Novel Use of Self-organizing Map for Q-matrix Calibration in Cognitive Diagnosis Assessment

Xi-tian Chen
Department of Psychology
Sun Yat-sen University
Guangzhou, China
chenxt47@mail2.sysu.edu.com

Zhengjia Dai
Department of Psychology
Sun Yat-sen University
Guangzhou, China
daizhengjia13@gmail.com

Ying Lin*
Department of Psychology
Sun Yat-sen University
Guangzhou, China
linying23@mail.sysu.edu.cn

***Abstract*—Cognitive diagnostic assessment (CDA) is a newly emerged tool for educational and psychological measurement. Unlike traditional assessment, CDA can measure a participant's detailed mastering pattern of finer-grained skills and knowledge (a.k.a. *attributes*). An important component in CDA is the Q-matrix, which associates each item with particular attribute(s) and thus forms the basis for performing diagnosis. However, in real-world applications, the Q-matrix is often defined by hand, which induces high risks of subjective errors. The purpose of the current investigation is to describe how the self-organizing map (SOM), a typical kind of unsupervised artificial neural network, can be used to validate and calibrate the Q-matrix. To do so, we propose to convert the Q-matrix calibration problem into a clustering-labeling process: first cluster the participants' responses to identify groups of test items and then label the item(s) with unknown or obscure attribute(s) based on group similarity. Following this idea, a one-phase method and a two-phase method are proposed. The one-phase method employs a single SOM to implement Q-matrix calibration at a fast speed. The two-phase method, which adopts a double-SOM design, is a bit more time-consuming but can provide a diagnosis of participants as a by-product. Experimental results on simulated data generated under various conditions confirm the effectiveness of the proposed method. As a real-data illustration, the proposed methods are also applied to analyze large-scale educational reading assessment data and achieve promising results.**

Keywords*—Cognitive diagnosis, polytomous response, Q-matrix, self-organizing map

I. INTRODUCTION

To evaluate the progress of students and monitor the quality of educational systems, there are lots of large-scale educational assessments, such as the Program for International Student Assessment (PISA) and Trends in International Mathematics and Science Study (TIMSS). In the past decade, there has been a surge of interests in cognitive diagnostic assessment (CDA). Unlike traditional assessments that can only provide a total score as the overall ability, CDA can provide a mastering pattern of different finer-grained skills and knowledge. In the CDA context, skills and knowledge are referred to as *attributes*. The goal of CDA is to estimate the mastery of such attributes of students based on the observed item responses.

Cognitive diagnosis is a complicated work that includes "Q-matrix theory" and "diagnosis classification" [1], corresponding to two important components in the CDA, Q-matrix and cognitive diagnosis model (CDM), respectively. CDM, despite of the rich variants in the psychometric literature, is essentially a latten variable model for evaluating student ability. Q-matrix indicates the set of skills that are required to answer a particular item correctly [2]. Notably, so far Q-matrix is necessary under all CDM assumptions for the application of CDA. However, the definition of Q-matrix is typically constructed by domain experts, which results in subjective errors and misspecification and can negatively affect the estimation results [3][4]. To reduce these errors, the Q-matrix validation becomes an important issue in the practical application of CDA [5][6]. For decades, researchers proposed a number of test-level Q-matrix validation procedures that relied on the specific CDM and the statistical indexes [4][7][8]. Nevertheless, these methods are not only computational expensive but also susceptible by the quality of preset Q-matrix.

Recently, the artificial neural network (ANN) approach has been widely applied in CDA. Gierl *et al.* [9][10] employed ANN to estimate students' attribute patterns measured on tests under the framework of the attribute hierarchy method. ANN was also employed under the *deterministic inputs, noisy "AND" gate* (DINA) model [11] to provide the prior estimates of students' attribute patterns [12]. ANN was also adopted in a range of applications in CDA, including word recognition problems [13] and computer-based performance assessments [14].

Though many studies have used the ANN approach to assist CDA, few have focused on the Q-matrix validation. Recently, the supervised ANN was used to improve the diagnostic accuracy of student attribute patterns, with the process of the Q-matrix estimation included [15]. However, the training of supervised ANN depends on the label of each input vector (i.e. the known q-vector of items in the Q-matrix validation problem). Compared with supervised ANN, the training of unsupervised ANN doesn't require the label of input vectors, which means the application of unsupervised ANN is less susceptible by the misspecified q-vectors. Besides, researchers have also found that unsupervised ANN tends to perform better in student classification than using supervised ANN in CDA [16]. Thus, our work tries to use unsupervised ANN to validate and calibrate Q-matrix. It is a novel ANN application in a new issue in the domain of educational and psychological measurement. Compared with traditional Q-matrix validation

* This work was supported by the National Natural Science Foundation of China under Grant 61772569 and the Fundamental Research Funds for the Central Universities (No. 20wkzd11). Ying Lin is the corresponding authors (linying23@mail.sysu.edu.cn).

methods based on statistics, the results of this data-driven approach should be more objective and accurate.

II. Background

A. Brief Introduction to CDA

Cognitive diagnostic assessment (CDA) is devised to establish the relationship between student response data and the mastery of different finer-grained skills and knowledge. Every student has a vector as the attribute pattern to represent her/his mastery and non-mastery. The CDA assumes the students with the same attribute pattern have the same probability of answering an item correctly.

The binary $J \times K$ Q-matrix indicates whether test item j, $j = 1, \ldots, J$, requires that a student master attribute k, $k = 1, \ldots, K$, to answer this item correctly. Each "q-entry", q_{jk}, of the Q-matrix is coded 0 (not required) or 1 (required). Each row of the Q-matrix, which called "q-vector", $\boldsymbol{q}_j$, describes the required attributes for a particular item j. Every student has a K-element vector as his or her attribute pattern. Each element of this vector is coded 0 (not mastered) or 1 (mastered). For a test measuring K attributes, there are 2^K possible attribute patterns and 2^K possible q-vector patterns.

