Neural networks applied to Item Response Theory

BENITEZ ROCHEL, R., TRELLA LOPEZ, M., CONEJO MUÑOZ, R.
Departamento de Lenguajes y Ciencias de la Computación
Universidad de Málaga
Campus de Teatinos s/n, 29079 Málaga.
ESPAÑA
benitez@lcc.uma.es trella@apolo.lcc.uma.es conejo@lcc.uma.es

Abstract: **-** This paper reports on the application of artificial neural networks to Item Response Theory. In particular the competitive learning paradigm is used to build a neural network that determines a particular student's ability level given his answers to items with different levels of difficulty. The results are analysed and compared with other models within the framework of Item Response Theory.

Key Words: **-** Neural networks, competitive learning, Item Response Theory.

1.- Introduction

Most of the practical applications in the Theory of the Measurement in Psychology and Education are based in the *Classical Tests Theory*, deficiencies of which encourage the search of alternative models. The most relevant are the ones based in the *Item Response Theory* (IRT) [1], initially known as latent trait theory. The IRT, based on strong hypothesis, tries to give probabilistic foundations to the problem of non-observable traits measurement.

All the IRT based models have some common features: (1) they suppose the existence of latent traits or aptitudes (in our particular case the trait is the student knowledge level) that allow predicting or explaining examinee behaviour with a test item; (2) the relation between the trait and the response of a person to the item can be described with an increasing monotonous function called *Item Characteristic Curve* (ICC), that establishes the responses likelihood.

This paper describes the use of artificial neural networks in Item Response Theory. In particular competitive paradigm has been useful in providing an estimation of Item Characteristic Curve and classifying a student in a certain knowledge level. The current work emphasises how the competitive learning is a valuable aid in clustering the training patterns into representative groups. In the area of competitive learning a rather large number of models exist which have similar goals but differ considerably in the way they work [2]. Two different models have been considered in order to study if the performance of a supervised net is better than an unsupervised one in this case. A simulator program has been constructed to study the empirical characteristics of each one.

This paper is structured in the following way: First of all, the Item Response Theory and the classical methods of ICC parameter estimation is presented. After that, the neural models used in this paper are briefly presented. The IRT problem representation by using these neural models is presented in the next section. Finally, section 5 presents the empirical results obtained by using a simulation program, and analyses the self-learning capabilities. At the end, the advantages obtained with the use of neural models instead of the classical approach are summarised and some open issues in this line are proposed.

2.- IRT.

Item Response Theory (IRT), also known as latent trait theory, was originated in the late 1960s and had its roots in psychological measurement theory. As Lord and Novick explain [3], the term latent trait original referred to a hypothetical psychological construct that was suposed to underlie an individual's observed behaviour. In a testing context, according to Wainer and Messick [4], this definition translates into a latent trait, an attribute that accounts for the consistency of tests' responses. Such an underlying trait can be visualised as a continuum on which persons and test items can be placed according to their ability or level of difficulty. The position of students in this continuum, which may be represented by a numerical scale, can be estimated on the basis of their responses to suitable test items [5].

In the classical way, each question or item in a test is assigned an Item Characteristic Curve (ICC) which is a function that represents the probability of given a right answer to that question given a certain student's knowledge level *q Î (-¥,+¥)*, defined as a real number. Lets represents this by the expression: $P(U_i=1|\ q)$ or simply P_i. Logically, the probability of failing the question is $P(U_i=0|\ q) = 1-P(U_i=1|\ q)$, or simply Q_i. If the test is composed by n questions, knowing the ICCs, and supposing local independence of items, a likelihood function can be constructed:

$$L(u_1, u_2 \ldots u_n | q) = \prod_{i=1}^{n} P_i^{u_i} Q_i^{1-u_i}$$

The maximum of this function gives an estimation of the most likely value of q. A distribution of the probability of q can be obtained applying n times the Bayes' rule.

One of the main problems in IRT theory is to find out the ICCs. Several models has been proposed. The most popular are those that suppose that the ICCs belong to a family of functions that depends on one, two or three parameters. These functions are constructed based on the normal or the logistic distribution function. For instance, based on the logistic function the ICC might be described by:

$$P_i(q) = c_i + (1 - c_i)\frac{1}{1 + e^{-1.7 a_i (q - b_i)}}$$

where b_i is known as the *difficulty* of the question; c_i is the *guessing factor*, and a_i the *discrimination factor*.

