

Using Mutual Information for Adaptive Item Comparison and Student Assessment

Chao-Lin Liu

Department of Computer Science
National Chengchi University
Wen-Shan, Taipei 11605, Taiwan
chaolin@nccu.edu.tw

ABSTRACT

The author analyzes properties of mutual information between dichotomous concepts and test items. The properties generalize some common intuitions about item comparison, and provide principled foundations for designing item-selection heuristics for student assessment in computer-assisted educational systems. The proposed item-selection strategies along with some common and conceivable methods, including mutual information-based methods and Euclidean and Mahalanobis distance-based methods, for student classification are evaluated in a simulation-based environment. The simulator relies on Bayesian networks for capturing the uncertainty in students' responses to test items. Simulated results indicate that the heuristics built upon the theoretical properties offer satisfactory performance profiles for item selection, and, not surprisingly, mutual information-based methods offer better performance for the task of student classification than distance-based methods.

Keywords

Educational assessments, Item selection, Intelligent tutoring, Mutual information, Bayesian networks, Mahalanobis distance, Classification, Adaptive interfaces, Uncertain reasoning

Introduction

Testing is the dominant way for assessing students' knowledge levels. Possible applications of the assessment include, but are not limited to, assigning scores to the students and diagnosing students' incompetence in some concepts (Yan et al., 2003). Given an item bank, test administrators face the challenge of selecting the proper subset of items which will facilitate the revealing of students' internal competence levels. In the recent decade, practitioners and researchers consider adaptivity as an additional important factor in item selection. In an interactive environment, adaptive item selection offers the chance of achieving the assessment goals with relatively shorter test length (Welch and Frick, 1993). Taking these two subgoals together, a good item selection strategy should attempt to select items from the item bank so that we can assess students both effectively and efficiently.

A main challenge for making good selection of items come from the uncertainty that item-response patterns may not reflect students' competence patterns perfectly. In the ideal case, students always respond correctly to the items for the concepts that the students already understand and can apply, and always respond incorrectly to items otherwise. In such an ideal world, there will be few difficulties, if any, in diagnosing students' deficiency by their item-response patterns. In the real world, students' item-response patterns are "fuzzy" (Birenbaum et al., 1994) because students may slip (responding incorrectly to items that they are supposed to respond correctly) and guess (responding correctly to items that the students do not have necessary knowledge).

The research community has admitted that uncertainty is a common challenge in many educational applications, and has proposed probability-based methods to cope with the problem. Researchers employ Bayesian networks (Pearl, 1988) for inferring students' actions in an interactive environment (Conati et al., 2002) and for modeling students' competence in concepts about arithmetic (Mislevy and Gitomer, 1996) and physics (Vanlehn and Martin, 1997). Despite the consensus on the applicability of probability theories to educational applications, researchers may apply probabilistic information in different ways. For instance, Collins et al. (2002) and Millán et al. (2000) investigate applications of adaptive item selection with the help of Bayesian networks, but they do not agree on the formula that they use to compare test items. The disagreement can lead to different selected subsets of test items for the assessment task, and results in different system efficiency. From the author's standpoint, this agreement was a result of relying on intuition-based heuristics, and information theory offers a chance to find a more acceptable common ground for the research community.

In this paper, we concern ourselves with a latent class analysis problem in which we observe students' item-response patterns for classifying the students into a limited number of groups (Dayton, 1991). We compute mutual information (Cover and Thomas, 1991) with the help of Bayesian networks, for adaptively selecting test

items that are more likely to reveal students' mastery of concepts and students' groups in a simulated environment. Experimental results indicate that guiding the item selection process with mutual information-based measures offers relatively better performance in classifying students into their unobservable types than guiding the selection with distance-based measures. We also investigate theoretical properties of mutual information. These properties shed light on the nature of item comparison, and offer a good basis for designing heuristics for item selection when computing exact mutual information is considered computationally costly. Experimental results show that the mutual information-based heuristics, designed based on the theoretical properties, provide satisfactory performance in item selection and student assessment.

