Structural Learning of BN2A models

Iván Pérez[1,2,3] and Jiří Vomlel[1]

[1]Institute of Information Theory and Automation, Czech Academy of Sciences
{*cabrera, vomlel*}*@utia.cas.cz*
[2]Institute of Computer Science, Czech Academy of Sciences
[3]Faculty of Mathematics and Physics, Charles University

Abstract

Probabilistic graphical models, particularly Bayesian networks, provide a flexible framework for representing dependencies among random variables and have been widely applied in domains such as medicine, biology, and educational testing. Our work focuses on BN2A networks - a specialized bipartite Bayesian network architecture, where the first layer consists of hidden variables and the second layer consists of observed variables. In BN2A models all variables are assumed to be binary. The variables in the second layer depend on the variables in the first layer and this dependence is characterized by conditional probability tables representing Noisy-AND models. In this work, we propose an Expectation-Maximization (EM) algorithm for learning the structure of BN2A models, that is, for learning the relationship between hidden variables and observed variables. To validate our structural learning algorithm, we designed two experiments using educational assessment data. For the first experiment, we used synthetic data generated from a BN2A model that we previously defined, while for the second experiment we used a well-known real-world dataset in the field of Cognitive Diagnostic Models, the Fraction Subtraction dataset. Our proposed algorithm has interesting potential use cases. One key application is to generate a reasonably accurate BN2A structure model for educational diagnosis, particularly in scenarios where no prior model exists. Depending on the required level of accuracy, the estimated model can be used directly to analyze skill profiles or serve as an initial framework for test designers, who can further refine it before implementation.

1 Introduction

Bayesian networks are a popular framework for modeling probabilistic relationships between random variables and have been used successfully in educational tests (Almond et al. (2015), Vomlel (2004)). There is interest in a particular type of Bayesian networks we have called BN2A, which are characterized as bipartite networks, where the first layer consists of hidden variables (which commonly represent skills) and the second layer consists of observed variables (which represent questions in a test). In BN2A models all variables are assumed to be binary. The variables in the second layer depend on the variables in the first layer and this dependence is characterized by conditional probability tables (CPTs) representing a Noisy-AND model. In Fig. 1 we give an example of a directed bipartite graph that can define the structure of a BN2A model.

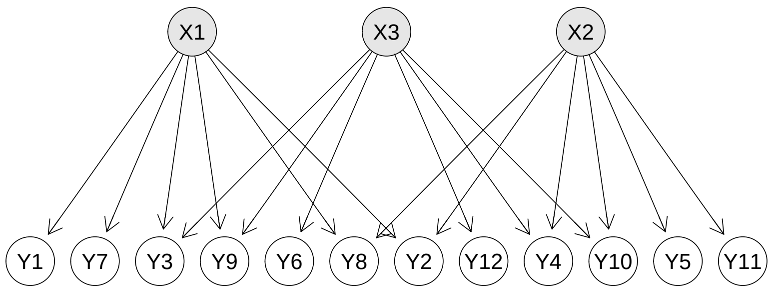

Figure 1: BN2A model with three hidden variables and 12 observed variables.

This paper is structured as follows. In Section 2 we formally introduce the BN2A model and present their corresponding CPTs, leaky Noisy-AND. In Section 3 we present an algorithm for learning the structure of the BN2A models while in Section 4, we illustrate how this algorithm works conducting two experiments. For the first experiment, we used synthetic data generated from a BN2A model that we previously defined, while for the second experiment we used a well-known real-world dataset in the field of Cognitive Diagnostic Models, the Fraction Subtraction dataset (Tatsuoka, 2002). Finally, we summarise the contribution of this paper and discuss our future work in Section 5.