In the psychometric literature, Researchers proposed a lot of CDMs to evaluate students and items. Some CDMs have strong assumptions about the processes involved in problem-solving, for example, the DINA model and the *deterministic inputs, noisy "OR" gate* (DINO) model [17]. Some CDMs with weak assumptions are general, for example, the *generalized DINA* (GDINA) model and the *general diagnostic model* (GDM) [18][19]. Traditionally, CDMs are designed for dichotomous attributes (either mastered or not) and dichotomous responses (either correct or wrong). Lately, some CDMs accommodate multiple strategies and cognitively multiple-choice items [20][21]. Besides, for polytomous responses, researchers proposed the sequential GDINA model (sGDINA) and the general polytomous diagnosis modeling framework (GPDM) [22][23].

GDINA assumes that for a specific item, students with different attribute patterns have different probabilities of answering correctly (Fig. 1a) [18]. The probability that students with attribute pattern $\boldsymbol{\alpha}^*$ answer item j correctly can be calculated by the item response function (IRF),

$$P\left(\boldsymbol{\alpha}_{lj}^*\right) = \delta_{j0} + \sum_{k=1}^{K_j^*} \delta_{jk}\alpha_{lk} + \sum_{\Lambda\subseteq \mathrm{A}} \delta_j(\Lambda)\prod_{k\in\Lambda}\alpha_{lk} \quad (1)$$

where δ_{j0} is the intercept, δ_{jk} is the main effect due to attribute k, and $\delta_j(\Lambda)$ is the interaction effect due to the attributes in the subset Λ of A, where A is the universal set of attributes. The guessing parameter $g_j = \delta_{j0}$ is the probability that individuals who lack every required attribute for item j may answer it correctly. The slipping parameter $s_j = (1-\sum\delta_j)$ is the probability that individuals who possess all the required attributes may answer the item incorrectly [24]. The guessing and slipping parameters represent the quality of items and existed because of some complex reasons, such as inappropriate Q-matrix specification and the use of alternative strategies.

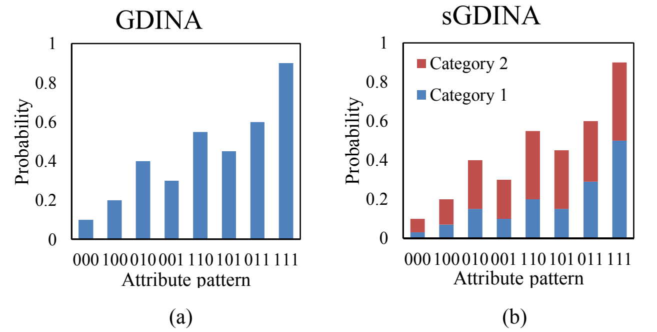

Fig. 1. An example item of GDINA(a) and sGDINA(b). The probability in (b) is the conditional probability of each category.

Although most of CDMs are designed for dichotomous responses, some CDMs for polytomous responses are proposed. sGDINA is designed for graded responses with a sequential item category based on the GDINA framework (Fig. 1b) [22]. Dichotomous responses and polytomous responses in sGDINA are processed on item-level and category-level, respectively. In sGDINA, students answer category c correctly after completed all previous steps successfully. To associate attributes and categories, two kinds of category-level Q-matrix (Q_C-matrix) were used instead of traditional Q-matrix. Similarly, the function of attribute patterns and the required attributes for category c of item j, can be formulated as

$$S_j\left(c\middle|\boldsymbol{\alpha}_{lj}^*\right) = \delta_{jc0} + \sum_{k=1}^{K_{jc}^*} \delta_{jck}\alpha_{lk} + \sum_{\Lambda\subseteq \mathrm{A}} \delta_{jc}(\Lambda)\prod_{k\in\Lambda}\alpha_{lk} \quad (2)$$

where the parameters (i.e. intercept, main effect, interaction effect, guessing, and slipping) are on category-level. Therefore, the conditional probability that students with attribute pattern $\boldsymbol{\alpha}^*$ score c on item j can be calculated by the category response function (CRF),

$$P\left(X_j = c\middle|\boldsymbol{\alpha}_{lj}^*\right) = \left[1 - S_j\left(c+1\middle|\boldsymbol{\alpha}_{lj}^*\right)\right]\prod_{x=0}^{c} S_j\left(x\middle|\boldsymbol{\alpha}_{lj}^*\right) \quad (3)$$

B. Q-matrix Validation Problem and Existing Methods

As the bridge connecting items and attributes, every element in the Q-matrix must be explicit. Misspecification of the Q-matrix can negatively affect the estimation of the model parameters, result in poor model-data fit and incorrect classification of students [3][4][6]. Unfortunately, it is difficult to construct a Q-matrix correctly and the fallible judgment of experts in the domain of the test will lead to a misspecified Q-matrix. Therefore, solving the Q-matrix validation problem may help CDA applied widely.

In the beginning, Q-matrix is validated by experts evaluating. Some of the experienced experts in the domain determine the Q-matrix together by discussing or sending out corresponding questionnaires. According to this method, researchers designed and validated the Q-matrix of the reading test of the Michigan English Language Assessment Battery [25]. However, different experts in the domain have different knowledge, experience, and standards. In summary, the greatest advantage of experts evaluating methods is the interpretability of their results. On the contrary, Q-matrix validation by experts checking is too subjective and not reliable enough.

