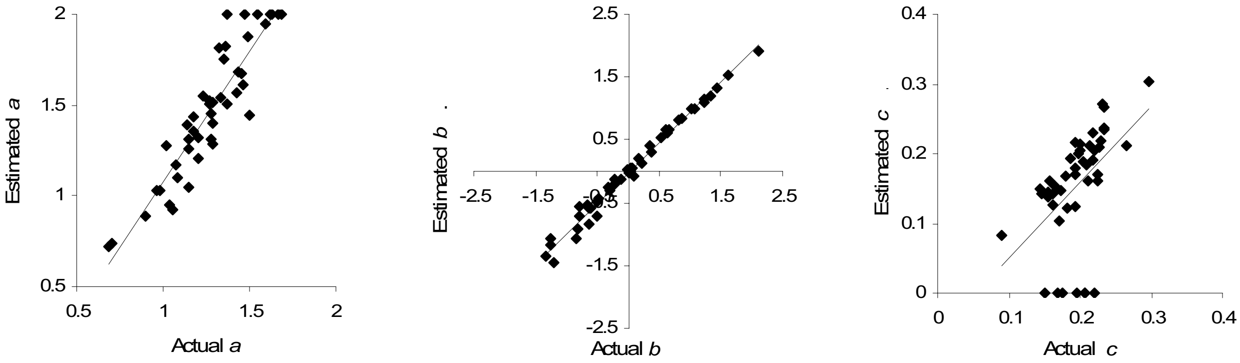


Fig. 5. Correlation between actual and estimated values for the parameters *a*, *b* and *c*. Estimation is carried out using response patterns only (method b). Correlation coefficients are 0.918, 0.993 and 0.496 respectively.