2 BN2A model

Let $\mathbf{X}$ denote the vector $(X_1, \ldots, X_K)$ of K hidden variables, and similarly let $\mathbf{Y}$ denote the vector $(Y_1, \ldots, Y_L)$ of L observed dependent variables. The hidden variables are also called attributes or skills in the context of cognitive diagnostic models (CDMs), and observed dependent variables are usually called items or questions in the same context. All variables are assumed to be binary, taking states from $\{0, 1\}$. The state space of the multidimensional variable $\mathbf{X}$ is denoted $\mathbb{X}$ and is equal to the Cartesian product of the state spaces of $X_k, k = 1, \ldots, K$:

$$\mathbb{X} \quad = \quad \times_{k=1}^{K} \mathbb{X}_k = \{0, 1\}^K \quad . \qquad (1)$$

Similarly, the state space of multidimensional variable $\mathbf{Y}$ is denoted $\mathbb{Y}$ and is equal to the Cartesian product of state spaces of $Y_\ell, \ell = 1, \ldots, L$:

$$\mathbb{Y} \quad = \quad \times_{\ell=1}^{L} \mathbb{Y}_\ell = \{0,1\}^L \ . \tag{2}$$

The basic building blocks of a BN2A model are conditional probability tables (CPTs) specified in the form of a Noisy-AND model. Let Y_ℓ be an observed dependent variable and $pa(Y_\ell)$ be the subset of indexes of related variables from $\mathbf{X}$. They are referred to as the parents of Y_ℓ.

Definition 1 (Noisy-AND model)**.**
A conditional probability table $P(Y_\ell|\mathbf{X}_{pa(Y_\ell)})$ represents a Noisy-AND model if

$$P(Y_\ell = y_\ell | \mathbf{X}_{pa(Y_\ell)} = \mathbf{x}_{pa(Y_\ell)}) \quad = \quad \begin{cases} q_{\ell,0} \cdot \prod_{i \in pa(Y_\ell)} (q_{\ell,i})^{(1-x_i)} & \text{if } y_\ell = 1 \\ 1 - q_{\ell,0} \cdot \prod_{i \in pa(Y_\ell)} (q_{\ell,i})^{(1-x_i)} & \text{if } y_\ell = 0. \end{cases} \tag{3}$$

Note that if $x_i = 1$ then $(q_{\ell,i})^{(1-x_i)} = 1$ and if $x_i = 0$ then $(q_{\ell,i})^{(1-x_i)} = q_{\ell,i}$. The interpretation is that if $X_i = 1$, then this variable definitely enters the AND relation with the value 1. If $X_i = 0$, then there is still a probability $q_{\ell,i}$ that it enters the AND relation with value 1. The model also contains an auxiliary parent X_0 which is always 0 and thus enters the AND relation with probability $q_{\ell,0}$ for the value 1. This probability is traditionally called *leak* probability and allows non-zero probability of $Y_\ell = 0$ even if all parents of Y_ℓ have value 1. In CDM, this model is known as the Reduced Reparametrized Unified Model (RRUM) (Hartz and Roussos, 2008) and it is a special case of the Generalized Noisy Inputs, Deterministic AND (GNIDA) gate model (de la Torre, 2011).

The prior probability of the hidden skill for $k = 1, \ldots, K$ is defined as

$$P(X_k = x_k) \quad = \quad (p_k)^{x_k}(1-p_k)^{(1-x_k)} \ , \tag{4}$$

which means that if $x_k = 1$ then it is p_k and if $x_k = 0$ then it equals $1 - p_k$.

Now we are ready to define a special class of Bayesian network models with hidden variables, called BN2A model.

Definition 2 (BN2A model)**.**
A BN2A model is a pair (G, P), where G is a directed bipartite graph with its nodes divided into two layers. The nodes of the first layer correspond to the hidden variables $X_1, \ldots, X_K$ and the nodes of the second layer correspond to the observed variables $Y_1, \ldots, Y_L$. All edges are directed from a node of the first layer to a node of the second layer. The symbol P refers to the joint probability distribution over the variables corresponding to the nodes of the graph G. The probability distribution is parameterized by a vector of model parameters $(\mathbf{p}, \mathbf{q})$:

$$(\mathbf{p}, \mathbf{q}) \quad = \quad \left((p_k)_{k \in \{1,\ldots,K\}}, (q_{\ell,k})_{\ell \in \{1,\ldots,L\}, k \in \{0\} \cup pa(Y_\ell)}\right) \ . \tag{5}$$