Later, to reduce the subjective errors of experts evaluating methods, researchers try to validate Q-matrix from the response of students inspired by iterative procedures based on item parameters. The Q-matrix is validated from a provisional Q-matrix created by experts during the phase of test development [4][7][26]. However, most of the existing Q-matrix validation methods rely on the specific CDM and statistical index. For example, a method is based on an item discrimination index, φ_j [3] which maximizes the difference between the probabilities that a student who masters all attributes can answers a particular item correctly or not. This method assumes that each item in the test conform to the DINA model. A simulation illustrates that this method can identify and replace the misspecified q-vectors and retain the correct q-vectors in the Q-matrix. Another method also assumes that items conform to the DINA model, and requires to predefine the potential misspecified entries in the Q-matrix [26].

Besides, the current Q-matrix validation methods are mostly restricted to dichotomous CDMs. Only few can be extended to the polytomous context. An empirical stepwise method based on the Wald test and an effect size measure is proposed [7]. As the number of CDMs for polytomous responses increases, the need for Q-matrix validation methods in the polytomous context becomes more and more urgent. In sGDINA, two kinds of Q_C-matrix (i.e. restricted Q_C-matrix and unrestricted) were proposed to adapt to different situations. Restricted Q_C-matrix needs prior knowledge about the association of attribute and category (TABLE I.). Different rows of restricted Q_C-matrix which indicated the q-vectors of category might be different. However, the prior knowledge may not be clear, especially when the CDA is retrofitted to existing assessments. Therefore, unrestricted Q_C-matrix is designed with the assumption that every category of the item needs all required attributes of this item (TABLE II.). Each row of unrestricted Q_C-matrix is the same as the item-level q-vector. Besides, the Q-matrix that used in GPDM is similar to the unrestricted Q_C-matrix. For this situation, the unrestricted Q_C-matrix needs to be validated into a restricted Q_C-matrix.

According to the above deficiencies, we try to develop a non-parametric, model-independent, flexible but computationally efficient Q-matrix validation method based on self-organizing map (SOM). Besides, our method can be used in the polytomous response.

TABLE I. AN EXAMPLE OF RESTRICTED Q_C-MATRIX FOR A FRACTION SUBTRACTION ITEM

Step	Category	Attribute[a]		
		α_1	α_2	α_3
$3\frac{9}{8}-\frac{3}{8}$	1	1	0	0
$3\frac{6}{8}$	2	0	1	0
$3\frac{3}{4}$	3	0	0	1

TABLE II. AN EXAMPLE OF UNRESTRICTED Q_C-MATRIX FOR A FRACTION SUBTRACTION ITEM

Step	Category	Attribute[a]		
		α_1	α_2	α_3
$3\frac{9}{8}-\frac{3}{8}$	1	1	1	1
$3\frac{6}{8}$	2	1	1	1
$3\frac{3}{4}$	3	1	1	1

[a] α_1, borrow from whole number part; α_2, subtract numerators; α_3, reduce answers to the simplest form

C. *Overview of SOM and Its Applications in CDA*

The self-organizing map (SOM) proposed by Kohonen is a popular ANN that generates a low-dimensional and discrete map according to data in the high-dimensional input space while preserving the original topological relationships of input data [27][28][29]. A typical SOM network is composed of an input layer and an output layer (Fig. 2). Input nodes are independent of each other and an input node represents an input feature. The number of input nodes corresponds to the dimensionality of the input vector. The output layer is typically a one- or two-dimensional map. The common two-dimensional map has two plane structures, rectangular (Fig. 2a) and hexagonal (Fig. 2b). Each input vector is classified into the closest output node. Each output node represents a distinct class associated with a kind of input vector, or the output nodes can be further grouped into a small number of classes. Each output node has a weight vector, as the geometric mean or the centroid of this group.

Instead of being trained by the error between the predicted and the ideal output, SOM is trained through competition among input nodes. Such a competitive training strategy enables SOM to realize unsupervised learning and makes it a promising and popular tool for exploratory data analysis, for example, applied in bilingual object naming [30].

Several SOM algorithms are proposed [31][32][33]. In our study, the batch-based training algorithm is employed for its fast speed and high stability [29]. The batch SOM contains the following four steps:

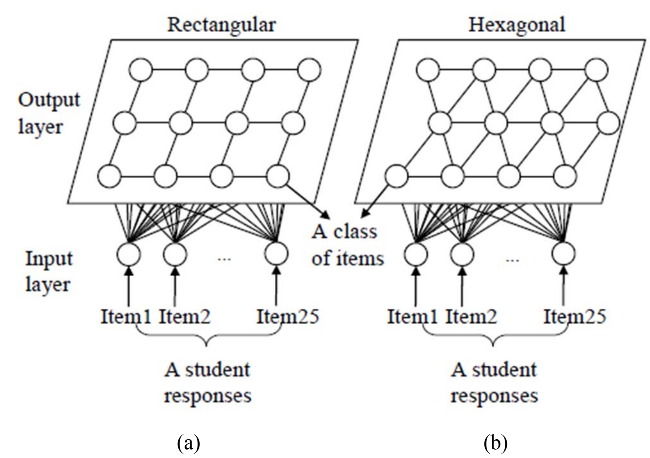

Fig. 2. An example of a self-organizing map that has an input layer with 25 input nodes and a two-dimensional output layer with 3 × 4 = 12 output nodes.

Step 1: Initialize the weight vector of each output node. The weight vector is of the same dimension as the input layer. For example, the SOM in Fig. 2 contains 25 input nodes. The weight vector of every output node is a vector of 25 real numbers.

Step 2: For each instance in the training set, compose an input vector using its features and calculate the Euclidean distance between the input vector and each output node. Choose the output node with the minimum distance as the best matching unit (BMU) for this input vector.

Step 3: For an output node *i*, let *j* denotes an output node in the neighbor of node *i*. For each node *j*, calculate the mean of the input vectors that choose node *j* as the BMU, denoted as $\bar{\mathbf{x}}_j$. Modify the weight vector of node *i* according to the mean input vector of neighbor nodes, formulated as

$$\mathbf{w}_i = \frac{\sum_j n_j h_{ji} \overline{\mathbf{X}}_j}{\sum_j n_j h_{ji}} \tag{4}$$

where n_j is the number of input vectors that chose node *j* as the BMU, and h_{ji} is the Euclidean distance between node *i* and *j*.

