International Journal of Testing, 20: 78–96, 2020

ISSN: 1530-5058 print / 1532-7574 online
DOI: 10.1080/15305058.2018.1551225

Routledge
Taylor & Francis Group

Log Data Analysis with ANFIS: A Fuzzy Neural Network Approach

Ying Cui and Qi Guo
University of Alberta, Canada

Jacqueline P. Leighton
Educational Psychology, University of Alberta, Canada

Man-Wai Chu
University of Calgary, Canada

This study explores the use of the Adaptive Neuro-Fuzzy Inference System (ANFIS), a neuro-fuzzy approach, to analyze the log data of technology-based assessments to extract relevant features of student problem-solving processes, and develop and refine a set of fuzzy logic rules that could be used to interpret student performance. The log data that record student response processes while solving a science simulation task were analyzed with ANFIS. Results indicate the ANFIS analysis could generate and refine a set of fuzzy rules that shed lights on the process of how students solve the simulation task. We conclude the article by discussing the advantages of combining human judgments with the learning capacity of ANFIS for log data analysis and outlining the limitations of the current study and areas of future research.

Keywords: ANFIS, fuzzy inference system, large-scale assessment, log data analysis, technology-based assessment

Rapid progress in digital technology has opened the door to the development of modern assessments with integrative, interactive, performance-based tasks that allow students to demonstrate complex competencies and higher order thinking skills. International and national large-scale assessment agencies have begun

Correspondence should be sent to Dr. Ying Cui, University of Alberta, Department of Educational Psychology, Centre for Research in Applied Measurement and Evaluation, 6-110D Education Bldg, Edmonton, AB T6G 2G5, Canada. E-mail: yc@ualberta.ca
Color versions of one or more figures in the article can be found online at www.tandfonline.com/hijt
This article has been republished with minor changes. These changes do not impact the academic content of the article.

experimenting with the use of technology in their assessment programs. For example, in 2006, the Program for International Student Assessment (PISA) for the first time pioneered in Denmark, Iceland and Korea a computer-based assessment in Science to "administer questions that would be difficult to deliver in a paper-and-pencil test—the relevant questions included video footage, simulations and animations" (OECD, 2010, p. 17). The 2009 National Assessment of Educational Progress (NAEP) science test tested about 2,000 students with interactive computer tasks to assess "how well students can perform scientific investigations, draw valid conclusions, and explain their results" (National Center for Educational Statistics, 2012, p. 2).

One of the most promising functions of technology-based assessment is its ability to record the history of everything a student does through the course of the assessment (Mayrath, Clarke-Midura, & Robinson, 2012). These highly detailed student data, often called log data, capture every action a student engages in during problem solving, which can help understand the process of how the student arrives at a conclusion, and therefore have the potential to provide valuable information on problem-solving strengths and weaknesses. Log data has potential as a valuable data source; however, the question of how to best analyze and interpret log data remains unanswered (Mayrath et al., 2012). Log data may contain hundreds of actions for a given student and some actions may be haphazard and unimportant in making inferences about the student. How do we know which data are relevant and useful? How do we make use of the huge amount of data to extrapolate meaningful patterns and inferences in ways that will provide reliable, valid, and useful information about student learning and problem solving? Traditional measurement models such as Classical Test Theory or Item Response Theory initially designed for analyzing highly structured multiple-choice items and written constructed responses are no longer adequate (Quellmalz, Timms, Buckley, Davenport, Loveland, & Silberglitt, 2012). To fulfill the full potential of technology-based assessments, methods of analyzing and interpreting log data must be developed to make the best use of all available data sources so fine-grained inferences about student learning and problem solving can be produced.

This study explores the use of the Adaptive Neuro-Fuzzy Inference System (ANFIS; Jang, 1993), a neuro-fuzzy approach, to analyze the log data of technology-based assessments to extract relevant features of student problem-solving processes, and develop and refine a set of fuzzy logic rules that could be used to interpret student performance. The term fuzzy logic was first introduced by Zadeh (1965) to mathematically model imprecision and uncertainty in data and problems that involve reasoning and problem solving. Unlike classical or Boolean logic in which each statement is either true (1) or false (0), fuzzy logic permits each statement to be associated with a fuzzy value between

0 and 1 to represent the degrees of truth, which is mathematically calculated through a membership function. For example, the statement "the temperature is warm" in classical logic would be either true or false. However, in fuzzy logic, each linguistic label such as warm or cold is associated with a separate membership function (e.g., Gaussian function). A temperature of 20° C could be associated with the linguistic label "warm" with a membership value of 0.8 and also be associated with the linguistic label "cold" with a membership value of 0.1.

A fuzzy inference system is a process to infer a set of fuzzy if-then rules to map data from input variables associated with the premise part of the rules to the output variables corresponding to the consequent part of the rules. The first step of a fuzzy inference system is to calculate the membership value of each linguistic label associated with each input variable using the membership function. Next, fuzzy rules are formed by connecting the input variables through the "AND" or "OR" operator. The firing strength of each rule is then calculated based on the membership values of the corresponding linguistic labels of input variables specified in the premise part of the rule (e.g., the product of the membership values). Finally, the consequent of each rule is calculated depending on the firing strength and the rules are aggregated to produce the final output value(s). Fuzzy rules mimic the process of human decision making and reasoning with uncertainty and imprecision. Although fuzzy rules can be defined by human knowledge to associate the premise with consequent, the fuzzy inference system could benefit from the further refinement of fuzzy rule parameters based on the observed data. ANFIS (Jang, 1993) is a fuzzy inference system implemented with the feedforward artificial neural network so as to utilize the learning capacity of the artificial neural network to refine a set of human-specified fuzzy if-then rules through minimizing a prescribed error measure.