The bipartite graph G of a BN2A model can also be specified by an incidence matrix, in the context of CDM, traditionally denoted by $\mathbf{Q}$. A $\mathbf{Q}$-matrix is an $L \times K$ binary matrix, with entries $\mathbf{Q}_{\ell,k} \in \{0,1\}$ that indicate whether or not the ℓ^{th} observed dependent variable is linked to the k^{th} hidden variable:

$$\mathbf{Q}[\ell,k] \quad = \quad \begin{cases} 1 & \text{if } X_k \in pa(Y_\ell) \\ 0 & \text{otherwise.} \end{cases}$$

3 Structural learning

As previously mentioned, our primary focus lies in applying BN2A models to educational testing. In practice, education experts often possess test outcome data but lack a predefined cognitive model structure. This creates a fundamental need for computational tools that can automatically learn the underlying model structure from available data — particularly important since BN2A models can not only effectively represent the relationship between latent skills and observed responses, but are also inherently interpretable for educational applications.

In this section we propose a method for learning the structure of a BN2A model from data $\mathbf{D}$ where states of variables $\mathbf{Y}$ are observed but variables $\mathbf{X}$ are hidden, i.e., their states are unobserved. The algorithm is basically a version of the structural EM algorithm proposed by Friedman (1998).

We further assume that the number K of hidden variables is known.[1] Clearly, a naive approach of evaluating all possible BN2A structures would quickly become intractable for larger values of K and L.

Recall that L denotes the dimension of the vector of observed variables $\mathbf{Y} = (Y_1, \ldots, Y_L)$ and K denotes the dimension of the vector of hidden variables $\mathbf{X} = (X_1, \ldots, X_K)$. Let for all $\mathbf{x} \in \mathbb{X}$ the function $N(\mathbf{x})$ denote the number of occurrences of vector $\mathbf{x}$ in data $\mathbf{D}$. Similarly, let for all $(\mathbf{x}, \mathbf{y}) \in \mathbb{X} \times \mathbb{Y}$ the function $N(\mathbf{x}, \mathbf{y})$ denote the estimated number[2] of occurrences of vector $(\mathbf{x}, \mathbf{y})$ in completed data $\mathbf{C}$ consisting of values of $(\mathbf{X}, \mathbf{Y})$.

3.1 Algorithm

The structure learning algorithm is presented in Algorithm 1. This algorithm requires a dataset $\mathbf{D}$ consisting of n data vectors with values of $\mathbf{Y}$ and the dimension K of $\mathbf{X}$ as its input. In general, the algorithm alternates between two major steps - the E-step and the M-step as described below.

[1] Generally, one can consider K also as a free parameter and from best models of different K values select one that maximizes an evaluation criteria, which in our study is BIC.

[2] The number of occurrences is computed as their expected number. Therefore, the numerical values are non-negative real numbers, i.e., they need not (and typically they are not) natural numbers. This does not cause any problems in subsequent computations.

```
Input  : D – dataset consisting of n data vectors with values of Y
         K – the dimension of X
         Q' – initial Q matrix, e.g. the unit L × K matrix
         (p, q) – initial parameter values, e.g. GDINA(D, Q', K, "RRUM")
Output: Q – the Q-matrix of BN2A
         p – prior probabilities of the hidden variables
         q – parameters of the Noisy-AND models

Set Q as the zero L × K matrix;
while Q ≠ Q' do
    if Q is not the zero L × K matrix then
        Q' ← Q;
    end
    E-step
    for x ∈ {0,1}^K do
        P(x) = ∏_{k=1}^{K} (p_i)^{x_i} · (1 − p_i)^{1−x_i} ;
        for ℓ ∈ {1,...,L} do
            pa(Y_ℓ) = {k ∈ {1,...,K} : Q'[ℓ,k] = 1};
            R = {0} ∪ pa(Y_ℓ) ;
            P(Y_ℓ = 1|x) = ∏_{i∈R} (q_{ℓ,i})^{1−x_i};
            P(Y_ℓ = 0|x) = 1 − ∏_{i∈R} (q_{ℓ,i})^{1−x_i};
        end
    end
    M-step
    p = (N(x_1)/n, ..., N(x_k)/n);
    for ℓ ∈ {1,...,L} do
        M = {1,...,L} \ {ℓ};
        N(x, y_ℓ) = Σ_{y_M} N(x, y);
        BIC* = −∞;
        for R ⊆ {1,...,K} do
            for (x_R, y_ℓ) ∈ {0,1}^{|R|+1} do
                S = {1,...,K} \ R;
                N(x_R, y_ℓ) = Σ_{x_S} N(x, y_ℓ) ;
            end
            q_{ℓ,R} = arg max_{q'} LL(q');
            BIC = LL(q_{ℓ,R}) − (log n)/2 · (|R| + 1);
            if BIC > BIC* then
                BIC* = BIC;
                q_ℓ = q_{ℓ,R};
                for k ∈ {1,...,K} do
                    Q[ℓ,k] ← 𝕀(k ∈ R) ;
                end
            end
        end
    end
end
```