Step 4: Repeat steps 2 and 3 until the maximum number of iterations is reached.

Trained by the above algorithm, SOM can map similar instances in the high-dimensional space to the same or neighboring output nodes. Each output node can correspond to a cluster or the neighboring output nodes can be further classified into a larger cluster. The clustering procedure is thus realized. To use the SOM, we need to determine the network structure according to the complexity of the data to be clustered, including the neighborhood function, the maximum number of iterations, the size and the topology of the output layer. Generally speaking, the number of the output nodes must be larger than five times of square root of the number of input vectors. We choose bubble neighborhood function and the rectangular structure based on the standards that previous researchers used [16].

Chui *et al.* trained a SOM to classify students, in which the input vector is an observed response vector of student *i*, and the output node would be associated with a master attribute pattern [16]. The number of input nodes is fixed to the number of items in the test. The size of the output layer is around four to ten times the number of attribute patterns. Each class is associated with a group of students with a specific attribute pattern. When the number of output nodes is less than the total number of possible classes, some classes are not represented in the SOM. Therefore, the number of output nodes should be large enough to label one or several output nodes as a single class.

For interpreted classification results properly, it is important to label each output node accurately. That is, each output node must be associated with a specific attribute pattern. They used the minimum Euclidean distance between the weight vector of the output node and a vector of ideal response patterns (IRP) that corresponds to the attribute pattern. IRP is the expected response matrix of specific attribute patterns on a specific Q-matrix without guessing and slipping. If the students who are grouped into these output nodes have the same specific attribute pattern, they expected the corresponding weight vector should be proximal to the respective vector of IRP (i.e. the minimum Euclidean distance).

It shows that SOM is a non-parametric and model-independent tool in the CDA context. After figured out how to associate an output node with a q-vector pattern, SOM can be used in the Q-matrix validation problem.

III. The Proposed Methods

We proposed a new method to associate an output node with a q-vector pattern called the one-phase method. We used SOM based on the two-phase method proposed by Wang *et al.* [15]. Besides, we described the dichotomizing and cumulating process to apply these methods on polytomous responses.

A. One-phase Method

For the Q-matrix validation method by SOM, the key question is how to associate an output node with a q-vector pattern. Compared with the complicated procedure based on the minimum Euclidean distance and IRP, we proposed a simple one-phase method to generate the label of output nodes. In the one-phase method, we only need to train a SOM. This method has three steps as following:

Step 1: Train the SOM using the batch-based algorithm to classify items. The input vector is the observed responses of all students on an item and the number of input nodes is equal to the total number of students, *N*. Therefore, an output node will be a class of similar items. Because every output node should correspond to a possible q-vector, the number of output nodes must be no less than 2^K.

Step 2: Generate the label of the output node based on the items in this node. For each output node, the q-vectors of items in this node consist of a sub-Q-matrix (Q’). Calculate the mean of each column of Q’-matrix, $\bar{\boldsymbol{q}}$, stripping missing values. And then dichotomize the mean of each column as the label of this output node, $\boldsymbol{q}_e$, based on the preset cutoff value. Fig. 3 shows an example of step 2.

Step 3: After generating all labels of output nodes, for each output node, if there is missing value in the label of this node, fill in the corresponding known element of the nearest node. For example, consider the label of output node *i* is (1 1 ?). Calculate the Euclidean distance between node *i* and the other nodes. If the label of nearest node j_1 is (1 1 ?) and the label of second nearest node j_2 is (1 1 1), the missing value of the label of node *i* is equal to 1, the third element of the label of node j_2.

Chui *et al.* claimed that the number of output nodes should be set at least more than the number of attribute patterns, four to ten times better [16]. Based on their experiments, bubble neighborhood function tended to provide more accurate classifications and therefore was chosen for the SOM analysis.

$$\mathbf{Q'} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & ? & 0 \\ 1 & 1 & ? \\ 0 & 1 & 0 \end{pmatrix}$$

$$\bar{\mathbf{q}} = (0.75 \quad 1 \quad 0)$$

$$\text{under cutoff value} = 0.5$$

$$\mathbf{q}_e = (1 \quad 1 \quad 0)$$

Fig. 3. An example of how to use the known elements in the same node to label. Consider Q' was the sub-Q-matrix of items in this node. $\bar{\boldsymbol{q}}$ is the mean of every column in Q' deleted the missing value. Based on the preset cutoff value 0.5, q-vector of every item in this node was (1 1 0).

B. Two-phase Method

Moreover, Wang *et al.* applied probabilistic neural network (PNN) to calculate the unknown q-vectors of the Q-matrix, after estimating the student attribute patterns [15]. This method validates the unknown elements of the Q-matrix based on the IRP. We used SOM instead of PNN in this method following these three steps:

Step 1: Split out the R-matrix which consists of the known q-vector from the Q-matrix. Split the sub-responses matrix U_R (known items in the test) from U.

Step 2: Train a SOM to estimate the student attribute patterns. Simulate the IRP of all possible attribute patterns on the R. Use the row vectors of U_R as input. Label the attribute pattern of each output node (the student attribute pattern) based on the minimum Euclidean distance between the weight vector and the row vectors of IRP.

Step 3: Train a SOM to estimate the q-vectors. Simulate the IRP' of attribute patterns of all students on all the possible q-vectors. Use the column vectors of U as input. Label the q-vector of each output node (the item q-vector) based on the minimum Euclidean distance between the weight vector and the column vectors of IRP'.