The current study employs ANFIS to analyze the log data that record student response processes while solving a simulation task. ANFIS is a promising approach to log data analysis because of its ability to incorporate human knowledge into the artificial neural networks to refine the set of human-generated rules with observed data. Although some machine learning techniques such as a typical artificial neural network (e.g., the multilayer perceptron) or support vector machines can be very powerful in terms of its modeling capability as a highly flexible nonlinear statistical technique for mapping complex relationships between inputs and outputs, the interpretability of the results is often minimal. For example, a typical neural network can predict from inputs to outputs up to an arbitrarily chosen precision. However, the network functions like a "black box" in the sense that examining its structure would not provide much insight on the actual form of the relationship between inputs and outputs. Even whether it is possible to determine which input variables

are more important in terms of predictive power is controversial (Masters, 1994). This poses a tremendous disadvantage for the use of artificial neural networks for the analysis of student log data as it is valuable to know which student steps/actions are more associated with successful problem solving for the purpose of instruction and remediation. ANFIS overcomes this problem by explicitly building a set of human-specified rules into the structure of artificial neural networks. ANFIS combines the qualitative aspect of human knowledge and reasoning process with the quantitative analysis of observed data, which makes the interpretation of results to be intuitive and is especially valuable in the context of educational assessments. The simulation task used in this study was initially developed for the 2009 NAEP science test to assess students' ability to perform scientific investigations, draw valid conclusions, and explain their results. This simulation task was administered to grade 8 students in a Canadian city. The goal of the study was to construct a set of fuzzy if-then rules to extract useful information from log data that can help understand and interpret student problem-solving strategies. The fuzzy rules are constructed in two steps: (1) a set of rules are identified through task analysis along with a review of student log data; and (2) the identified fuzzy rules are further improved by training student log data with ANFIS to optimize the associated parameters so as to produce the smallest possible prediction error. The next section provides an overview of ANFIS and its corresponding learning procedure of tuning model parameters. The following two sections describe the methods and results of our log data analysis with ANFIS. Finally, the advantages of combining human judgements with the learning capacity of ANFIS for log data analysis are discussed and limitations of the current study and areas of future research are outlined.

AN OVERVIEW OF THE ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM

ANFIS models a fuzzy inference system with a multilayer feed-forward network, where the nodes are arranged in a layered architecture. Each node is characterized by a mathematical function that performs operations on its incoming signals and generates a corresponding output. There are two types of nodes in ANFIS: a square node or a circle node. The function associated with a square node has adaptive parameters that could be modified based on the ANFIS learning algorithm. The modification of these parameters will change the node function and consequently the overall behavior of the adaptive system. A circle node does not have adaptive parameters and therefore its function is fixed. A typical ANFIS consists of five layers where the nodes in a particular layer do not directly interact with one another, but only connect with

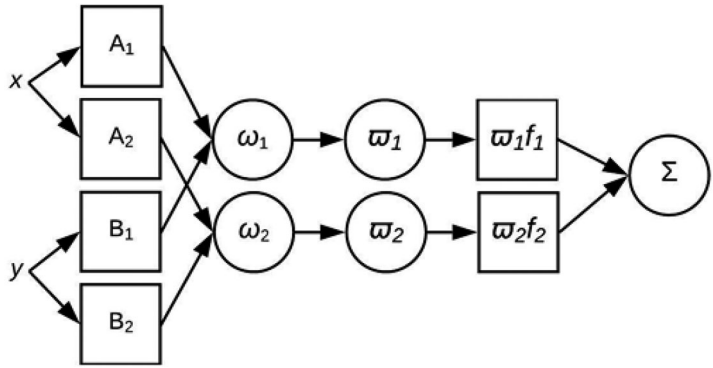

FIGURE 1
Two-input and one-output ANFIS with two rules.

those in the adjacent layer(s). Figure 1 shows an example of a two-input, one-output ANFIS architecture with two fuzzy if-then rules where rule 1 specifies "if x is A_1 and y is B_1, then $f_1 = p_1x + q_1y + r_1$" and rule 2 specifies "if x is A_2 and y is B_2, then $f_2 = p_2x + q_2y + r_2$."

The first layer of ANFIS has adaptive nodes with the membership functions as the node functions. This layer transforms the values of input variables into fuzzy values between 0 and 1 through the membership functions. In Figure 1, for example, there are two input variables x and y, each with two linguistic labels (i.e., A_1 and A_2; B_1 and B_2), and therefore a total of four nodes in layer 1. The membership functions receive the incoming signal x (or y), and produces the degree to which the given value of x (or y) satisfies the linguistic label A_i (or B_i). A commonly used membership function is the generalized bell shaped function with the maximum value equal to 1 and the minimum value equal to 0, such as

$$\mu_{A_i} = \frac{1}{1 + \left[\left(\frac{x - c_i}{a_i}\right)^2\right]^{b_i}},$$

where a_i, b_i, and c_i are the parameters for the node of A_i. c_i and a_i determine the center and width of the membership function, while the value of $-b_i/2a_i$ is the slope at the point of the membership value equal to 0.5. An example of bell shaped membership functions is presented in Figure 2. Layer 1 of the ANFIS architecture in Figure 1 has a total of 12 parameters (i.e., 4 nodes each with 3 parameters), which are referred to as "premise parameters."