Algorithm 1: Learning the structure of BN2A.

E-step

In the E-step it is assumed a model structure specified by a matrix $\mathbf{Q}$ is known. In the first iteration we assume the structure is a complete graph represented by an all-ones matrix $\mathbf{Q}$. For the given $\mathbf{Q}$ the parameters of Noisy-AND models of all CPTs are learned using an R library called GDINA (Ma and de la Torre, 2020) for the particular RRUM model, which is equivalent to the BN2A model. These parameters are then used to estimate values of hidden variables. In this way a complete data set $\mathbf{C}$ is created.

M-step

The complete data $\mathbf{C}$ represent the input of the M-step. In this step a BN2A model that best describes the data is learned. As the model quality criteria the well-known BIC criterion is used. It is the log-likelihood penalized by a penalty proportional to the number of model parameters. The important observation is that the learning algorithm finds the best parent set and optimal parameters independently for each variable Y_ℓ, $\ell = 1, \ldots, L$, which still guarantees the global optimality. This does not hold for general Bayesian networks since the independent learning may result in directed cycles that are forbidden. But for models with their structure defined by a bipartite graph the global optimality is guaranteed since bipartite graphs cannot contain directed cycles. For each considered parent set the parameters of the Noisy-AND model maximizing the log-likelihood for the given data are learned using a gradient method and its BIC is computed. Then, the parent set with the maximum BIC is selected, and based on it, the corresponding row of the matrix $\mathbf{Q}$ is formed, which serves as input for the E-step. An important observation is that the log-likelihood of Noisy-AND is a nicely shaped function with a unique maximum. This follows from the concavity of the log-likelihood of the Noisy-AND function:

$$\begin{aligned} LL(\mathbf{q}_\ell) &= \sum_{\mathbf{x}_R} N(\mathbf{x}_R, Y_\ell = 1) \cdot \log \prod_{i \in T} (q_{\ell,i})^{1-x_i} \\ &\quad + \sum_{\mathbf{x}_R} N(\mathbf{x}_R, Y_\ell = 0) \cdot \log \left(1 - \prod_{i \in T} (q_{\ell,i})^{1-x_i} \right) . \end{aligned} \tag{6}$$

The concavity can be proven similarly as in (Vomlel et al., 2023, Lemma 1).

4 Experiments

To show how Algorithm 1 works, we conducted two experiments. For the first experiment, we used synthetic data generated from a BN2A model representing a test consisting of 12 items measuring three skills, while for the second experiment we used a well-known real-world dataset in the field of Cognitive Diagnostic Models, the Fraction Subtraction dataset (Tatsuoka, 2002).

4.1 A BN2A model with $K = 3$ and $L = 12$

For the first experiment, we used the structure shown in Figure 1 which presents dependencies of 12 questions on three proposed skills. In this model, note that that six questions require only one skill, while the others questions require two skills to be answered correctly.