C. Dichotomizing and cumulating process for polytomous responses

One-phase method and two-phase method can be applied to validate Q-matrix on dichotomous responses directly. Although these methods can be applied to polytomous responses directly, we dichotomized the polytomous responses because the q-vectors of categories in an item may be different (e.g. restricted Q_C-matrix in sGDINA). Besides, we also cumulated the dichotomous sub-responses of the polytomous responses and Q_C-matrix, because graded responses represent that students can answer high category correctly after completed all previous categories successfully.

According to the feature of graded responses, a *c*-category polytomous item can be split into *c*-1 dichotomous sub-items for each non-zero category. For item *j* with $C_j + 1$ response categories, the response of polytomous item $X_j = x$, where x = 0, …, C_j, could be split into C_j dichotomous sub-items (without category 0). The item-split process function of item *j* was

$$x_{jc} = \begin{cases} 1, & \text{if } c \le x \\ 0, & \text{if } c > x \end{cases} \tag{5}$$

where *c* was the *c*th category from 1 to C_j. For example, consider a student response vector of 4 items with 3 categories $\mathbf{x}$ = (1 2 0 2), the matrix of dichotomous sub-response is

$$\mathbf{x} = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix} \tag{6}$$

The *n*th row of $\mathbf{x}$ was dichotomous sub-response of the *n*th category. This kind of sub-response is cumulative. Students can score the higher category only after scored all lower categories.

Based on the Q_C-matrix proposed in sGDINA, we proposed the cumulative Q_C-matrix (cQ_C-matrix). The cQ_C-matrix was more straightforward to adapt to the cumulative sub-response we used. We modified the Q_C-matrix to the cQ_C-matrix by two steps: (1) if $q_{cja} = 1$, set all 0 elements under category *c* to 1; (2) if q_{cja} is unknown, set all 0 elements under category *c* to unknown.

After the dichotomizing and cumulating process on data and Q-matrix, dichotomous sub-response and cQ_C-matrix can be used in two methods we mentioned above to validate the q-vector of categories.

IV. Using the Template

A. Simulation Settings

To evaluate the effects of using SOM for Q-matrix validation, we conducted two simulation studies, one for items with dichotomous responses (Study 1) and one for items with polytomous responses (Study 2). We compared the recovery accuracy (RA) of Q-matrix which validated by two methods: one-phase method and two-phase method. To make the simulation studies more comprehensive, three control conditions are considered: the sample size (N = 100, 500), the quality of items (guessing and slipping parameters = U[0, 0.2] (high item), U[0, 0.4] (low item)), and the clarity of Q-matrix (percentage of missing entries = 10% (high Q-matrix), 30% (low Q-matrix)). Taken together, each method was tested in (2×2×2 = 8) conditions in each study.

In both of the two simulation studies, the number of attribute K is set as 3. The number of output nodes in SOM is thus ($4\times2^3 = 32$). The termination criterion for the training procedure of SOM is set as 100 iterations, and each condition in simulation is replicated 100 times. We use the GDINA R package for data simulation [34], the kohonen R package for SOM method estimation [35], and the other code is written in R [36].

B. Study 1: Validating Q-matrix for items with dichotomous responses

We conducted this simulation to investigate the performance of four methods for validating the Q-matrix of dichotomous items. In this simulation, the true CDM is the GDINA model, the number of items (J) is 40, and each item measures 1 to 3 attribute(s), with each attribute measured for equal the number of times.

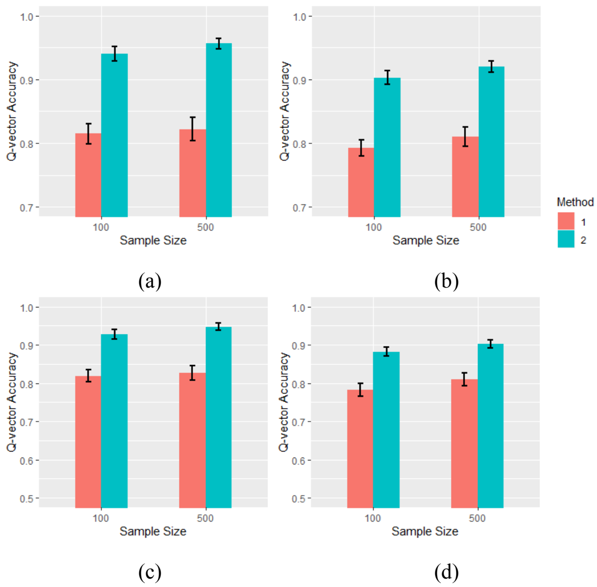

Fig. 4. The recovery accuracy of simulation study 1. (a) High Q-matrix and High Item; (b) High Q-matrix and Low Item; (c) Low Q-matrix and Low Item; and (d) Low Q-matrix and Low Item. Method 1 is two-phase method and method 2 is one-phase

Fig. 4 shows the recovery accuracy (RA) of the missing entries in Q-matrix. As can be observed, in each condition, the one-phase method achieved better RA than the two-phase method. The RAs of one-phase were 88.2%~95.6% and the RAs of two-phase were 78.1%~82.7%. Besides, the RA rose as item quality improved, sample size enlarged, or Q-matrix became more certain.

C. *Study 2: Q-matrix of polytomous responses*

We conducted this simulation to investigate the performance of four methods for validating the Q-matrix of polytomous items. In this simulation, the true CDM is the sGDINA model. The number of items (J) is 20, including three dichotomous items, 14 polytomous items with three categories, and three polytomous items with four categories. Each item measured 1 to 3 attribute(s), with each attribute measured for equal the number of times.

Similarly, Fig. 5 shows the RA of the missing entries in Q-matrix. In each condition, the one-phase method is better. The RAs of one-phase were 87.1%~93.5% and the RAs of two-phase were 81.3%~86.3%. As we expected, in most conditions, the RA rose as item quality improved, sample size enlarged, or Q-matrix became more certain. However, as the sample size decreased, the two-phase method had higher RAs in low item quality conditions, and the one-phase method had higher RAs in low item quality and high Q-matrix clarity.