The second layer of ANFIS has fixed nodes and each node represents a fuzzy rule. As a result, the number of nodes in this layer is equal to the number of fuzzy rules in the fuzzy inference system, which could be specified by

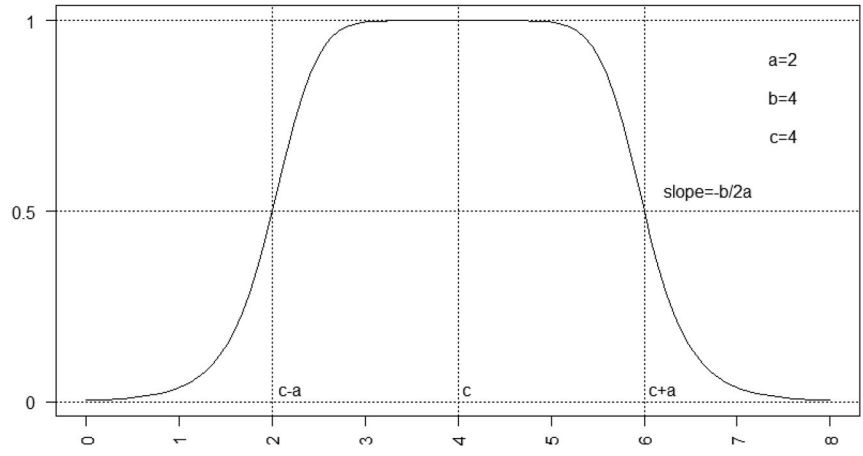

FIGURE 2
An example of bell shaped membership function.

human knowledge (i.e., 2 in this example) or simply equal to all the possible combinations of linguistic values from input variables (e.g., $2^2 = 4$ in the case of two input variables and each with two linguistic values). The incoming signals of each node are the membership values associated with the linguistic labels connected to the node. The output of each node is the product of the membership values, which indicates the weight or firing strength of the corresponding fuzzy rule. For instance,

$$w_1 = \mu_{A_1} \times \mu_{B_1},$$

where w_1 is the firing strength of rule 1 associated with linguistic labels A_1 and B_1.

The third layer of ANFIS normalizes the firing strengths of the rules by computing the ratio of each rule's firing strength relative to the sum of all rules' firing strengths. For example,

$$\overline{w}_1 = \frac{w_1}{w_1 + w_2},$$

where $\overline{w}_1$ is the normalized firing strength of rule 1. All the nodes in the third layer are fixed.

The forth layer of ANFIS contains square nodes that multiply the output of each rule f_i by its normalized firing strength $\overline{w}_i$ calculated from the third layer. For example, for the first rule, the node function calculates

$$\overline{w}_1 f_1 = \overline{w}_1 (p_1 x + q_1 y + r_1),$$

where p_1, q_1, and r_1 are the node parameters that need to be estimated and adjusted by the learning algorithm. These parameters are referred as "consequent parameters" because they are associated with the consequent part of the rules.

Finally, the fifth layer of ANFIS produce an overall output by summing directly the incoming signals from layer four. That is,

$$Overall\ output = \overline{w}_1 f_1 + \overline{w}_2 f_2.$$

To predict the values of the overall output for ANFIS, the premise parameters and the consequent parameters must be estimated. To achieve the best prediction, it is natural to minimize the errors between the actual and predicted values of the final output nodes. Typically, the process of error function minimization begins with random assignment of the initial estimates of the premise and consequent parameters. The output values of the system can be computed using these initial estimates together with the values of input variables. The calculated output values are then compared to the observed values, and the parameters are modified accordingly to make the value of the error function decrease as much as possible.

The most studied and used method for adjusting the parameters in the framework of artificial neural network is the *back propagation algorithm* (Rumelhart, Hinton, & Williams, 1986; Rumelhart & McClelland 1986) in which each parameter is modified by an increment that is proportional to the negative gradient of the error function (i.e., the partial derivatives of the error function with respect to the unknown parameters). This is because when the parameters changes in the direction of the negative gradient of the error function, then the error function decreases most rapidly. The back propagation algorithm is an iterative procedure. Within each iteration, a data point consisting of values of input and output variables is randomly selected and presented to the network. The result for the output nodes are then compared with the observed values from the corresponding output variables. Each parameter estimate is modified accordingly. Once the estimates have been adjusted, another data point is selected and presented, and a new iteration starts. To update the parameter estimates, the error function is evaluated for each data point. The process is repeated until the error term or the parameter estimates have converged to an acceptable level around zero or the maximum number of iterations is reached**.**

Jang (1993) proposed a hybrid algorithm that combines the back propagation algorithm with the least square estimator to update the parameters of ANFIS. Least square estimator can be used in ANFIS because the overall output nodes can be expressed as a linear combination of the consequent parameters given the values of premise parameters where the least square estimates

exist. In this case, each iteration of the hybrid algorithm includes a forward pass and a backward pass. In the forward pass, data are presented to the network and values of the nodes in layers one through three are calculated. Least square estimates of the consequent parameters in layer four are then identified to minimize the error function. Given the values of the consequent parameters, in the backward pass, the premise parameters are updated through the back propagation algorithm where each premise parameter is modified by an increment that is proportional to the negative gradient of the error function. The advantage of the hybrid algorithm is to reduce the number of parameters that need to be adjusted in the back propagation method during each iteration, which consequently cuts down the convergence time significantly.