As in any knowledge domain, some skills are easy to master and others are difficult; thus, tests typically require skills of varying mastery prevalence in the population. In our experiment, the simulated proportion of skill mastery (prior probabilities of mastering the skills) was set as $p_1 = 0.8$, $p_2 = 0.6$, and $p_3 = 0.4$. The leak parameters $q_{\ell,0}$ represent the probability of answering question ℓ correctly when the student masters all required skills. These parameters were selected to range from 0.6 to 0.9. The failure parameters $q_{\ell,k}$ act as penalty factors for lacking the k-th skill when answering question ℓ. These parameters were randomly chosen in the range of 0.1 to 0.4 to have a realistic model. With the structure in Figure 1 and the above parameters, we randomly generated a dataset $\mathbf{D}$ of size $n = 10^4$ —a feasible sample size for large-scale standardized tests.

Before running the algorithm, we randomly created ten initial vectors in the parameter space. For each, we computed the log-likelihood of the complete model (unit Q-matrix) using the GDINA library (Ma and de la Torre, 2020), and initialized the algorithm using the parameter vector corresponding to the highest log-likelihood value. This helps avoid getting stuck in a local maximum. Once we execute Algorithm 1 using the dataset $\mathbf{D}$ and the proposed number of latent variables ($K = 3$), the model structure was correctly learned in the first iteration, and was completed in the second iteration, confirming that the structure was the same in both iterations.

The algorithm learned the correct structure but with a different labeling of the hidden variables. To facilitate the comparison of the results we have permuted the columns of the learned model. Table 1 compares the prior probabilities for each skill, while Table 2 compares the leak and failure parameters for each test question. In Table 2, a dash (-) indicates that there is no relationship between the question and the corresponding skill. In general, it can be seen that, for the learned model, the prior probabilities and the leak and failure parameters are close to the original model.

p_1	p_2	p_3	p_1	p_2	p_3
0.800	0.600	0.400	0.7943	0.5954	0.4028

Table 1: Comparison of prior probabilities for skill mastery: original values (left) and learned values (right).

4.2 Tatsuoka's fraction subtraction dataset

The fraction subtraction test (Tatsuoka and Tatsuoka, 1987) was designed as a diagnostic tool to detect common error patterns and maladaptive solution strategies in fraction arithmetic. A detailed presentation of the test questions appears in Table 3.

ℓ	$q_{\ell,0}$	$q_{\ell,1}$	$q_{\ell,2}$	$q_{\ell,3}$	ℓ	$q_{\ell,0}$	$q_{\ell,1}$	$q_{\ell,2}$	$q_{\ell,3}$
1	0.900	0.400	-	-	1	0.8936	0.3707	-	-
2	0.800	0.200	0.100	-	2	0.7885	0.1942	0.1079	-
3	0.700	0.100	-	0.300	3	0.6836	0.0725	-	0.3059
4	0.600	-	0.200	0.100	4	0.5990	-	0.1897	0.0895
5	0.700	-	0.200	-	5	0.6916	-	0.2097	-
6	0.800	-	-	0.300	6	0.7867	-	-	0.2965
7	0.950	0.300	-	-	7	0.9473	0.2889	-	-
8	0.850	0.400	0.200	-	8	0.8522	0.4057	0.2092	-
9	0.750	0.300	-	0.200	9	0.7363	0.2969	-	0.1917
10	0.650	-	0.200	0.400	10	0.6378	-	0.1905	0.3929
11	0.750	-	0.100	-	11	0.7438	-	0.1070	-
12	0.850	-	-	0.200	12	0.8457	-	-	0.2016

Table 2: Comparison of leak and failure parameters: original values (left) and learned values (right).

The fraction subtraction dataset has become a benchmark in cognitive diagnostic modeling research, serving as a foundational testbed for over three decades. This seminal dataset has been extensively used to develop, validate, and compare diagnostic classification models due to its well-documented cognitive structure and pedagogical relevance. Its popularity stems from the clear mapping between mathematical skills and item responses, making it an ideal case study for educational testing research. The dataset's widespread adoption across numerous studies (e.g., de la Torre and Douglas (2004), DeCarlo (2011), Culpepper (2019)) establishes it as a gold standard for evaluating new methodological approaches in cognitive diagnosis.

The dataset contains binary response patterns from N = 536 middle school students on J = 20 test questions. The data matrix (536 × 20) represents each student's performance, where rows correspond to individual students and columns represent test questions. A value of 1 indicates a correct response, while 0 denotes an incorrect response.