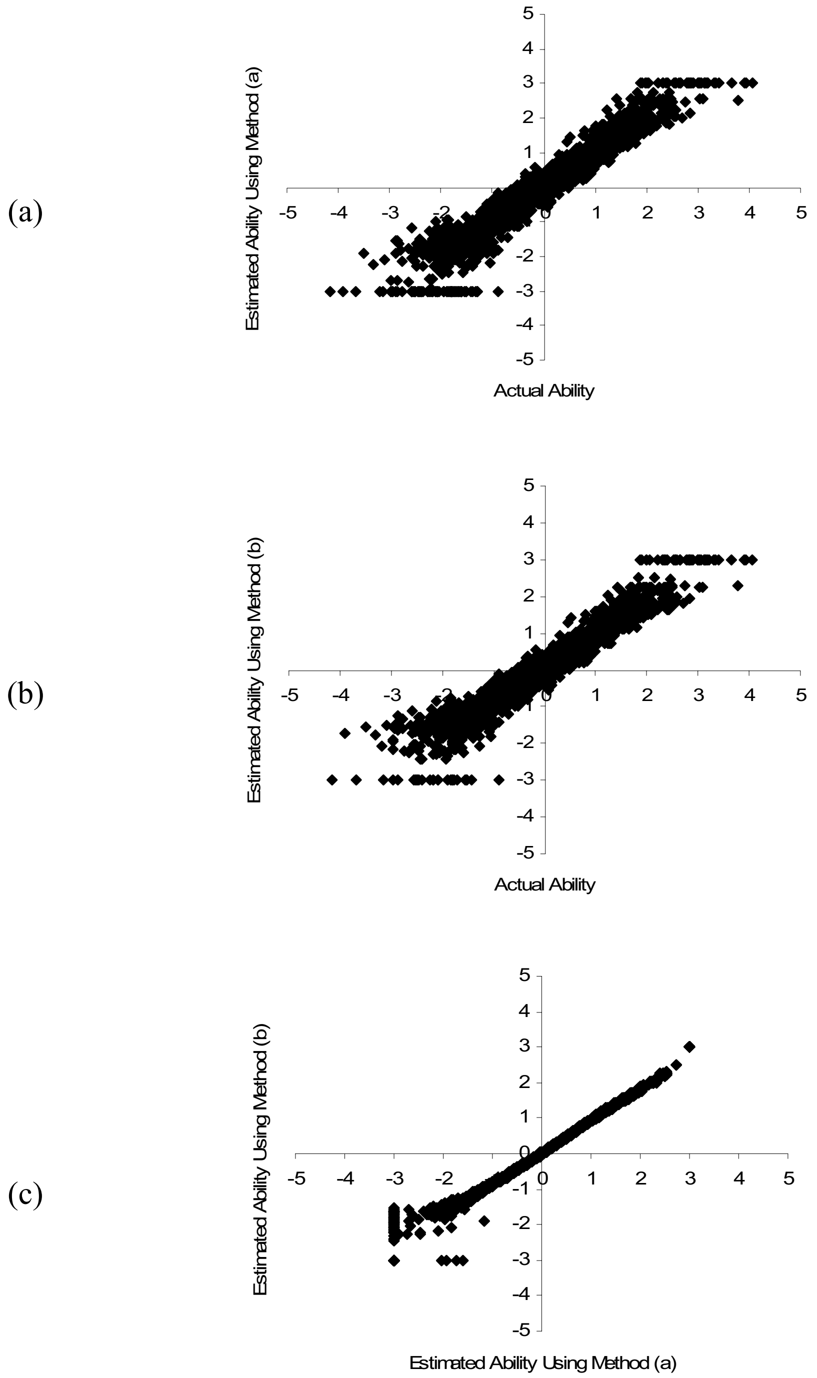


Fig. 6. (a) Correlation between true ability and estimated ability using original item parameters and response patterns (b) Correlation between true ability and estimated ability using response patterns only (c) correlation between the two estimated ability parameters. The correlation coefficients are 0.960, 0.958 and 0.991 respectively.

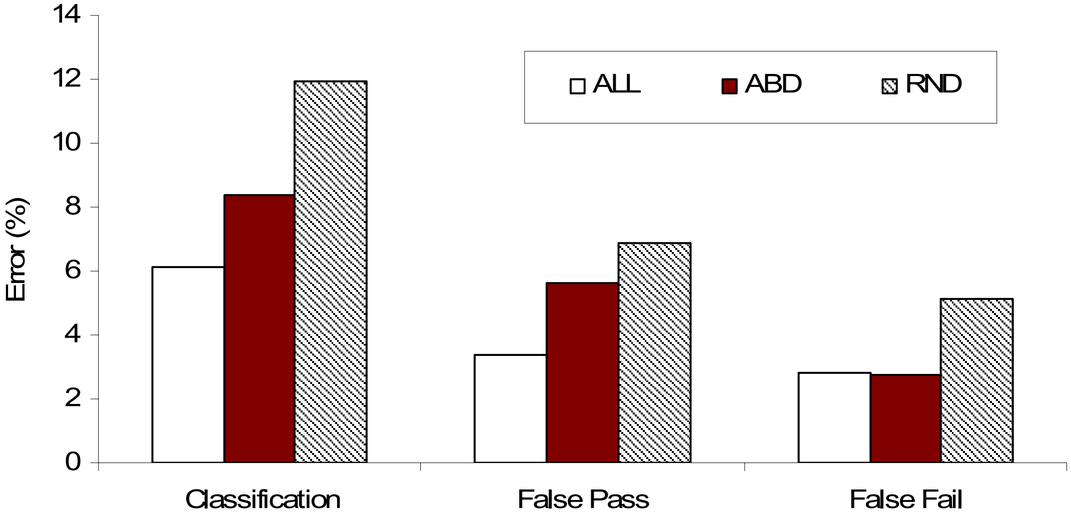


Fig. 7. Classification errors for three pass/fail tests. ALL: test composed of all 45 test items; ABD: test composed of the 12 items selected as inputs for the optimum abductive network model with CPM = 0.5; RND: test composed of a randomly selected subset of 12 items {4, 8, 10, 12, 13, 22, 23, 27, 28, 31, 37, 43}.