We employ a Bayesian network-based simulation environment in evaluating the effectiveness of different approaches for item selection and student classification. Using simulated students in intelligent tutoring systems is not new to the research community. For instance, VanLehn et al. (1994) apply simulated students to helping people to adjust their teaching and learning strategies where the models are constructed based on some reasonable cognitive analyses (cf. Mislevy et al., 1998), and Beck (2002) employs simulated students for locating and improving poorly performing components in his system. VanLehn refers to the simulated students as **simulees**, and we will continue to use this term. Although the simulees may not mimic human behavior closely, it will be clear shortly that the Bayesian network-based models offer a convenient infrastructure for capturing the fuzziness in students' responses to test items and the dependent relationships among the test items. One advantage of this simulation-based evaluation is that it is easy to generate thousands of simulated students for the evaluation task for this theoretical study, though we have to take the simulated results with grain of salt.

This paper compiles and extends related material partially presented in three conference papers (Liu, 2004; Liu et al., 2004; Liu, in press), and reports experimental results of broader coverage. In the following section, we formulate our applications with Bayesian networks and elaborate on the applications of mutual information to adaptive item selection. Useful theorems and corollaries of mutual information will be presented and discussed, and we will apply the theorems and the corollaries for designing heuristics for item selection. Next, we look into the Bayesian network-based simulation environment that we employ for generating students' data. The simulated data will be used in evaluating different approaches for item selection and student classification. Finally, we examine and discuss the simulation results before concluding the paper with a brief discussion.

Adaptive Student Assessments

Consider the domain in which students should learn a set of n concepts $\mathcal{C} = \{C_1, C_2, \dots, C_n\}$. Some of the concepts in \mathcal{C} are **basic concepts**, and others are **composite** ones that are integrated from the basic concepts. For easier identification, we use cX and dY to denote the basic and the composite concepts, respectively, where Y signifies the components that comprise the composite concept. For instance, dAB is integrated from cA and cB . We also assume that, for each concept C_j , there is a set of $m(j)$ test items for evaluating students' competence in C_j , and denote this set of items by $\mathcal{J}_j = \{I_{j,1}, I_{j,2}, \dots, I_{j,m(j)}\}$. For easier reference, we refer to the basic concepts of the composite concepts as the **parent concepts** of the composite concepts. We also refer to C_j as the parent concept of items in \mathcal{J}_j .

We classify students according to whether students are competent in concepts in \mathcal{C} , so there are at most 2^n **competence patterns**. However, we assume that there are a limited number of competence patterns that the students really exhibit, and denote the set of these s types of students by $\mathcal{G} = \{g_1, g_2, \dots, g_s\}$.

We employ the *Q-matrix* that (Tatsuoka, 1983) originally used to encode the relationships between items and concepts for representing the relationship between student types and their competence patterns in \mathcal{C} . Let $q_{g,c}$ be a cell in the *Q-matrix*. If c represents a basic concept, then $q_{g,c}=1$ signifies that the g -th type of students are competent in c . If c represents a composite concept, then $q_{g,c}=1$ signifies that the g -th type of students are competent in integrating basic concepts for c . Note particularly that, when c represents a composite concept, $q_{g,c}=1$ is not a sufficient condition for the g -th type of students to be competent in c . In principle, it is possible that students might have the potential to integrate the ingredient concepts, while they do not have sufficient knowledge in the ingredient concepts. Note also that, although we use 1 or 0 in the matrix, our simulator embraces a randomization mechanism to make the relationships between student groups and competence patterns a bit uncertain, which will become clear later in this paper, i.e., the subsection on *Generating the Simulees*. Table 1 contains a sample *Q-matrix* where we assume only 9 types of students.

student types	cA	cB	cC	dAB	dBC	dAC	dABC
1	1	1	1	1	1	1	1
2	1	1	1	1	1	0	0
3	1	1	1	0	0	1	1
4	1	1	0	1	0	0	0
5	0	1	1	0	1	0	0
6	1	0	1	0	0	1	0
7	1	1	1	0	0	0	0
8	1	1	0	0	0	0	0
9	0	0	1	0	0	0	0

Table 1. A sample Q -matrix

Formulation with Bayesian Networks

In the past decade or so, Bayesian networks (Pearl, 1988; Jensen, 2001) have become an important formalism for representing and reasoning about uncertainty, using probability theories as their substrate. Researchers of educational assessment have also studied the applications of Bayesian networks in education (e.g., Conati et al., 2002; Mislevy et al., 1999).