Fraction subtraction problems were constructed to include the basic skills required for solving problems correctly, such as borrowing, converting a whole number to a simple fraction, and getting the common denominator. Each question assesses combinations of eight core cognitive skills. Table 4 displays the operational definitions of the eight measured skills while Table 5 (left) shows the corresponding skill-question mapping (Q-matrix) for the Fraction subtraction test.

Several noteworthy observations emerge from examining Tatsuoka's proposed Q-matrix. First, the importance of the skill X_7 (Subtract numerators) becomes evident, as it is required for all test items except Y_9. Second, half of the measured skills (specifically X_1, X_3, X_6, and X_8) appear in no more than three questions each. This distribution aligns with both the skill definitions and question requirements: for instance, while many problems involve mixed numbers (whole numbers with fractions), only specific items like Y_7, Y_{15} and Y_{19} necessitate skill X_1 (Convert a whole number to a fraction) for their solution.

No.	Question	No.	Question
1	$\frac{5}{3} - \frac{3}{4}$	11	$4\frac{1}{3} - 2\frac{4}{3}$
2	$\frac{3}{4} - \frac{3}{8}$	12	$\frac{11}{8} - \frac{1}{8}$
3	$\frac{5}{6} - \frac{1}{9}$	13	$3\frac{3}{8} - 2\frac{5}{6}$
4	$3\frac{1}{2} - 2\frac{3}{2}$	14	$3\frac{4}{5} - 3\frac{2}{5}$
5	$4\frac{3}{5} - 3\frac{4}{10}$	15	$2 - \frac{1}{3}$
6	$\frac{6}{7} - \frac{4}{7}$	16	$4\frac{5}{7} - 1\frac{4}{7}$
7	$3 - 2\frac{1}{5}$	17	$7\frac{3}{5} - 2\frac{4}{5}$
8	$\frac{2}{3} - \frac{2}{3}$	18	$4\frac{1}{10} - 2\frac{8}{10}$
9	$3\frac{7}{8} - 2$	19	$4 - 1\frac{4}{3}$
10	$4\frac{4}{12} - 2\frac{7}{12}$	20	$4\frac{1}{3} - 1\frac{5}{3}$

Table 3: Tatsuoka's Fraction subtraction test

In our experiment, we tested different values of K ranging from 2 to 5. Table 5 (right) presents the learned Q-matrix and represents the optimal question-skill relationships for $K = 5$. To compare the two Q-matrices in Table 5, we computed their BIC values: the Tatsuoka expert-defined Q-matrix yields -5,303.9, while our data-learned Q-matrix achieves -4,635.5. This result suggests that our model, despite its lower dimensionality (5 vs. 8 skills), captures the data's essential structure more parsimoniously.

When learning a model's structure solely from response data, the latent variables lack explicit definitions. However, by analyzing the test items and their associations with learned skills (hidden variables), we can interpret these relationships. Our analysis of the learned Q-matrix reveals two significant patterns:

Skill X_4 appears in all items except Y_{18}, aligning with Tatsuoka's definition of "Subtract numerators" as a fundamental operation in fraction subtraction. Skill X_2 is associated with 15 of the 20 test items, with 12 of these matching Tatsuoka's original classification as "Separate a whole number from a fraction" - indicating the learned structure successfully recovers this semantically meaningful dimension from response patterns alone.

Examining the learned model parameters, the prior probabilities are: $p_1 = 0.582, p_2 = 0.525, p_3 = 0.764, p_4 = 0.805$, and $p_5 = 0.707$. Notably, the highest prior probability corresponds to X_4 (that we align with "Subtract numerators"), suggesting this is a foundational skill typically mastered even before formal fraction operation instruction.

Table 6 presents the corresponding BN2A model parameters (leak and failure parameters) of the learned structure. The leak parameters ($q_{\ell,0}$) generally exceed 0.8, consistent with theoretical expectations - these represent the probability of correct response given

Skill	Description
X_1	Convert a whole number to a fraction.
X_2	Separate a whole number from a fraction.
X_3	Simplify before subtracting.
X_4	Find a common denominator.
X_5	Borrow from whole number part.
X_6	Column borrow to subtract the second numerator from the first.
X_7	Subtract numerators.
X_8	Reduce answers to simplest form.