The *guessing factor* is the probability of that a student with no knowledge at all solves the question The *difficulty* represents the knowledge level in which the student has equal probability to answer or fail the question, besides the guessing factor. The discrimination factor is proportional to the slope of the curve. If the discrimination factor is high then the probability students with level lower than b will probably fail and students with lever higher than b with more surely give the right answer.

Assuming that the ICC forms belong to this family, the problem is now formulated as the estimation of the parameter that fits better.

In order to estimate this parameters, it is known that one-parameter model (where a_i and c_i are supposed to be constants) is relatively simple to use and yields reliable estimates with as few as a hundred subjects. The two-parameter model (c_i constant) takes the possible differences in the items' discrimination into account and, though this results in higher accuracy, it also means that the increased complexity of the model will require a higher number of participants, two hundred at least. The three-parameter model even takes guessing into account, but the price to pay for this includes a sample of a minimum of one thousand subjects [6]

The most common situation is found when both person ability and item difficulty parameters are unknown. In this case it is needed a matrix formed with the responses given by N students to a set of n items.

There are a lot of methods of estimating such parameters but Join Likelihood Maximum and Marginal Likelihood Maximum are the methods commonly used.

2.1. Estimation by Joint Likelihood Maximum

Join Likelihood function, when it is considered N students responding to n questions, is:

$$L(u_1, u_2, \cdots, u_N \mid q, a, b, c) = \prod_{a=1}^{N} \prod_{i=1}^{n} P_{a_i}^{u_{a_i}} Q_{a_i}^{1-u_{a_i}}$$

where local independence is supposed and u_a is the response pattern of a student corresponding to the n items considered, q is a vector which components are N parameters of person ability (one for each student); $a, b,$ c are also vectors formed by item difficulty parameters. In the three-parameter model there are $3n$ item parameters, in the two-parameter model $2n$ and in the one-parameter model n parameters. So in the three-parameter model a total of $3n + N$ parameters will have to be estimated.

The values of the parameter are found by maximising the likelihood function, or its logarithm given by

$$\ln L = \sum_{a=1}^{N} \sum_{i=1}^{N} \left[u_{a_i} \ln P_{a_i} + (1 - u_{a_i}) \ln Q_{a_i} \right]$$

The item difficulty and person ability are not univocally determined by this method. In order to eliminate this indeterminate solution an arbitrary scale is chosen for q y b, usually the standard scale with mean 0 and standard deviation 1. Then initial values for the person ability parameter is selected, normally ln(nº success/nº fails); to each subject. Now the item parameters are estimated in the way described above. In a second phase item parameters are considered known and ability parameters are estimated. This procedure is repeated until the minimum differences in the estimators of the parameters between two successive stages are reached.

2.2. Estimation by Marginal Likelihood Maximum

In this method tested subjects represent a random sample and the ability parameters have a specific probability distribution. Excepting special cases, conditional independence of different item responses given by subjects with the same ability q is supposed so it is possible to calculate the joint probability of item patterns in responses corresponding to a subject with ability q. The probability of observing a pattern u of a person with unknown ability q randomly selected from a population with a continue density distribution $f(q)$ for q is the unconditional probability defined by

$$P(u) = \int_{-\infty}^{\infty} P(X \mid q) f(q) dq$$

This is called marginal probability of u, and it is important to note that it only depends on item parameters. In [6] can be found further details in estimating marginal probability in applications

Simulation studies have shown that Marginal Likelihood Maximum estimators and its typical errors are more consistent and reliable than Joint Likelihood Maximum, when they are applied to short sized samples.

3.- Competitive learning

An important feature of neural networks is the ability to learn from their environment and through learning to improve performance in some sense. In competitive learning, as the name implies, the output neurons of a neural network compete among themselves for being the one to be active (fired). It is this feature that makes competitive learning highly suited to discover those statistically salient features that may be used to classify a set of input patterns.