A Bayesian network is a directed acyclic graph, consisting of a set of nodes and directed arcs. The nodes represent random variables, and each node can take on a set of possible values. The arcs signify direct dependence between the connected nodes in the applications qualitatively. A node at the terminal with an arrow of the arc is a *child* node of the *parent* node that is located at the terminal without an arrow. In addition to the graphical structure, associated with each node in the network is a conditional probabilistic table (CPT) that specifies the probabilistic relationship between values of the child and the parent nodes. Roughly speaking, the contents of CPTs quantitatively specify the strength between the directly dependent random variables that are connected by the arcs. By construction, the contents of the CPTs of all nodes in the network indirectly and economically encode the joint distribution of all variables in the network. As a result, we can compute any desired probabilistic information with a given Bayesian network.

Let C be the random variable that encodes the degree of the mastery of the concept, and X be the random variable representing the outcomes of using an item for testing the mastery of C . In this paper, we assume that variables for both concepts and items are dichotomous. A variable for the mastery of a concept takes the value of either **good** or **bad**, and a variable for the response to an item takes the value of either **correct** or **incorrect**. For simplicity of notation, we use a small letter of the variable to denote the “positive” value of the random variable, and a small letter with a bar to denote the “negative” value of the variable. For instances, $\Pr(x|c)$ denotes $\Pr(X = \text{correct} | C = \text{good})$, and $\Pr(\bar{x}|\bar{c})$ $\Pr(X = \text{incorrect} | C = \text{bad})$. We use the special symbol PR and capital names of random variables to denote the probability values of all possible combinations of the values of the involved random variables. For instance, $PR(X|C)$ denotes $\{\Pr(x|c), \Pr(x|\bar{c}), \Pr(\bar{x}|c), \Pr(\bar{x}|\bar{c})\}$. Similarly, we use simplified notation for the conditional probability of a composite concept, whose state depends on its parent concepts. For instance, we use $\Pr(dab|\bar{c}a)$ for $\Pr(dAB = \text{good} | cA = \text{bad})$.

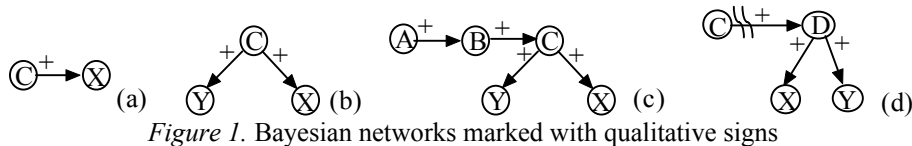


Figure 1. Bayesian networks marked with qualitative signs

We can use the very simple Bayesian network shown in Figure 1(a) to represent that C is the parent concept of a test item X . In practice, we have no reason to assume that the probability of answering X correctly would decrease when a particular student gets a hand on C . Therefore, in Figure 1(a), we also have $\Pr(x|c) \geq \Pr(x|\bar{c})$, and we show this **positive influence** of C on X by marking the link between them with a “+” symbol, following the tradition of Qualitative Probabilistic Networks (QPNs) (Wellman, 1990). (In a fully-fledged QPN, random variables may have relationships of **negatively influence**, denoted by “-”, and **ambiguity**, denoted by “?”. One marks the relationship between a concept and an item by “-”, when understanding the concept hinders a student from answering the item correctly.) We can use the network shown in Figure 1(b) when we have two items available for testing the competence in C . Notice that, when we accept Figure 1(b), we assume that the student’s responses to X and Y become independent given the information about the student’s

mastery of C . When we believe that mastering a parent concept, e.g., B in Figure 1(c), helps the mastery of C , we can add a node for B and draw a link with a plus sign from B to C as well. According to the inference rules for QPNs, we can infer that mastering concept A in Figure 1(c) indirectly improves the mastery of C , and further increases the chances of responding to X correctly. In written form, we use $S^+(A,C)$ and $S^+(A,X)$ to denote the positive influences of A on C and X , respectively. (We will discuss matters about Figure 1(d) in a later section.)