METHODS

Participants

Data used in this project were collected for a larger study (Chu, 2017) investigating factors that can enhance students' learning and performance on technology-based assessments in science. A total of 193 students from 14 Grade 8 science classes in Alberta, Canada, have given consent to participate in this study. Ninety five students self-identified as male (49%), 89 as female (46%), and 9 did not disclose gender (5%). The students represented more than 11 ethnicities with a majority of them indicating they were Caucasian (38%). All students were between 13 and 14 years of age at the time of data collection. One hundred and eighty seven students (97%) self-disclosed that they had access to a computer at home.

Simulation Task

Students were asked to solve a series of three science simulation problems, which were originally created in the NAEP Problem Solving in Technology-Rich Environments study (Bennett, Persky, Weiss, & Jenkins, 2007). The simulation problems, presented with a scenario involving a helium balloon, require students to design and conduct experiments to deduce the relationship between payload mass, the amount of starting helium, and the altitude of a balloon. The targeted knowledge and skills are highly relevant to the local science program in Alberta. The simulation problems utilize the ideas of the particle model of matter which is covered in the Grade 8 science unit named Mix and Flow of Matter (Alberta Education, 2014). Additionally, some of the material used in the simulation was taught during Grade 6 science in the units Air and Aerodynamics and Flight (Alberta Education, 1996). The skills required by the tasks, moreover, are well aligned with the key targeted skills outlined in the

junior high (i.e., Grades 7–9) science program-of-study (Alberta Education, 2014) where students are expected to develop the skills required for scientific and technological inquiry, for solving problems, for making informed decisions, and for communicating scientific ideas and results. Hence, the simulation problems are well suited for Grade 8 students in Alberta by allowing them the opportunity to assess their applications of Grade 8 content knowledge and skills. During the simulation, students have the option to design an experiment, manipulate parameters, run their experiment, record the data, and graph the results before they reach a conclusion. For each student, all actions and the time used for each action were logged and saved for later analysis. For the purpose of this study, log data related to the first simulation problem was used.

Data Analysis

The analysis of the log data recorded while students were working on the simulation problem was completed in two steps. In the first step, a task analysis was performed by two reviewers who have sufficient knowledge and expertise in the content area of the simulation task to (1) identify the key knowledge, skills, and processes required by the simulation, (2) review student log data to generate hypotheses of how students solve the task and identify strategies utilized by students in their problem solving, and (3) construct fuzzy if-then rules that could help explain student performance. The input variables of the fuzzy if-then rules are student knowledge and skills required by the task, which are reflected through their actions/steps in the log data. The rules are formed by identifying the combinations of linguistic values of the input variables (e.g., if A is high and B is high) that could lead students to solve the problem of the simulation task—to figure out how different payload masses affect the altitude of the balloon (i.e., the output variable of the fuzzy rules). The two reviewers first completed the task analysis independently, and then they met as a group to discuss their results and resolve the disagreements so as to finalize the set of fuzzy if-then rules for further analysis in step 2.

In the second step, ANFIS is utilized to refine the identified rules in step 1. Specifically, the membership function parameters and the consequent parameters were tuned based on the actual log data so as to maximize the prediction power of the fuzzy rules. To explore the capability of ANFIS to identify rules based on log data in a more exploratory way, a second ANFIS analysis was conducted where the second layer of the network includes all the possible fuzzy rules formed by each possible combination of linguistic values from input variables. This type of analysis might be useful when one is concerned that the rules specified based on human judgement may not be able to capture

all the variations of student problem solving or when rules from human judges are not available.

In statistical modeling, overfitting is problematic because the model may mistakenly treat random noises as part of the true relationship among the variables. As a result, the model fits really well the sample from which the model was derived but fails to generalize to other samples. To test whether overfitting occurs, we randomly split the data into two samples, one for the training of ANFIS to estimate network parameters ($n = 150$) and the other used for the purpose of cross validation ($n = 43$).

To compare the performance of ANFIS with other popular machine learning techniques, the log data of the simulation task were analyzed with artificial neural networks, logistic regression, and support vector machines. Two measures were used as the basis of the comparison: precision (i.e., the percentage of true positives with respect to the total number of model-predicted positives) and recall (i.e., the percentage of true positives with respect to the total number of positives in the sample). The values of precision and recall were calculated with both training and validation data using each of the four methods, respectively. The analyses of ANFIS, logistic regression, and support vector machines were conducted with the MATLAB program (MathWorks, 2017). Artificial neural network analysis was conducted with SPSS 24 (IBM Corp, 2016).