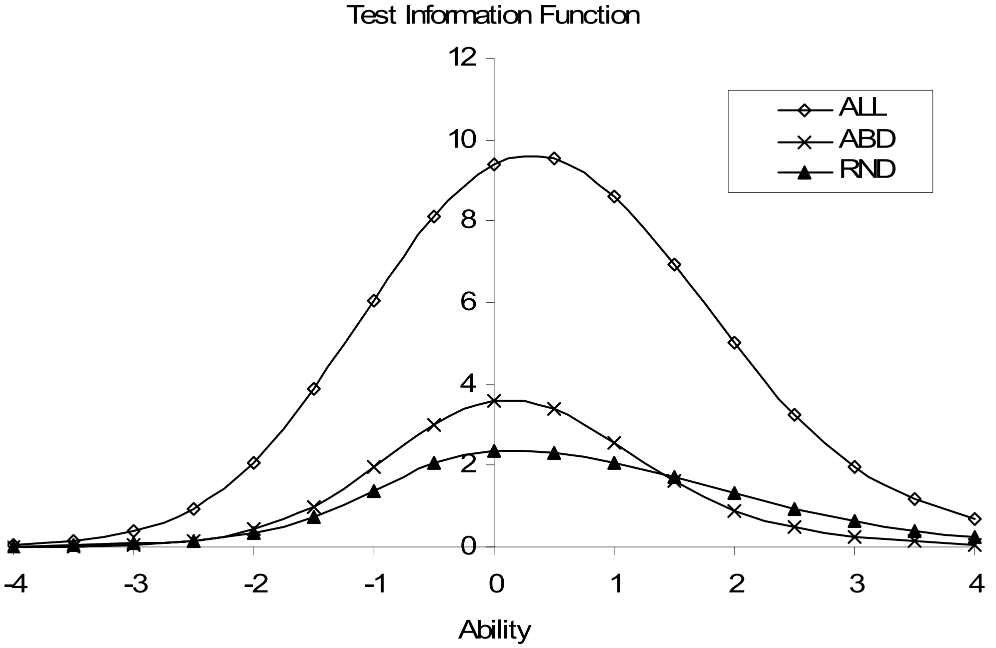


Fig. 8. Test information functions for the three tests of Fig. 7. ALL: test composed of all 45 test items; ABD: test composed of the 12 items selected as inputs for the optimum abductive network model with CPM = 0.5; RND: test composed of a randomly selected subset of 12 items {4, 8, 10, 12, 13, 22, 23, 27, 28, 31, 37, 43}.

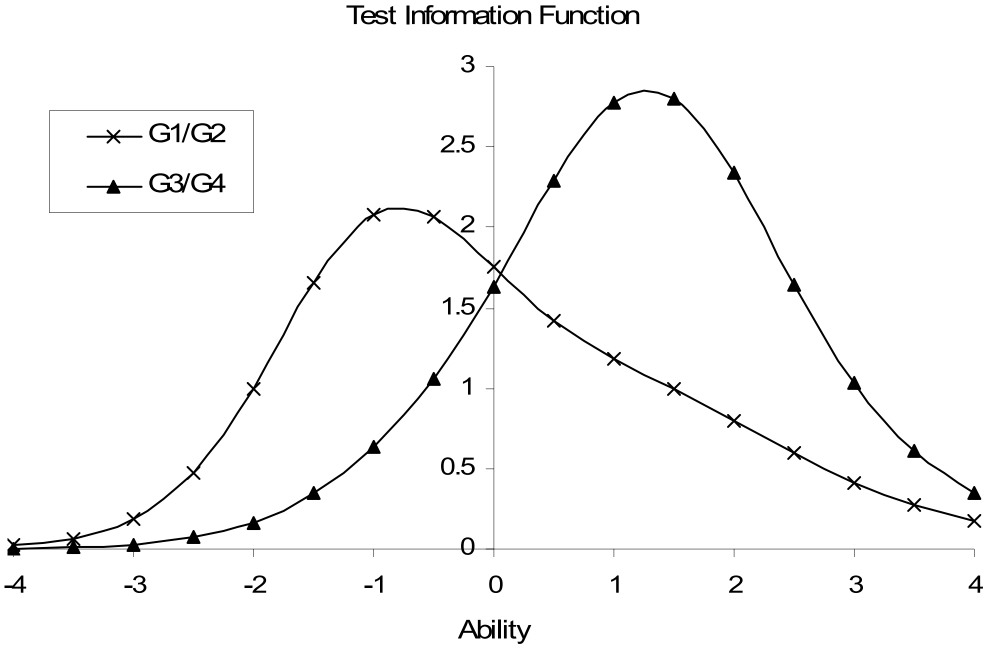


Fig. 9. Test information functions for the test items selected by abductive machine learning for the two models that perform G1/G2 and G3/G4 classification within the fail and pass groups, respectively. See Fig. 2, Table 2, and Table 5.