Table 4: Skills definition for the Fraction subtraction test.

mastery of required skills. The single exception ($q_{13,0} = 0.684$) aligns with failure parameters that strongly penalize lack of skill mastery.

Finally, the BN2A model's failure parameters ($q_{\ell,k}$) quantify the performance penalty for lacking specific skills per question. These values could help educators to: (a) identify particularly skill-sensitive items, (b) develop targeted instructional interventions, and (c) create adaptive assessment variants.

5 Discussion

In this paper, we focused on BN2A models—Bayesian networks where conditional probability tables (CPTs) are represented by Noisy-AND models with a bipartite graph structure, and all nodes in the first layer are hidden. These models are of particular interest due to: a) their interpretability in educational testing contexts, and b) their parameter efficiency: the number of parameters scales proportionally to $K \cdot L$, significantly fewer than in general bipartite Bayesian networks where CPTs can grow exponentially.

We proposed a Structural EM algorithm for learning the structure of BN2A models. A key advantage of this approach is that it independently identifies the optimal parent set and parameters for each observed variable while still guaranteeing global optimality, a property not shared by general Bayesian network learning methods.

To validate the algorithm, we conducted two experiments. On synthetic data the algorithm successfully recovered the ground-truth BN2A structure and estimated parameters with high accuracy. On Real-world data – the Fraction Subtraction dataset (a benchmark in Cognitive Diagnostic Modeling) – the algorithm learned a simpler model (with fewer hidden variables) without sacrificing interpretability.

Our algorithm enables practical applications, such as generating data-driven BN2A models for educational diagnostics—especially in settings where no prior expert-defined model exists. Depending on accuracy requirements, the learned model can directly analyze skill profiles, or serve as an initial framework for test designers to refine before deployment. As a next step, we plan to apply this method to a large-scale dataset of mathematics test results provided by the Czech Republic's Ministry of Education, Youth, and Sports.

	Skill							
	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8
Y_1	0	0	0	1	0	1	1	0
Y_2	0	0	0	1	0	0	1	0
Y_3	0	0	0	1	0	0	1	0
Y_4	0	1	1	0	1	0	1	0
Y_5	0	1	0	1	0	0	1	1
Y_6	0	0	0	0	0	0	1	0
Y_7	1	1	0	0	0	0	1	0
Y_8	0	0	0	0	0	0	1	0
Y_9	0	1	0	0	0	0	0	0
Y_{10}	0	1	0	0	1	0	1	1
Y_{11}	0	1	0	0	1	0	1	0
Y_{12}	0	0	0	0	0	0	1	1
Y_{13}	0	1	0	1	1	0	1	0
Y_{14}	0	1	0	0	0	0	1	0
Y_{15}	1	0	0	0	0	0	1	0
Y_{16}	0	1	0	0	0	0	1	0
Y_{17}	0	1	0	0	1	0	1	0
Y_{18}	0	1	0	0	1	1	1	0
Y_{19}	1	1	1	0	1	0	1	0
Y_{20}	0	1	1	0	1	0	1	0

	Skill				
	X_1	X_2	X_3	X_4	X_5
Y_1	0	0	1	1	1
Y_2	0	0	0	1	1
Y_3	0	0	0	1	1
Y_4	0	1	0	1	1
Y_5	1	1	1	1	1
Y_6	0	0	0	1	0
Y_7	1	1	0	1	0
Y_8	0	1	0	1	1
Y_9	1	0	1	1	0
Y_{10}	0	1	0	1	1
Y_{11}	0	1	0	1	1
Y_{12}	0	1	0	1	0
Y_{13}	0	1	0	1	1
Y_{14}	0	1	0	1	0
Y_{15}	1	1	0	1	1
Y_{16}	0	1	1	1	1
Y_{17}	1	1	0	1	0
Y_{18}	0	1	0	0	1
Y_{19}	1	1	0	1	1
Y_{20}	0	1	0	1	1

Table 5: Q-matrix Structures: Original Expert-defined (Tatsuoka, left) and Data-driven Learned ($K = 5$, right).