RESULTS

Step 1: Task Analysis

The task analysis of the simulation helped identified the steps each student needed to take in order to successfully complete the task. During the simulation, students were first told that their tasks were to use a simulation tool to solve a set of problems related to balloon science, along with a brief introduction of simulation as a scientific tool. Next, a tutorial of the current simulation was provided by presenting the research questions, how to design simulation experiments by selecting different amount of payload mass, how to make predictions before conducting the simulation, how to run the experiments and observe the results, and how to construct tables and graphs to visualize the results. After the tutorial, the research question of the first simulation task was presented to students: "How do different payload masses affect the altitude of a helium balloon?" In order to solve this problem (assuming students had no prior knowledge about this problem), students need to conduct one or more experiments by choosing different payload masses ranging from 10 lbs. to 90 lbs., run these experiments, and based on the results of their experiments construct a table or graph with variable payload masses as one variable and altitude as the other variable. Finally students were prompted to the question

"*Based on your experiments, which statement most accurately and completely describes how different payload masses affect balloon altitude?"* where students could choose from a set of four statements.

Based on the task analysis and review of log data, reviewers extracted three key variables about student behaviors/actions that could explain and predict student task performance in the simulation. These three variables were (1) the number of experiments run by the student, (2) the number of tables constructed by the student with the correct variables (i.e., payload masses and altitude), and (3) the number of graphs constructed by the student with the correct variables (i.e., payload masses and altitude). There are four issues worth mentioning here. First, it should be noted that for each experiment, students could select different payload masses for the experiment. However, we considered that the actual quantity of mass a student selected for the experiment did not matter very much. Rather the total number of unique experiments a student run was important: the more experiments the student run, it would be easier for him/her to see the relationship between payload masses and the altitude of the helium balloon. Second, when a student constructed tables (or graphs) with variables not related to the research question, these tables (or graphs) would not help much for solving the task. Therefore, information regarding tables (or graphs) with irrelevant variables was deemed unimportant. Third, although during the simulation, students could access the glossary, science help, and computer help, very few students used these options and most students who clicked these buttons exited the screen immediately without picking any topics/terms. As a results, these actions were not extracted for further analysis. Fourth, students were asked to make predictions prior to each experiment. However, majority of the students choosing the "I don't know" option and therefore this action was not considered to be informative and therefore excluded from the analyses.

With the three variables extracted from log data, a total of four rules were formed by reviewers to provide insight on the process of students generating conclusions regarding the research question of how different payload masses affect balloon altitude:

1. If the number of experiments run by the student is small, then the chance of the student solving the task is low;
2. If the number of experiments run by the student is large and the number of tables constructed by the student with related variables is large, then the chance of the student solving the task is high;
3. If the number of experiments run by the student is large and the number of graphs constructed by the student with related variables is large, then the chance of the student solving the task is high; and

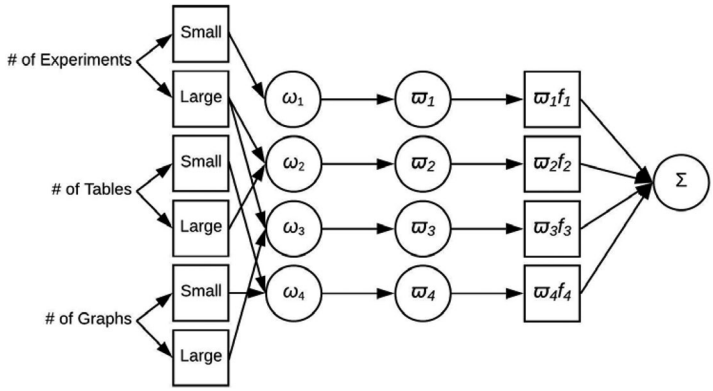

FIGURE 3
Three-input ANFIS with four rules.

4. If the number of tables constructed by the student with related variables is small and the number of graphs constructed by the student with related variables is small, then the chance of the student solving the task is low.

In the next step, ANFIS was used to refine these four rules. Specifically, membership functions can be hypothesized and tuned to assign membership values (i.e., a value between 0 and 1 to reflect the degree of truth) to a student for each linguistic label (e.g., small or large) based on his/her value of the input variable (e.g., the student run a total of 5 experiments). The output variable of each rule was also adjusted based on the actual log data.

Step 2: ANFIS Analysis

Figure 3 presents the ANFIS model with the four rules identified in step 1. The input variables of the ANIS model were the three variables extracted from log data based on the task analysis and review of log data, where the output of the network was whether or not a student produced a correct answer to the question presented at the end of the simulation task. Each input variable was associated with two linguistic labels, "small" and "large." The membership functions were specified as the generalized bell shaped functions. The training data with 150 students was used to fit the ANFIS model. The maximum number of epochs was set at 500. The adequacy of the sample size required for the ANFIS analysis depends largely on the complexity of the problem. For the current analysis, there are three input variables, each with two linguistic labels (i.e., large and small), and therefore a total of six nodes in layer 1 of the

ANFIS model. Using the generalized bell shaped membership functions, each with three parameters, a total of 12 premise parameters need to be estimated. The output of each rule was set to be a constant and therefore a total of six consequent parameters to be estimated. Given the simplicity of the current ANFIS model, the sample size was considered as sufficient. In addition, an inadequate sample size often leads to an overfitting model in which too many parameters need to be estimated from too small a sample. To evaluate whether the model overfit the data, the values of precision and recall were calculated, which will be presented momentarily for both training and validation data.

Figure 4 depicts the final membership functions for the three input variables, respectively. For the input variable "# of experiment", the range of the input values is from 1 to 16, and the crossover point of the two membership function has an x value close to 2, suggesting if a student conducted 2 or more experiments, his/her membership value for "large" would be higher than that for "small." The more experiments a student conducted, the larger the discrepancy between the membership values for "large" and "small." For the input variables "# of tables" and "# of graphs," the final membership functions showed that if a student constructed at least one table (or graph) with relevant variables, the membership values for "large" would be higher than that for "small." And the difference in values increased as the student constructed more tables (or graphs) with relevant variables.