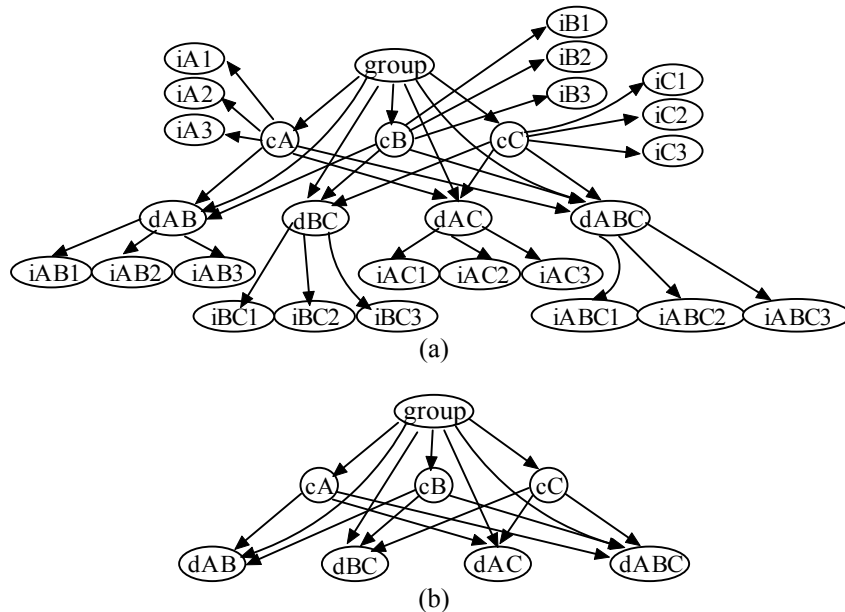


Figure 2. A Bayesian network for encoding data in Table 1

Figure 2(a) shows a possible Bayesian network for a realizing the Q -matrix in Table 1. The *group* node represents the types of students. Nodes whose names start with 'i' represent correctness of students' responses to test items, and can take either *correct* or *incorrect* as their values. The other nodes are also dichotomous, each representing whether or not a student understands the concept that is denoted by the names of the nodes.

The arcs connecting the related basic and composite concepts, e.g., those between cA , cB , and dAB , suggest that the competence in the parent concepts directly influences the competence in the composite concepts. The arcs connecting the *group* node and cX nodes capture the assumption that different student groups show different competence in cX , while the arcs connecting the *group* node and dY nodes capture the assumption that different student groups have different ability in integrating the basic concepts for a dY . We could have put a "+" sign on the links from basic to composite concepts, as increasing the mastery of basic concepts increases the chance of achieving better mastery of composite concepts. Figure 2 does not include these signs for qualitative relationships for readability of the figure. Figure 2(b) and Figure 5 that will come up later do not include nodes for test items, because depicting nodes for all items makes the picture less readable as Figure 2(a) has proved. If $m(j)=3$ for all C_j in \mathcal{C} , we will have to add three nodes for each concept, and add links from the parent concepts to their test items.

Note that, in our formulation, the responses to test items are *not* independent given the student's group identity, as many systems that rely on the item-response theory (IRT) (Hambleton, 1991) may have assumed. Similar to the discussion for Figure 1(b), we also assume that the responses to items in \mathcal{J}_j are independent, given the mastery of the parent concept C_j . However, using the network in Figure 2(a) as an example, the responses to items designed directly for $dABC$ and dAB remain *dependent* given the mastery of cB and the student's subgroup. The mastery of cA makes the responses to items for $dABC$ and dAB remain dependent. Hence the Bayesian network-based models are more general than the IRT-based models.

Given the network structure, we still need to provide a CPT for each node. Figure 2 does not show the CPTs of the network, but more details about the CPTs will be provided later in this paper. Similar to how people fit IRT models to collected test data, we can use statistical methods to estimate these parameters, e.g., (Mislevy, 1999). Once the numerical information becomes available, the network is ready to serve our applications.