D. *Study 3: Real data illustration*

In this section, we employed real data to illustrate whether and how the two proposed methods can be applied in practice. We used 31 items from booklet 8 of the PISA 2000 reading test [37]. The data were collected from 1431 students in Canada. To measure general reading literacy, test developers targeted at

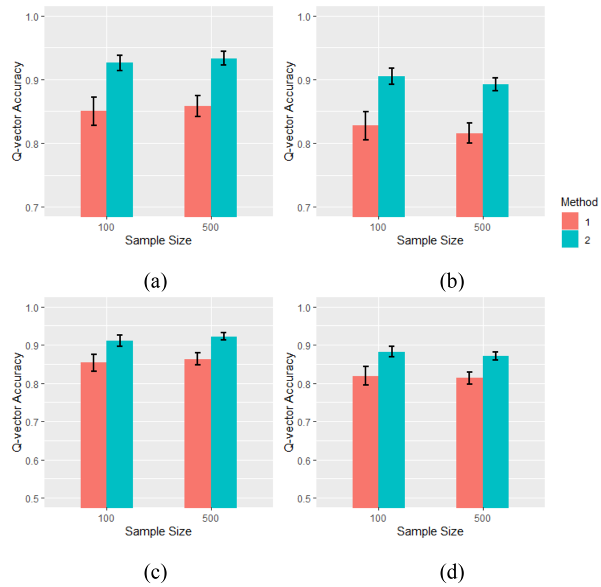

Fig. 5. The recovery accuracy of simulation study 2. (a) High Q-matrix and High Item; (b) High Q-matrix and Low Item; (c) Low Q-matrix and Low Item; and (d) Low Q-matrix and Low Item. Method 1 is two-phase method and method 2 is one-phase

three distinct reading sub-processes: “interpreting text”, “reflection and evaluation”, and “retrieving information” [37]. Each item was intended to require at least one of these processes. We treated a sub-process as an attribute and the definitions of attributes were given in TABLE III. [38]. Three items (R077Q05, R088Q03, R088Q04) were constructed polytomously with three order response categories (0, 1 and 2). For these items, we set partial credit as category 1 and full credit as category 2.

The original item-level Q-matrix was developed by Hanson [38]. To fit the sGDINA model, polytomous items (i.e. items 8, 11, and 12) required category-level q-vectors. That is, the category-level information of polytomous items was unclear. There was missing value in the category-level Q-matrix which recomposed by the original item-level Q-matrix. For example, item-level q-vector of item 12 required attribute α_1, but the category-level q-vectors of category 1 and 2 of item 12 are unknown. Thus, except for those entries which were 1, we assumed the other entries were missing. TABLE IV. shown the Q-matrix of PISA data that should be determined.

There was no true Q-matrix in practical application, so RA cannot be used to evaluate the estimated Q-matrix. TABLE V. showed the estimated Q-matrix of four methods (one-phase method, two-phase method, and traditional method with different preset Q-matrices). For polytomous item 8, one- and

TABLE III. ATTRIBUTE DEFINITIONS FOR PISA 2000 DATA.

α_1	Retrieving information:	Locating one or more pieces of information in a text.
α_2	Interpreting:	Constructing meaning and drawing inferences from one or more parts of a text.
α_3	Reflecting/evaluating:	Relating a text to one’s experience, knowledge and ideas.

TABLE IV. THE CQ_C-MATRIX FOR PISA 2000 DATA.

Item	Category	Code	Attribute α_1	α_2	α_3
1	1	R040Q02	0	0	1
2	1	R040Q03a	0	0	1
3	1	R040Q03b	0	1	0
4	1	R040Q04	1	0	0
5	1	R040Q06	1	0	0
6	1	R077Q02	0	0	1
7	1	R077Q04	1	0	0
8	**1**	**R077Q05**	**999**	**999**	**999**
8	**2**	**R077Q05**	**999**	**1**	**999**
9	1	R077Q06	1	0	0
10	1	R088Q01	1	0	0
11	**1**	**R088Q03**	**999**	**999**	**999**
11	**2**	**R088Q03**	**999**	**999**	**1**
12	**1**	**R088Q04**	**999**	**999**	**999**
12	**2**	**R088Q04**	**1**	**999**	**999**
13	1	R088Q05	0	1	0
14	1	R088Q07	0	1	0
15	1	R110Q01	1	0	0
16	1	R110Q04	0	0	1
17	1	R110Q05	0	0	1
18	1	R110Q06	0	1	0
19	1	R216Q01	1	0	0
20	1	R216Q02	0	1	0
21	1	R216Q03	1	0	0
22	1	R216Q04	0	0	1
23	1	R216Q06	1	0	0
24	1	R236Q01	1	0	0
25	1	R236Q02	1	0	0
26	1	R237Q01	0	0	1
27	1	R237Q03	1	0	0
28	1	R239Q01	1	0	0
29	1	R239Q02	0	0	1
30	1	R246Q01	0	0	1
31	1	R246Q02	0	0	1

[b] Polytomous items are shown in bold. "999" denoted an unknown element.

two-phase methods suggested that two categories both required attributes α_1 and α_2. For items 11 and 12, different methods suggested that category required different attribute patterns. The problems of traditional methods were that depended on different preset Q-matrices, traditional methods did not change the category q-vectors of polytomous items. For dichotomous items, the two-phase method suggested a lot of modifications compared with the other methods. In summary, the one-phase method is best in practical applications.