The output value of each rule was estimated as $f_1 = 0.02$, $f_2 = 0.60$, $f_3 = 1.04$, and $f_4 = 0.07$ respectively. It should be noted that the output values of each rule was not restricted to the range from 0 to 1. However, the comparison of the magnitude of these values revealed that the pattern was consistent with our expectations of the consequent part of the four rules. That is, rules 2 and 3 tend to be associated with the increased chance of solving the problem. The output value of each rule was then weighted by the normalized firing strength of rule and summed across all the rules to produce the final output of the network. For each student, the final output value was then used as basis to predict whether student can solve the problem or not: if the value was closer to 1, then predict "yes"; otherwise "no." The prediction was then compared to the correctness of the student's actual response to the research question. In this way, the values of precision and recall for the training set can be estimated, which was equal to 75.00% and 88.73%, respectively. Results suggest that, based on the three student variables extracted from log data, 75% model-predicted correct performances are true positives, and 88.73% of the correct responses in the sample are predicted by the model. To evaluate whether the model overfit the training data, cross validation was performed on the validation data with 43 students. The values of precision and recall was found to be 76.00% and 90.48%, respectively, similar to those calculated for the training

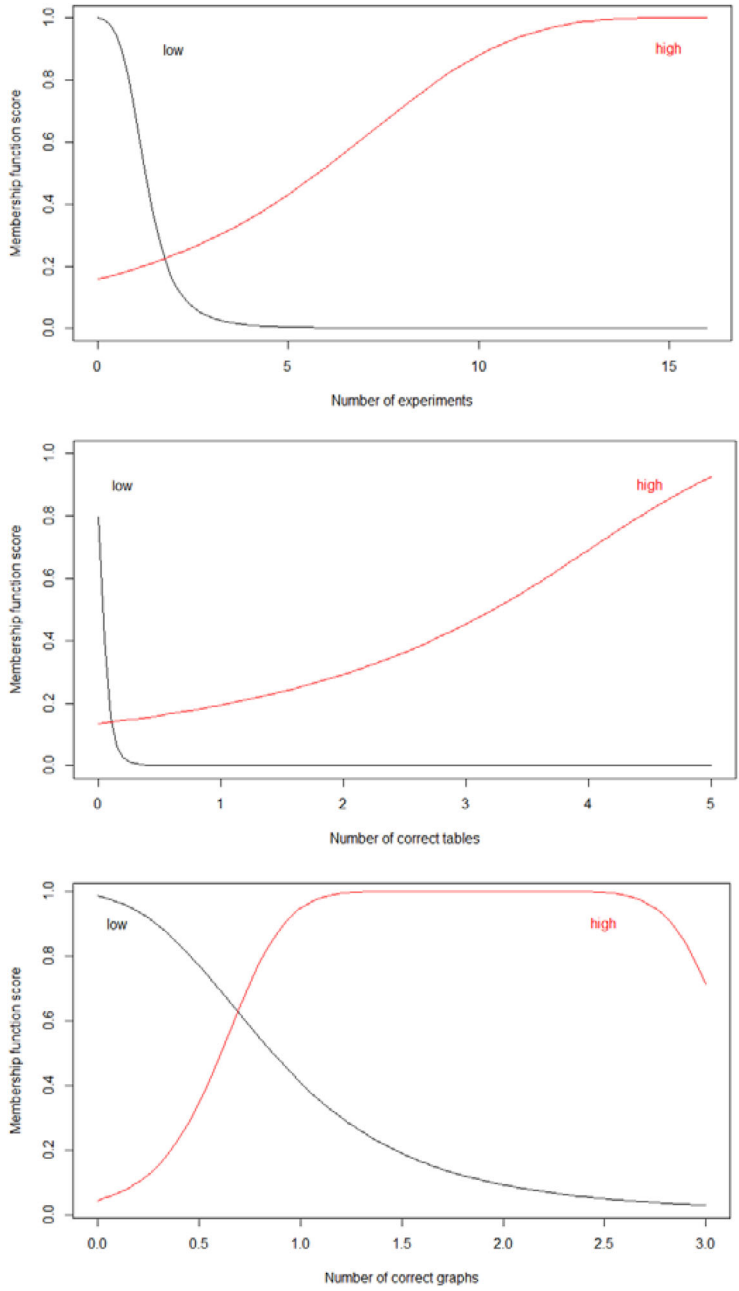

FIGURE 4
Final memberships functions for the three input variables of ANFIS.