The goal of competitive learning is to cluster or categorise the training patterns into representative groups such that patterns within a cluster are more similar to each other patterns belonging to different clusters. Based on a learn only if it wins, that is, winner-take-all principle, neurons in a network based on competitive learning compete to move to the centroids of similar patterns and consequently uncorrelated patterns will be encoded by different neurons. Depending on the nature of applications, the available training patterns could be unlabelled or labelled, and hence unsupervised and supervised competitive algorithms were proposed accordingly. In the application considered in this paper two different competitive algorithms of which one is unsupervised and one is supervised are used. The unsupervised algorithm is the standard competitive algorithm. Among the supervised algorithms, it was chosen Kohonen's learning vector quantization (LVQ), with negative and positive reinforcement learning. The standard competitive learning is described in the next subsection and the Kohonen algorithm is described as a variation.

3.1. Standard Competitive Learning Algorithm

The competitive learning algorithm is usually associated with a layered feedforward network with fixed output nodes as shown in Figure 1.

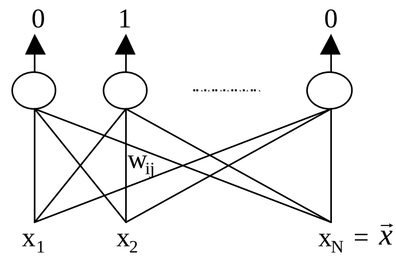

Figure 1

The algorithm can be defined with the next steps:

Step1 -. Initialisation:

- Set the number of competing neurons *c*.
- Initialise the neuron's weights vectors $\vec{w}_j(0) = (w_{1j}, w_{2j}, \cdots, w_{Nj})' : j = 1, \cdots, c$.

Step2 -. Distance computation:

- For input pattern $\vec{x}_k$, compute $\mathrm{h}_j = \vec{w}_j \vec{x}_k - q_j$ for all competing neurons $j = 1, \cdots, c$. It represents the influence that input units, weighted connections and the bias $q_j = \vec{w}'_j \vec{w}_j / 2$ have on neuron.

Step3 -. Competition:

- Determine the wining neuron r having $\mathrm{h}_r = \max_i \{h_i\}$ that neuron is nearest to the input pattern and produces a +1 output for it and zero outputs for the losing neurons.

Step4 -. Learning:

- Update the winning neuron's weight vector as

$$\Delta\vec{w}_r = \vec{w}_r(t+1) - \vec{w}_r(t) = a(t)(\vec{x}_j - \vec{w}_r)$$

where $\alpha(t)$ is the learning rate the is usually monotically decreasing.

Step5 -. Termination:

- Repeat steps 2-4 until the terminating criterion is met.

3.2. Learning Vector Quantization Algorithm

Unlike Standard Competitive Learning, Kohonen's learning vector quantization (LVQ) [7] is a supervised type competitive learning that works the misclasification rate. Competing neurons have to be labelled a prior to a specific class and the number of competing neurons should be equal to or greater than the number of classes considered. The LVQ algorithm may well described by the Standard Competitive Learning algorithm with the following modification in the learning rule:

$$\vec{w}_j(t+1) = \vec{w}_j(t) + \alpha(t) I_j(t) \left[\vec{x}_k - \vec{w}_j\right]$$

where

$$I_j(t) = \begin{cases} +1 & if \quad Cl(\vec{w}_j) = Cl(\vec{x}_k) \\ -1 & if \quad Cl(\vec{w}_j) \neq Cl(\vec{x}_k) \end{cases}$$

and $Cl(\cdot)$ denotes the *class of* operator. Thus, LVQ adopts a reinforce-or-punish learning principle in the competitive process so as to move the winning neuron's weight vector $\vec{w}_j$ closer to the class centroid if the current training pattern $\vec{x}_k$ is correctly clasified and to move $\vec{w}_j$ out of the misclassified region if $\vec{x}_k$ is wrongly clasified.

4.- IRT neural representation

When a practical neural network application is considered simply knowing how neural network models behave is quite only half of the problem in creating such application. The other aspect of building successful neural-network applications is the process of acquiring and modelling the application data, selecting the most appropriate network model for the application.

Our study of this neural network application design in IRT will begin by first describing, in detail, data representation, network architecture selection and training options. We conclude by illustrating the performance of the network model.

One of the first tasks to be done to apply neural computation to IRT is to formulate the theory in the domain of full and finite numbers, instead of real numbers. The knowledge level of a student is no longer a real number between $-\infty$, and $+\infty$, but a full number in the range *[0,N]*. This approximation is accurate enough for most real applications. This simplification makes sense because to evaluate a student it is normally assume by teachers that the final qualifications are tags like A+, A, B+, etc.