Mutual Information-Bases Assessments

For the extremely simple case shown in Figure 1(b), it would be helpful if we have a principled way for determining whether we should administer X or Y for assessing the subject student's competence in C . We can compute the mutual information $MI(X;C)$ (Cover and Thomas, 1991) between C and X for item comparison with a Bayesian network.

$$MI(X;C) = \sum_{c \in \text{domain}(C)} \sum_{x \in \text{domain}(X)} \Pr(c, x) \log \frac{\Pr(c, x)}{\Pr(c)\Pr(x)}$$

Since $MI(X;C) = H(C) - H(C|X)$, where $H(C)$ and $H(C|X)$, respectively, denote the entropy of C and the conditional entropy of C given X , $MI(X;C) \geq MI(Y;C)$ implies that $H(C|X) \leq H(C|Y)$. Hence we should prefer an item that has larger mutual information with C because the information about such an item allows less uncertainty about C . (Note that, although $MI(X;C) = H(C) - H(C|X)$, we do not need to compute $H(C)$ and $H(C|X)$ separately for obtaining the mutual information between C and X . We can compute the mutual information with the definition directly. The purpose of discussing the relationship $MI(X;C) = H(C) - H(C|X)$ is simply to explicate the usefulness of mutual information.)

Based on this observation, we use the following procedure for classifying students by their item-response patterns. Given a Bayesian network for the assessment task, this procedure iteratively selects and administers the test item that has the largest conditional mutual information with *group*. Step 2 updates the distribution over *group* based on the student's responses, each of which must be either *correct* or *incorrect*. The most probable subgroup is considered to be the student's subgroup

MI-ADAPT: Procedure for adaptive student assessment

1. Select and administer the item that has the largest mutual information with *group*
2. Select the most probable subgroup in *group* as the student's subgroup, based on the posterior probability distribution over *group*, updated for the results of administering the selected items
3. Stop the classification task, if every item has been administered; otherwise continue
4. Compute the mutual information between each available item and *group*, given the results of administering previous items
5. Select and administer the item that has the largest condition mutual information with *group*, and return to step 2

The records collected at step 2 allow us to inspect the transient performance of this adaptive procedure. At step 3, the simulation will not stop until it uses up all the available test items for each examinee. This is certainly not to occur in a realistic assessment. We choose to do so because we would like to observe the performance profiles as much as possible in experiments.

Heuristics for Item Selection

Although we achieved high accuracy of classification in (Liu, 2004), the computational costs of step 2 in MI-ADAPT remain a concern. The computation of a particular mutual information between a pair of random variables may need just one propagation in the Bayesian network, but this "one propagation" can be quite costly as computing either exact or approximate probabilities in Bayesian networks is NP-hard (Cooper, 1990; Dagum and Luby, 1993). The problem will be exacerbated when we need to compute the mutual information between each test item and the random variable of interest. In Figure 2, we have to compute the mutual information between each untested item with *group*, and there may be hundreds or thousands of test items available in a realistic test-item database. We investigate theoretical properties of mutual information that shed light on the nature of item comparison and help us to design heuristics for item selection, and explore some distance-based heuristics in this section.

Useful Properties of Mutual Information for Item Comparisons

Our theorem and corollaries originate from Theorem 1. Notice that, except in Theorem 1, we assume all variables are dichotomous. Although we do not have to restrict the interpretation of variables C and X while deriving the mathematical relationships, it will be easier to understand the whole procedure by considering C and X , respectively, as the competence of a parent concept of a test item and the correctness of a response to the test item.

Theorem 1 (Cover and Thomas, 1991) Let $\Pr(c, x) = \Pr(x | c)\Pr(c)$ be the joint distribution of C and X . The mutual information $MI(X; C)$ is a concave function of $PR(C)$ for fixed $PR(X | C)$ and a convex function of $PR(X | C)$ for fixed $PR(C)$.