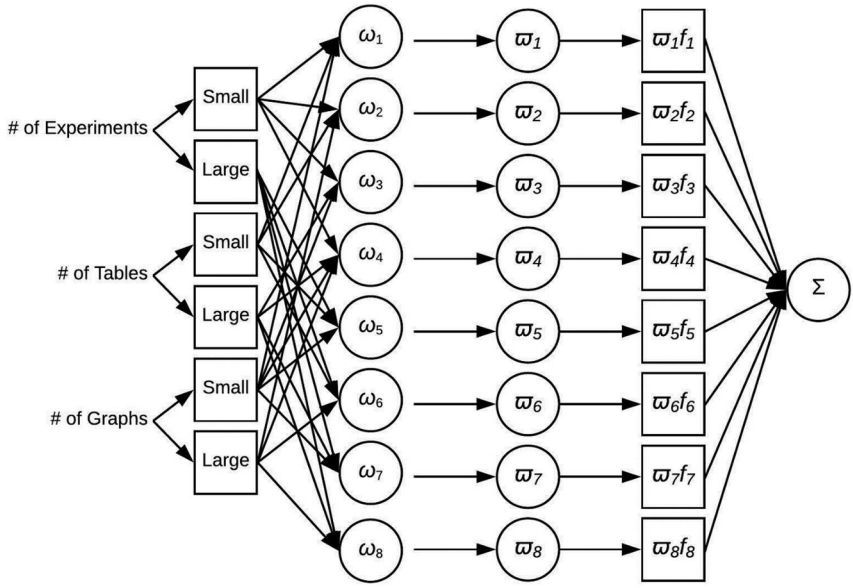

FIGURE 5
Three-input ANFIS with eight rules.

data, indicating no major concerns of overfitting, which also verified the adequacy of the sample size in the train data.

A second ANFIS analysis was conducted to examine the performance of a less restrictive model (see Figure 5) where all the combinations of linguistic labels across the three input variables (i.e., a total of $2^3 = 8$ rules) were included in the model. The model showed very similar performance compared to the model with four rules. The values of precision and recall were 76.54% and 87.32% for the training data, and 76.00% and 90.48% for the validation data. This result suggested that the model with four rules had similar level of prediction power as the full model with eight rules. Therefore, the rules generated based on human knowledge were able to adequately account for the variability underlying the data.

For the purpose of comparison, analyses of logistic regression, artificial neural network, and support vector machines were also conducted using the three student variables extracted from the log data to predict whether the student can solve the problem. For the artificial neural network analysis, the number of hidden nodes were determined empirically. That is, the neural networks with zero hidden nodes was first trained and the value of the error function was recorded. The number of hidden nodes was then increased by one

TABLE 1
Precision and Recall for ANFIS, Logistic Regression, Support Vector Machines, and Artificial Neural Network

		ANFIS	Logistic	ANN	SVM
Training data	Precision	75.00%	73.91%	76.00%	81.82%
	Recall	88.73%	71.83%	77.03%	88.73%
Validation data	Precision	76.00%	72.73%	76.47%	79.17%
	Recall	90.48%	76.19%	68.42%	90.48%

and the process was repeated. It was found that the error became stable after three hidden nodes. As a result, three hidden nodes were used in the final model. For the support vector machines analysis, a kernel function is used to map the data from the original space into a new feature space and finds an optimal decision boundary with the maximum margin from data in the two categories of the output variable. The radial-based kernel function was found to produce the best performance and therefore was adopted in the final model. For each of the models considered in our analysis, the values of precision and recall with the training and validation data were presented in Table 1. It was found that ANFIS outperformed logistic regression and artificial neural network in terms of both precision and recall for the training and validation data, and support vector machines showed comparable performances on recall but slightly superior performance in terms of precision (79.19% vs. 76% for the validation data).

In addition, in order to explore whether the time-related variables could add additional information to the prediction of student performance, the effect of time duration for task completion on student performance was investigated but found to be minimal. For example, with logistic regression, time duration was added as a fourth predictor in the model, in addition to the three student variables identified through task analysis. It was found that according to the Wald criterion, time had no significant effect on student performance after controlling for the other three variables, $\chi^2(1) = 2.55,\ p = 0.11$. Similarly, the time duration variable also showed negligible impact on the predictive accuracy for other models. As a result, the analysis of time-related variables was not further pursued.

DISCUSSION

Technology-based assessments have the potential to reform the types of learning that can be assessed as well as the type of information that can be gathered as evidence for interpreting learning. These assessments can record every step a student performs while solving the problem, which has the potential to offer

insights on how students arrive at a conclusion. This information may provide valuable information on the strengths and weaknesses of student problem solving strategies. Although technology-based assessment has received increased attention from educators, researchers, and assessment agencies, its large-scale application is still relatively rare. Continued research and development is critical to explore ways of how to design assessment tasks within technology-rich environments as well as to advance our knowledge of how to analyze and interpret student data generated from the assessment.

This study used a neuro-fuzzy approach, ANFIS, for the analysis of log data from a science simulation task. The training of ANFIS requires data with both inputs and outputs. In the context of technology-based assessment, the input variables are typically student behaviors/actions that are key to the effectiveness and efficiency of student problem solving, while the output variables can be indicators of student overall performance on the assessment, such as the speed and accuracy of task completion. It is important to correctly identify the input and output variables used in ANFIS so that the fuzzy rules could provide useful insights to student problem solving. However, this is not an easy task and might require an iterative process in which variables identified based on human knowledge need to be revised to better capture the variability presented in the observed data. In this study, the inputs of our ANFIS analysis are student variables extracted based on the task analysis and review of log data and the output is the correctness of student final response to the research question that guides the entire simulation. The performance of ANFIS was compared to three other commonly-used machine learning techniques, including logistic regression, artificial neural network and support vector machines. Our results indicated that ANFIS outperformed logistic regression and artificial neural network for both the training and validation data, but produced slightly inferior classification results when compared to support vector machines. However, the advantage of ANFIS, compared with other machine learning techniques such as support vector machines, is that it is highly interpretive as the set of rules are simple to understand and grounded in substantive knowledge. These rules shed lights on the process of how inferences are made regarding student performance in the context of technology-based assessments.