V. DISCUSSION AND FURTHER STUDIES

The purpose of this study was to describe how unsupervised ANN can be used to validate Q-matrix on CDAs and to evaluate the performances of ANNs using simulation and real data illustration. CDA was designed to evaluate student performance on the finer-grained skills based on the Q-matrix. The importance of Q-matrix validation had been recently recognized. The SOM one-phase method we proposed is non-parametric and model-independent compared with the traditional Q-matrix validation method. According to the simulation studies and real data illustration, we found that the SOM one-phase method had the best performance which indicated that the SOM one-phase method could be applied widely with high accuracy.

TABLE V. FOUR ESTIMATED Q-MATRICES.*

	One-phase			Two-phase			Traditional 1			Traditional 2		
Item	α_1	α_2	α_3	α_1	α_2	α_3	α_1	α_2	α_3	α_1	α_2	α_3
1	0	0	1	1*	1*	1	0	0	1	0	0	1
2	0	0	1	1*	1*	0*	0	0	1	0	0	1
3	0	1	0	1*	1	0	0	1	0	0	1	0
4	1	0	0	0*	0	1*	1	0	0	1	0	0
5	1	0	0	1	1*	0	1	1*	0	1	0	0
6	0	0	1	0	0	1	0	0	1	0	0	1
7	1	0	0	0*	1*	1*	1	1*	0	1	0	0
8[b]	**1**	**1**	**0**	**1**	**1**	**0**	**1**	**1**	**1**	**0**	**1**	**0**
8[b]	**1**	**1**	**0**	**1**	**1**	**0**	**1**	**1**	**1**	**0**	**1**	**0**
9	1	0	0	1	1*	0	1	1*	0	1	0	0
10	1	0	0	1	1*	0	1	0	0	1	0	0
11[b]	**1**	**0**	**0**	**0**	**0**	**1**	**1**	**1**	**1**	**0**	**0**	**1**
11[b]	**1**	**0**	**1**	**1**	**1**	**0**	**1**	**1**	**0**	**0**	**0**	**1**
12[b]	**0**	**0**	**1**	**0**	**0**	**1**	**1**	**1**	**1**	**1**	**0**	**0**
12[b]	**1**	**0**	**1**	**1**	**1**	**0**	**1**	**1**	**0**	**1**	**0**	**0**
13	0	1	0	0	0*	0	0	1	0	0	1	0
14	0	1	0	0	0*	1*	0	1	0	0	1	0
15	0*	0	1*	0*	0	0	1	1*	0	1	0	0
16	0	0	1	0	0	0*	0	0	1	0	0	1
17	0	0	1	0	0	0*	0	0	1	0	0	1
18	0	1	0	0	0*	0	0	1	0	0	1	0
19	1	0	0	1	0	0	1	0	0	1	0	0
20	0	1	0	1*	1	0	0	1	0	0	1	0
21	1	0	0	1	1*	0	1	0	0	1	0	0
22	0	0	1	1*	1*	0*	0	0	1	0	0	1
23	1	0	0	1	0	0	1	0	0	1	0	0
24	1	0	0	1	0	0	1	0	0	1	0	0
25	1	0	0	1	1*	0	1	0	0	1	0	0
26	0	0	1	0	0	1	0	0	1	0	0	1
27	1	0	0	1	1*	0	1	0	0	1	0	0
28	1	0	0	1	1*	0	1	0	0	1	0	0
29	0	0	1	0	0	1	0	0	1	0	0	1
30	0	0	1	0	0	0*	0	0	1	0	1*	1
31	0	0	1	1*	1*	0*	0	0	1	0	0	1

[b] Polytomous items are shown in bold.

* noted a modified entry.

Compared with traditional methods, SOM methods had some advantages. First, most traditional methods had the assumption about the specific CDM, but SOM methods can be used in any item. Second, in simulation, SOM methods had high recovery accuracy in a small sample size when traditional methods asked a big sample size (e.g. $N = 1000$). Third, most traditional methods required that the preset Q-matrix was largely correct. When there was missing value in Q-matrix, missing values must be set as 0 or 1 to use traditional methods. Thus, traditional methods did not suggest any validation on category q-vectors. However, missing values can be retained in SOM methods.

Our results showed that ANN was suited to the Q-matrix validation problem better than the traditional statistical index methods. The possible q-vector pattern was regarded as a high-dimension network when treating each attribute as a dimension. However, because the essence of the Q-matrix validation problem was a cluster and classification problem, ANN needs to be compared with other clustering or classification algorithms. Further, optimizing ANN algorithms can accommodate more situations of this issue, for example, polytomous attributes.

REFERENCES

[1] K. K. Tatsuoka, "Multivariate applications series. Cognitive assessment: An introduction to the rule space method.," Routledge/Taylor & Francis Group, 2009. .

[2] K. K. Tatsuoka, "Rule space: an approach for dealing with misconceptions based on item response theory," J. Educ. Meas., vol. 20, no. 4, pp. 345–354, 1983.

[3] J. de la Torre, "An empirically based method of Q-matrix validation for the DINA model: Development and applications," J. Educ. Meas., vol. 45, pp. 343–362, 2008.

[4] J. de la Torre and C. Y. Chiu, "A General Method of Empirical Q-matrix Validation," Psychometrika, vol. 81, no. 2, pp. 253–273, 2016.

[5] C.-Y. Chiu, "Statistical refinement of the q-matrix in cognitive diagnosis," Appl. Psychol. Meas., vol. 37, no. 8, pp. 598–618, 2013.

[6] A. A. Rupp and J. Templin, "The effects of Q-matrix misspecification on parameter estimates and classification accuracy in the DINA model," Educ. Psychol. Meas., vol. 68, pp. 78–96, 2008.

[7] W. Ma and J. de la Torre, "An empirical Q-matrix validation method for the sequential generalized DINA model," Br. J. Math. Stat. Psychol., 2019.

[8] L. T. DeCarlo, "On the analysis of fraction subtraction data: The DINA model, classification, latent class sizes, and the q-matrix.," Appl. Psychol. Meas., vol. 35, no. 1, pp. 8–26, 2011.