Lemma 1 $\Pr(x | c) = \Pr(x | \bar{c}) \Rightarrow MI(X; C) = 0$ (because of independence between C and X)

Theorem 2 For a fixed $\Pr(c)$, when $\Pr(x | c) \geq \Pr(x | \bar{c})$, $MI(X; C)$ is a monotonically increasing function of $\Pr(x | c)$ for a fixed $\Pr(x | \bar{c})$, and a monotonically decreasing function of $\Pr(x | \bar{c})$ for a fixed $\Pr(x | c)$.

Proof. Consider the space of $\Pr(x | c)$ and $\Pr(x | \bar{c})$ shown in Figure 3. Each point in the space represents a pair of $\Pr(x | c)$ and $\Pr(x | \bar{c})$ for a particular distribution $PR(X | C)$. The square contains all possible combinations of $\Pr(x | c)$ and $\Pr(x | \bar{c})$, and the diagonal line segment represents the situations when $\Pr(x | c) = \Pr(x | \bar{c})$.

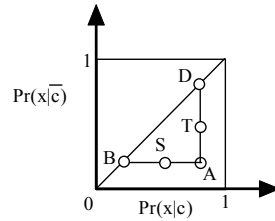


Figure 3. The space for $PR(X | C)$ represented by $(\Pr(x | c), \Pr(x | \bar{c}))$

Let $PR_a(X | C)$, $PR_b(X | C)$, $PR_d(X | C)$, $PR_s(X | C)$, and $PR_t(X | C)$, respectively, denote the probability distributions represented by A , B , D , S , and T in Figure 3. Assume that B , S , and A are on a horizontal line segment, and that D , T , and A are on a vertical line segment. The coordinates of S must be a linear combination of the coordinates of the terminals of the line segment where S resides, and this geometric fact applies to T analogously. As a result, we can express $PR_a(X | C)$, $PR_b(X | C)$, $PR_d(X | C)$, $PR_s(X | C)$, and $PR_t(X | C)$ in the following manner, where $n \leq m$ and $0 \leq \gamma, \delta \leq 1$.

$$PR_a(X | C) = (m, n); \quad PR_b(X | C) = (n, n); \quad PR_d(X | C) = (m, m);$$

$$PR_s(X | C) = \gamma PR_a(X | C) + (1 - \gamma) PR_b(X | C) \quad (1)$$

$$PR_t(X | C) = \delta PR_a(X | C) + (1 - \delta) PR_d(X | C) \quad (2)$$

Let $MI_a(X; C)$, $MI_b(X; C)$, $MI_d(X; C)$, $MI_s(X; C)$, and $MI_t(X; C)$, be the mutual information $MI(X; C)$ when $PR(X | C)$ takes on the distribution represented by A , B , D , S , and T , respectively. Applying Lemma 1, $MI_b(X; C)$ and $MI_d(X; C)$ must be zero. In addition, because $PR_s(X | C)$ is a linear combination of $PR_a(X | C)$ and $PR_b(X | C)$ in (1), the following inequality must hold according to Theorem 1.

$$\begin{aligned} MI_s(X; C) \leq \gamma MI_a(X; C) + (1 - \gamma) MI_b(X; C) &\Rightarrow MI_s(X; C) \leq \gamma MI_a(X; C) \quad \because MI_b(X; C) = 0 \\ &\Rightarrow MI_a(X; C) \geq MI_s(X; C) \quad \because 0 \leq \gamma \leq 1 \end{aligned}$$

In Figure 3, the only difference between A and S is that the $\Pr(x | c)$ of A is larger than that of S . Hence we have shown that, when $\Pr(x | c) \geq \Pr(x | \bar{c})$, $MI(X; C)$ is a monotonically increasing function of $\Pr(x | c)$ for fixed $\Pr(c)$ and $\Pr(x | \bar{c})$.

Analogously, the following inequality must hold according to Theorem 1, because $PR_t(X|C)$ is a linear combination of $PR_a(X|C)$ and $PR_c(X|C)$ in (2).

$$\begin{aligned} MI_t(X;C) \leq \delta MI_a(X;C) + (1-\delta)MI_d(X;C) &\Rightarrow MI_t(X;C) \leq \delta MI_a(X;C) \quad \because MI_d(X;C) = 0 \\ &\Rightarrow MI_a(X;C) \geq MI_t(X;C) \quad \because 0 \leq \delta \leq 1 \end{aligned}$$

In Figure 3, the only difference between A and T is that the $\Pr(x|\bar{c})$ of A is smaller than that of T . Hence we have shown that, when $\Pr(x|c) \geq \Pr(x|\bar{c})$, $MI(X;C)$ is a monotonically decreasing function of $\Pr(x|\bar{c})$ for fixed $\Pr(c)$ and $\Pr(x|c)$.