Regarding future research, our analysis could be strengthened at least in two ways. First, our ANFIS analysis does not take into account the sequence of student actions, which might provide additional insights above and beyond what individual actions could offer. For the NAEP science simulation task, for example, being able to construct a table or graph with relevant variables specified in the research question is the key to the successful problem solving. Based on our review of log file, some students failed to make the correct

tables/graphs in their initial attempts, but after conducting a series of experiments, they correctly created the tables/graphs and consequently figured out the relationship examined in the research question. In comparison, some other students were able to construct the correct tables/graphs in their initial attempts, but then they started to generate tables/graphs with irrelevant variables before they conducted enough experiments, and as a result, these students failed to correctly solve the problem. Although both groups of students had non-zero frequencies of the action "correct tables/graphs," depending on where the action occurred in the problem solving sequence, the effect of the same action on the outcome may be very different. The ANFIS analysis may be improved by inputting some important information related to the sequence of student actions. However, this is not an easy task considering the complexity and variety of student problem-solving strategies. Additional research is needed to examine how best to incorporate sequential information into fuzzy rules to improve the performance of ANFIS.

Second, our analysis generated and refined a set of fuzzy rules that may shed light on the process of how students solve the simulation task. Although we consider technology-based assessments to be most beneficial for formative purposes, summative score reporting might be desirable. For example, an overall summative profile score may prove useful for students wishing to know their final performance level or for teachers wishing to use it as part of the classroom assessment. It is worth exploring how the set of fuzzy rules based on the ANFIS analysis may be combined with other modeling techniques such as Bayesian Networks (Almond, Mislevy, Steinberg, Yan & Williamson, 2015) and cognitive diagnostic models (Rupp & Templin, 2008) to gauge student mastery of key processes and skills with the aim to generate summative profile scores.

REFERENCES

Alberta Education. (1996). *Science.* Alberta, Canada: Alberta Education. Retrieved from http://education.alberta.ca/media/654825/elemsci.pdf.

Alberta Education. (2014). *Science grades 7-8-9: Program of studies.* Alberta, Canada: Alberta Education. Retrieved from http://education.alberta.ca/media/654829/sci7to9.pdf

Almond, R. G., Mislevy, R. J., Steinberg, L. S., Yan, D., & Williamson, D. M. (2015). *Bayesian networks in educational assessment.* New York, NY: Springer.

Bennett, R., Persky, H., Weiss, A., & Jenkins, F. (2007). Problem solving in technology-rich environments: A report from the NAEP technology-based assessment project. *The Nation's Report Card* (p. 180). Retrieved from http://nces.ed.gov/nationsreportcard/pdf/studies/2007466.pdf.

Chu, M.-W. (2017, March). Using computer simulated science laboratories: A test of pre-laboratory activities with the learning error and formative feedback model. Unpublished doctoral dissertation, University of Alberta, Edmonton, Canada.

IBM Corp. (2016). IBM SPSS statistics for windows, version 24.0. Armonk, NY: IBM Corp.

Jang, J. S. R. (1993). ANFIS: Adaptive network based fuzzy inference systems. *IEEE Transactions on Systems, Man, and Cybernetics*, *23*(3), 665–685. doi:10.1109/21.256541

Masters, T. (1994). *Practical neural network recipes in C++*. San Diego, CA: Academic Press.

Mathworks. (2017). *Fuzzy logic toolbox: User's guide (r2017b)*. Retrieved from https://www.mathworks.com/help/pdf_doc/fuzzy/fuzzy.pdf

Mayrath, M. C., Clarke-Midura, J., & Robinson, D. (2012). *Technology-based assessments for 21st century skills: Theoretical and practical implications from modern research*. Charlotte, NC: Information Age.

National Center for Educational Statistics. (2012). *Science in action: Hands-on and interactive computer tasks from the 2009 science assessment* (NCES 2012–468). Washington, DC: U.S. Department of Education.

OECD. (2010). *PISA 2009 results: What students know and can do: Student performance in reading, mathematics and science (Volume I)*. PISA, OECD Publishing. Retrieved from http://dx.doi.org/10.1787/9789264091450-en.

Quellmalz, E. S., Timms, M., Buckley, B., Davenport, J., Loveland, M., & Silberglitt, M. (2012). 21st century dynamic assessment. In M. Mayrath, J. Clarke-Midura, & D. Robinson (Eds.), *Technology-based assessments for 21st century skills: Theoretical and practical implications from modern research* (pp. 55–90). Charlotte, NC: Information Age Publishers.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533–536. doi:10.1038/323533a0

Rumelhart, D. E., & McClelland, J. L. (Eds.). (1986). *Parallel distributed processing: Explorations in the microstructure of cognition: Vol. 1. Foundations*. Cambridge, MA: MTT Press.

Rupp, A., & Templin, J. (2008). Unique characteristics of diagnostic classification models: A comprehensive review of the current state-of-the-art. *Measurement: Interdisciplinary Research and Perspectives*, *6*(4), 219–262. doi:10.1080/15366360802490866

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, *8*(3), 338–353. doi:10.1016/S0019-9958(65)90241-X