[9] J. P. Leighton, M. J. Gierl, and S. M. Hunka, "The attribute hierarchy method for cognitive assessment: A variation on Tatsuoka's rule-space approach.," J. Educ. Meas., vol. 41, pp. 205–237, 2004.

[10] M. J. Gierl, Y. Cui, and S. Hunka, "Using connectionist models to evaluate examinees' response patterns on tests.," J. Mod. Appl. Stat. Methods, vol. 7, pp. 234–245, 2008.

[11] J. de la Torre and J. A. Douglas, "Higher-order latent trait models for cognitive diagnosis," Psychometrika, vol. 69, no. 3, pp. 333–353, 2004.

[12] Z. Shu, R. Henson, and J. Willse, "Using neural network analysis to define methods of dina model estimation for small sample sizes.," J. Classif., vol. 30, no. 2, pp. 173–194, 2013.

[13] R. K. Price, E. L. Spitznagel, T. J. Downey, D. J. Meyer, and N. K. Risk, "Applying artificial neural network models to clinical decision making," Psychol. Assess., vol. 12, pp. 40–51, 2000.

[14] R. H. Stevens, J. Ikeda, A. M. Casillas, J. Palacio-Cayetano, and S. G. Clyman, "Artificial neural network-based performance assessments.," Comput. Human Behav., vol. 15, pp. 295–313, 1999.

[15] W. Wang, S. Ding, L. Song, Z. Kuang, and H. Cao, "Application of Neural Networks and Support Vector Machines to Cognitive Diagnosis," J. Psychol. Sci., vol. 39, no. 4, pp. 777–782, 2016.

[16] Y. Cui, M. Gierl, and Q. Guo, "Statistical classification for cognitive diagnostic assessment : an artificial neural network approach Statistical classification for cognitive diagnostic assessment : an," Educ. Psychol., vol. 36, no. 6, pp. 1069–1092, 2016.

[17] J. L. Templin and R. A. Henson, "Measurement of psychological disorders using cognitive diagnosis models," Psychol. Methods, vol. 11, pp. 287–305, 2006.

[18] J. de la Torre, "The Generalized DINA Model Framework," Psychometrika, vol. 76, no. 2, pp. 179–199, 2011.

[19] M. von Davier, "A general diagnostic model applied to language testing data," Br. J. Math. Stat. Psychol., vol. 61, pp. 287–307, 2008.

[20] J. de la Torre and J. Douglas, "Model evaluation and multiple strategies in cognitive diagnosis: An analysis offraction subtraction data," Psychometrika, vol. 73, pp. 595–624, 2008.

[21] M. Birenbaum and K. K. Tatsuoka, "Open-ended versus multiple-choice response formats - It does make a difference for diagnostic purposes," Appl. Psychol. Meas., vol. 11, pp. 385–395, 1987.

[22] W. Ma and J. de la Torre, "A sequential cognitive diagnosis model for polytomous responses," Br. J. Math. Stat. Psychol., vol. 69, no. 3, pp. 253–275, 2016.

[23] J. Chen and J. de la Torre, "Introducing the General Polytomous Diagnosis Modeling Framework," Front. Psychol., vol. 9, no. August, pp. 1–9, 2018.

[24] B. W. Junker and K. Sijtsma, "Cognitive assessment models with few assumptions, and connections with nonparametric item response theory," Appl. Psychol. Meas., vol. 25, pp. 258–272, 2001.

[25] H. Li and H. K. Suen, "Constructing and Validating a Q-Matrix for Cognitive Diagnostic Analyses of a Reading Test," Educ. Assess., vol. 18, no. 1, pp. 1–25, 2013.

[26] L. T. DeCarlo, "Recognizing uncertainty in the Q-matrix via a Bayesian extension of the DINA model," Appl. Psychol. Meas., vol. 36, pp. 447–468, 2012.

[27] T. Kohonen, "Self-organized formation of topologically correct feature maps," Biological Cybernetics, vol. 43, no. 1, pp. 59–69, 1982.

[28] T. Kohonen, "The self-organizing map," Proc. IEEE, vol. 78, pp. 1464–1480, 1990.

[29] T. Kohonen, Self-organizing maps (3rd ed.). 2001.

[30] P. Li and X. Zhao, "Self-organizing map models of language acquisition," Front. Psychol., vol. 4, pp. 828–828, 2013.

[31] H. Shahhosseini and R. Safabakhsh, "TASOM: a new time adaptive self-organizing map," IEEE Trans. Syst. Man, Cybern. Part B, vol. 33, no. 2, pp. 271–282, 2003.

[32] H.-U. Bauer and T. Villmann, "Growing a hypercubical output space in a self-organizing feature map," IEEE Transactions on Neural Networks, vol. 8, no. 2, pp. 218–226, 1997.

[33] E. Berglund and J. Sitte, "The parameterless self-organizing map algorithm," IEEE Transactions on Neural Networks, vol. 17, no. 2, pp. 305–316, 2006.

[34] W. Ma and J. de la Torre, "GDINA: The generalized DINA model framework. [Computer software version 2.7.3]," Retrieved from https://CRAN.R-project.org/package=GDINA, 2019.

[35] F. Leisch, E. Dimitriadou, and B. Gruen, "flexclust: Flexible Cluster Algorithms. [Computer software version 1.4.0]," Retrieved from https://CRAN.R-project.org/package= flexclust, 2018.

[36] R. C. Team, "R: A language and environment for statistical computing [Computer software manual]," Vienna, Austria. Retrieved from https://www.R-project.org/, 2019.

[37] R. J. Adams and M. L. Wu, "PISA 2000 Technical Report," OECD Publishing, Paris, 2002.

[38] M. P. Hansen, "Hierarchical item response models for cognitive diagnosis," Proquest Llc, vol. 191, 2013.