Theorem 2 provides a basis for preferring one test item against others without having to actually compute the mutual information. The following corollary of the theorem allows us to compare two items by examining their associated CPTs in Bayesian networks, and is applicable for determining when and explaining why we should prefer X to Y in Figure 1(b).

Corollary 1 Let C be the parent concept of items X and Y . We have $MI(X;C) \geq MI(Y;C)$ if $\Pr(x|c) \geq \Pr(y|c) \geq \Pr(y|\bar{c}) \geq \Pr(x|\bar{c})$.

Proof. This corollary results directly from Theorem 2.

As an extreme case, when $\Pr(x|c) = 1$ and $\Pr(x|\bar{c}) = 0$, the item X will have the largest mutual information with C , and is the top choice for testing students' competence in C . On the other hand, when $\Pr(x|c) = \Pr(x|\bar{c})$, no item offers less amount of information with C than X , so X is the worst item to administer. Corollary 1 dictates that the distribution $PR_a(X|C)$, represented by A in Figure 3, offers the largest $MI(X;C)$ among all points within the triangle ΔABD . Hence, Corollary 1 generalizes intuitions for item comparison. Nevertheless, Corollary 1 does not allow us to obtain a total ordering of the mutual information between the test items and any given concept. Corollary 1 does not guarantee specific relationships between A and other points outside ΔABD . Our experiments show that the mutual information offered by other points outside of the triangle can have any possible relationship with that offered by A , depending on the numerical peculiarities.

Figure 1(d) shows an additional scenario when Theorem 2 applies. The tilted short curves represent that C and D do not have to have a direct relationship. In this figure, D is the parent concept of two dichotomous items, X and Y , and there is a concept C that positively influences D . The following corollary shows when an item is better than the other for assessing the mastery of a related concept. Corollary 2 holds as long as C positively influences D , i.e., $S^+(C,D)$, as is defined in QPNs.

Corollary 2 We have $MI(X;C) \geq MI(Y;C)$ if $\Pr(x|d) \geq \Pr(y|d) \geq \Pr(y|\bar{d}) \geq \Pr(x|\bar{d})$ and $\Pr(d|c) \geq \tau \geq \Pr(d|\bar{c})$, where

$$\tau = \frac{\Pr(y|\bar{d}) - \Pr(x|\bar{d})}{(\Pr(x|d) - \Pr(y|d) + (\Pr(y|\bar{d}) - \Pr(x|\bar{d})))}$$

Proof. This corollary extends Corollary 1. The proof involves some algebraic manipulations of the probabilistic terms.

Corollary 2 dictates that even if X is more related to D than Y is *does not* imply that X is more related to C than Y is. This is against what one might have intuitively thought. In realistic assessment, test administrators need to watch whether $\Pr(d|c) \geq \tau \geq \Pr(d|\bar{c})$ really holds in the tests to make sound inference about students' competence based on their item responses.

Mutual Information-Based Heuristics

In previous work, researchers choose some probability-based heuristics for selecting items with Bayesian networks that have subtly different structures than ours. Collins et al. (1996) use $|\Pr(c|x) - \Pr(\bar{c}|\bar{x})|$, and Millán et al. (2000) argue for $(\Pr(c|x) - \Pr(c))\Pr(x) - (\Pr(\bar{c}|\bar{x}) - \Pr(\bar{c}))\Pr(\bar{x})$. Using different criteria for test item selection will lead to different test procedures for students, and have a great impact on the effectiveness of adaptive tests.