Therefor, the ICCs are now viewed as (N+1)-component vectors, whose components are the probabilities that a student of each knowledge level could give the right answer to the question.

Knowing the ICCs a competitive neural network can be constructed in the following way:

First of all the problem is defined precisely; that is, mathematically. It is a considered like a problem of clustering because the responses given by N students to a set of n items are given in order to determine the knowledge level of a student that is a full number in the range *[0,10]*.

Given such identification of the problem the training input set of N examples with n features can be represented by a set of N vectors where $x_{kj} \in \{0,1\}$, $j = 1,\cdots n$. When the j component of vector k is $x_{kj} = 0$ means that the student k has failed the question j, and $x_{kj} = 1$ indicates that this student gave a correct answer to the question.

The goal is to construct a neural network that forms its own classification of the data from the training examples into predetermined number of clusters. Each group represents the knowledge level associated to a student. A network paradigm that performs that task required is competitive learning. So a competitive layer with 11 neurons and n input units was chosen as network architecture (see figure 2).

In the net described above there is an associated set of weights $\vec{w}_i$ for each cluster $i = 0,1,\ldots,10$. These weights of the net are the components of the ICCs, and taking into account that the competitive learning implements the bayesian classification [8], the winner neurone will give the best estimation of the student's knowledge level q_i , $i = 0,1\ldots,10$.

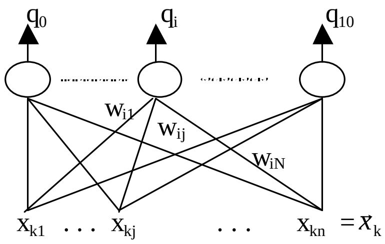

Figure 2

In a neural network with the architecture described above, two learning algorithms were employed to train the weights in the clusters so an estimation of ICCs are calculated. First, the standard competitive learning as unsupervised algorithm was considered and then Kohonen's learning vector quantization, a supervised competitive mechanism was applied. In the next subsection a comparative analysis of both nets is shown.

One of the advantages of this approach is that it is not necessary to consider any particular form or family of curves for the ICC. For each ICC, N+1 parameters are directly inferred from the learning mechanism

On the other hand, when a reasonable good estimation is done, the results of the evaluation of a student are quite accurate, and could be used as a sort of unsupervised learning mechanism. This mechanism improves the performance of the network. We do not have a theoretical prove of this result, but empirical results using a simulator of students and questions.

5.- Empirical results evaluation

In order to obtain empirical results about the application of neuronal networks with competitive learning to the Item Response Theory, a tests simulator has been developed. The tests simulator is a program that has five main processes: (a) Generates a set of questions with their corresponding ICCs to be used in a test. (b)-Simulates the answer of a set of students while making that test (c) Classifies the simulated students according to their knowledge level using a neural network with the structure described in the previous section (d) Applies the learning mechanism to the neural model to improve the weights that represents the ICC of each question, and (e) Presents the results and some statistical information..

There are several factors that can be configured in the simulation of a test, both in the ICCs of the questions and in their global features. The main are: *(1) Evaluation method*: that can be *modal*, *percentage* or *real*; (2) *Learning method*: that can be data-adaptive, or *block-adaptive*. (3) *Student knowledge distribution*: that can be *homogeneous* or *binomial*. (4) *Number* of students simulated (=number of tests to simulate); (5) *Number of classes* or *knowledge levels*, that equals the number of the output layer. (6) *Number of questions per test*; or *length of the questions database, that equals the number of cells in the input layer; a*nd (7) *Initial questions set*; that can be *correct*, *equilibrated* or *undefined*. Along this main parameters it is possible to define the ICCs of the questions, by giving the classical parameters: *(a) difficulty (real or estimated), (b) discrimination factor* (real o estimated); and (c) *guessing factor* (real or estimated); and

5.1.-Types of questions sets

The first process to execute a test simulation is to generate a questions set. Each questions has associated three ICC that are used during the simulation:

- *Actual Item Characteristic Curve*, that represents the actual distribution of the question ICC and it is supposed to be unknown by the system. This curve is used to simulate the answer of the students to the question.
- *Estimated Item Characteristic Curve*, which is the estimated distribution used to evaluate the students during the simulation, and that are related to the weights assigned to the connections of the net.
- *Learned Item Characteristic Curve, which is* distribution that is constructed during the simulation, and are related to the new weights calculated after the learning mechanism. Learning is done by replacing the estimated curves with these ones, either data-adaptive, or block-adaptive.