ℓ	$q_{\ell,0}$	$q_{\ell,1}$	$q_{\ell,2}$	$q_{\ell,3}$	$q_{\ell,4}$	$q_{\ell,5}$
1	0.919	-	-	0.819	0.020	0.076
2	0.969	-	-	-	0.031	0.031
3	0.885	-	-	-	0.023	0.005
4	0.903	-	0.254	-	0.735	0.699
5	1.000	0.799	0.838	0.555	0.418	0.499
6	0.953	-	-	-	0.149	-
7	0.974	0.316	0.310	-	0.001	-
8	0.952	-	0.758	-	0.666	0.795
9	0.878	0.674	-	0.843	0.411	-
10	0.813	-	0.091	-	0.043	0.532
11	0.941	-	0.108	-	0.144	0.874
12	0.952	-	0.808	-	0.154	-
13	0.684	-	0.100	-	0.011	0.128
14	0.963	-	0.823	-	0.029	-
15	0.977	0.572	0.427	-	0.053	0.453
16	0.992	-	0.818	0.767	0.084	0.921
17	0.948	0.812	0.067	-	0.052	-
18	0.813	-	0.214	-	-	0.598
19	0.928	0.363	0.088	-	0.005	0.635
20	0.837	-	0.014	-	0.033	0.880

Table 6: Learned parameters of the BN2A model with $K = 5$.

References

R. G. Almond, R. J. Mislevy, L. Steinberg, D. Yan, and D. Williamson. *Bayesian Networks in Educational Assessment*. Springer Publishing Company, Incorporated, 2015. ISBN 149392124X. doi: https://doi.org/10.1007/978-1-4939-2125-6.

S. A. Culpepper. Estimating the cognitive diagnosis Q matrix with expert knowledge: Application to the fraction-subtraction dataset. *Psychometrika*, 84(2):333–357, June 2019. doi: https://doi.org/10.1007/s11336-018-9643-8.

J. de la Torre. The generalized DINA model framework. *Psychometrika*, 76:179–199, 2011. doi: https://doi.org/10.1007/s11336-011-9207-7.

J. de la Torre and J. A. Douglas. Higher-order latent trait models for cognitive diagnosis. *Psychometrika*, 69(3):333–353, 2004. doi: https://doi.org/10.1007/BF02295640.

L. T. DeCarlo. On the analysis of fraction subtraction data: The dina model, classification, latent class sizes, and the q-matrix. *Applied Psychological Measurement*, 35(1): 8–26, 2011. doi: https://doi.org/10.1177/0146621610377081.

N. Friedman. The Bayesian structural EM algorithm. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, pages 129—138, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc. ISBN 155860555X. URL `https://dl.acm.org/doi/10.5555/2074094.2074110`.

S. Hartz and L. Roussos. The fusion model for skills diagnosis: Blending theory with practicality. *ETS Research Report Series*, 2008(2):i–57, 2008. URL `https://doi.org/10.1002/j.2333-8504.2008.tb02157.x`.

W. Ma and J. de la Torre. GDINA: An R package for cognitive diagnosis modeling. *Journal of Statistical Software*, 93(14):1–26, 2020. doi: https://doi.org/10.18637/jss.v093.i14.

C. Tatsuoka. Data analytic methods for latent partially ordered classification models. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 51(3):337–350, 07 2002. ISSN 0035-9254. doi: https://doi.org/10.1111/1467-9876.00272.

K. K. Tatsuoka and M. M. Tatsuoka. Bug distribution and statistical pattern classification. *Psychometrika*, 52(2):193–206, 1987. doi: https://doi.org/10.1007/BF02294234.

J. Vomlel. Building adaptive tests using Bayesian networks. *Kybernetika*, 40(3):333–348, 2004. URL `https://www.kybernetika.cz/content/2004/3/333`.

J. Vomlel, V. Kratochvíl, and F. Kratochvíl. Structural learning of mixed noisy-OR Bayesian networks. *International Journal of Approximate Reasoning*, 161:108990, 2023. doi: https://doi.org/10.1016/j.ijar.2023.108990.