As it has been mentioned above, the system can work with different initial sets of questions that have particular conditions. The types of sets are:

- *Correct*: is a set of questions that are well calibrated by the instructor or test designer. The estimated ICCs are equals to the actual ICCs. There are two subtypes:
  - *Homogeneous:* there is the same number of questions for each difficulty level.
  - *Random*: the number of generated questions for every level is not the same.
- *Equilibrated:* This set represents the case in which an instructor calibrates empirically a set of questions. Not all the questions with real difficulty equal to *k* have assigned this value, but the mean value of his/her estimation is correct. more precisely, if dr_i is the actual difficulty for the question *i* and de_i is its estimated difficulty given by the teacher. A set of question is equilibrated if:
  - $\forall k \quad \forall i \quad dr_i = k, \quad mean(de_i) = k$,
- That is, it is mandatory that the mean of the difficulty parameters estimated a priori will be the same than the actual difficulty of all of them. There are three kinds of equilibrated sets:

- *Homogeneous*: if the real difficulty level is *dr*, there is the same number of questions badly classified of estimated level dr+*n* than *dr*-*n* ($n \in [0, min(9-dr, dr-1)]$).
- *Random:* the number of questions badly classified is not the same than the witch ones in the levels *dr*+*n* and *dr*-*n*. It is obtained with a binomial random distribution whose mean is centred in the set real difficulty.
- *Extreme*: it is an equilibrated set with a maximum variance of the difference of estimated and actual difficulties.

The initial weights of the network connections are calculated from the values of the initial estimated ICCs which are constructed taking discrete values of the normal distribution function.

5.2. -Learning

There are two types of learning implemented depending on the frequency of update

- *Data-adaptive*: the weights are updated at each iteration, one update per datum.
- *Block*-adaptive: the update is executed upon the completion of each sweep, that is after a certain number of students has passed the test.

The network learning mechanism is applied directly. In the competitive standard network the weights w_{ij} of the connections represents exactly the j values that define the learned ICCs of the questions i

In the simulator there are two options for the learning algorithms:

- *Actual evaluation*: the final classification of the network is discarded and the actual value of student knowledge is taken as the winner neuron.
- *Modal evaluation*: it is obtained by considering the network classification itself,

The most interesting question is to know if the network can improve the quality of the set of questions' initial estimated ICC. That is, if the estimated ICCs of the questions will be closer to the actual ICCs, after applying the learning mechanism and if this task can be done with the unsupervised mechanism. If the questions set quality is improved it will consequently improve the number of student well classified.

5.3.-Test simulation.

The simulation process starts after the system has generate the set of questions that will be used. The system simulates as many students as number of tests has been indicated in the input parameters. Each student is represented by his/her real knowledge level (it is assigned a random number between the knowledge level classes bounds) and his/her estimated knowledge that is computed by the net.

Each generated student will make a test. The answers of a student or input vector to the network is composed by the answers to each question in the set. The answer to each question is generated randomly with a certain probability according to the actual knowledge of the student and the actual ICC of each question. It is represented by 0 or 1 in the corresponding input cell.

The trained neural model then will be used later in the retrieving phase to process the pattern corresponding to this student and yield classification.

5.4. - Empirical results

In order to observe the results of applying neuronal networks to the IRT a set of simulations have been realised. Several tests have been systematically simulated changing the initial question set (equilibrated random or homogeneous), and the competitive learning algorithm (standard or LVQ).

Diverse measurements to compare the results have been used. The mains are the mean of the distance between the actual and learned ICCs and the percentage of students well classified with and without learning.

The distance is a statistical estimator that indicates how different is the actual ICC from the estimated one. Similar estimators have been used in psychometric to define the parameter estimation degree. The distance for each question is:

$$\sum_{k=0}^{k=N} [ActualICC(k) - EstimatedICC(k)]^2$$

to compare two sets of questions the mean of the distances is used.

The simulations have been executed with 1000 students (=number of tests). The results are showed in the tables 1 and 2.

The firsts proofs have been done to confirm that if the initial questions set is correct then the percentage of students well classified (with *modal evaluation*) is quite near to the total number of students (96%-97%). That indicates that the net constructed with its initial weights as the questions ICC works correctly.

The next experiments are oriented to observe what happens if the initial questions set is not correct. Improving the weights of the nets will be equivalent to calibrate the ICC of the questions (firstly defined by the test designer or the instructor).

The questions initial set used it is *equilibrated homogeneous* and the mean of its ICCs distances is 0,0278.

If the *actual evaluation* is applied then we obtain a reduction in the ICCs distances regarding to the ICCs initial distances. That is, the net improves its weights and consequently the ICCs become better.

If *modal evaluation* is applied the improvement of the ICCs it is worse than with the *actual*.

In general the results are better with LVQ algorithm, it confirms that a supervised algorithm suites to IRT. A substantial difference respect to the percentage of students classified correctly exists when data-adaptive learning is considered, with LVQ is superior. That is because Standard Competitive Algorithm has some disadvantages like the dependence of data presentation order.

		Block adaptive	Data adaptive
Modal	Final Means of ICCs distance	0,0049	0,0744
	Students classified correctly	77,7%	20,1%
Actual	Final Mean of ICCs distance	0,0021	0,0021
	Students classified correctly	100%	100%

Table 1 Simulation results applying Standard Algorithm

		Block adaptive	Data adaptive
Modal	Final Means of ICCs distance	0,0119	0,0118
	Students classified correctly	78,6%	86,8%
Actual	Final Mean of ICCs distance	0,0099	0,0094
	Students classified correctly	100%	100%

Table 2 Simulation results applying LVQ algorithm.

6. - Conclusions

A new representation of the IRT procedure has been presented in this paper. One of the main differences between the classical and the neural approach is that the first one uses continuous functions defined in the real number domain to represents the ICC and the student estimated knowledge, and the latest uses discrete values between 0 and N.

The discrete approach is easier to implement by using a neural competitive net and its computational cost is less than solving the continuous equations.

One of the main advantages of using neural nets is that the learning mechanism of the network can be applied to improve the initial estimation of the parameters that define the questions ICC. The classical methods for parameter estimations are very complex and usually they are applied only once for an initial calibration set of students. The incremental mechanism of the neural learning can be implemented during all the life of the test, and so the test will gain information and improves its performance each time it is applied.

There are many open issues that still remains: We do not have a theoretical prove of the convergence of the ICCs parameters for the equilibrated set, but it makes sense and is easily proved if the equilibrated set is symmetrical, (which is a very restrictive and unreal condition). Nor there is no prove for other times of initial sets that reflects common error of the teacher while creating a IRT test. It is also desirable to use real world data for questions ICCs and students answers, instead of simulated values.

It is also planned to construct other neural models and study its performance compared to the competitive model.

REFERENCES

[1] Hamblenton, R. K. (1989) Principles and selected applications of item response theory. In R.L. Linn (Ed.). *Educational Measurement.* (3ª ed.). New York, Macmillan.

[2] Haykin, S. (1995) *Neural Networks: A comprehensive foundation* pp 352-473, IEEE Press

[3] Lord, F.M. & Novick, M.R. (1968). *Statistical theories of mental test scores* p. 359. Reading, MA: Addison-Wesley.

[4] Wainer, H. & Messick, S. (1983). *Principles of modern psychological measurement* p. 343. Hillsdale, NJ: Lawrence Erlbaum Associates.

[5] Hulin, C. L., Drasgow, F. & Parsons, C. K. (1983). *Item response theory* p. 15. Homewood, IL: Dow-jones Irwin.

[6] Alderson, J. C., Claphan, C., & Wall, D. (1995). *Language test construction and evaluation. Cambridge: Cambridge University Press.*

[7] Kohonen, T. .(1984) *Self-Organization and Associate Memory, Series in Information Science*, volume 8. Springer-Verlag, New York, (pp. 73, 7, 85, 116, 117, 122)

[8] Funahashi, K. (1998) Multilayer Neural *Networks and Bayes Decision Theory.* Neural Networks Vol11 pp 209-213.