Theorem 2 and its corollaries provide the support for a different heuristic. Given two items, X and Y , and their parent concept C , an imprecise interpretation of Corollary 1 suggests that X has more mutual information with C than Y does, if $\Pr(x|c) - \Pr(x|\bar{c}) \geq \Pr(y|c) - \Pr(y|\bar{c})$. This interpretation is problematic because $\Pr(x|c) - \Pr(x|\bar{c}) \geq \Pr(y|c) - \Pr(y|\bar{c})$ is a necessary condition of, but not a sufficient condition of, $\Pr(x|c) \geq \Pr(y|c) \geq \Pr(y|\bar{c}) \geq \Pr(x|\bar{c})$. Corollary 2 further states that items that have larger mutual information with their parent concepts may have larger mutual information with a concept that is related to their parent concepts, when $\Pr(d|c) \geq \tau \geq \Pr(d|\bar{c})$ holds. Putting these together, an item X with larger $\Pr(x|c) - \Pr(x|\bar{c})$ might have larger mutual information with a concept that is remotely related with C , *under ideal circumstances*. The heuristic score of an item X , with C as its parent concept, is thus defined as follows.

$$s(X) = \Pr(x|c) - \Pr(x|\bar{c}) \quad (3)$$

When the ideal conditions do not hold, we may select a non-optimal item. The heuristic is also a static measure that does not change with the students' item responses on the fly as the conditional mutual information that we compute in MI-ADAPT would.

The previous heuristic helps us to pick the best item designed for a particular concept, but does not provide clues for selecting items of which concept that we should examine. At present, we rely on the "distance among concepts" to select the concept, and define a distance measure based on the information contained in the Q -matrix. Let $q_{j,k}$ denote the cell at the j -th row and the k -th column in the Q -matrix. Recall that $q_{j,k}$ represents the competence of typical students of type g_j in C_k . Assuming that there are s subgroups of students, the Euclidean distance between the vectors $(q_{1,h}, q_{2,h}, \dots, q_{s,h})$ and $(q_{1,k}, q_{2,k}, \dots, q_{s,k})$ can be used as an indication of how the concepts C_h and C_k can help us distinguish students of different subgroups. Hence we define the distance between C_h and C_k as (4).

$$d_1(C_h, C_k) = \left[\sum_{t=1}^s (q_{t,h} - q_{t,k})^2 \right]^{1/2} \quad (4)$$

Let $U \subseteq \mathcal{C}$ denote the set of parent concepts of the administered items. The distance between concepts C_m and a subset $U \subseteq \mathcal{C}$ is defined in (5). If $C_m \in U$, $d_2(C_m, U) = 0$. Based on the idea of content balancing (Leung et al., 2003), the item that is designed for a concept $C \notin U$ may help us to gather more unknown information than a $C' \in U$. Moreover, among all items for such untested concepts, we prefer the concept that has the largest $d_2(C, U)$ because such a C appears to be most dissimilar to concepts in U . Since there are more items than concepts in our experiments, we reset U to an empty set every time one item of each concept has been administered, whenever necessary.

$$d_2(C_m, U) = \sum_{C_t \in U} d_1(C_m, C_t), \quad C_m \notin U \quad (5)$$

Distance-Based Heuristics

For comparison purposes, we evaluate the possibility of classifying students without relying on Bayesian networks. We use distance-based measures for student classification, and prefer the subgroup that has the shortest distance between its standard competence pattern and the student's item-response pattern. Given a student's item-response pattern, we create a competence pattern $R = (rC_1, rC_2, \dots, rC_n)$, where rC_m is the ratio of the student's correct responses to administered items for C_m . In the extreme cases, rC_m will be, respectively,

1 and 0, if the student responds to all administered items for rC_m correctly and incorrectly. rC_m will be 0.5, if either no item for C_m is administered yet or the student responds correctly to half of the items for C_m .

Given a vector R , we compute the Euclidean distance between the stereotypical competence patterns of the student and each subgroup g_k of *group* as follows. (Recall that n is the number of different concepts when we defined $\mathcal{C}=\{C_1, C_2, \dots, C_n\}$, and that $(q_{k,1}, q_{k,2}, \dots, q_{k,n})$ represents the typical competence patterns of students of subgroup g_k .)

$$d_3(R, g_k) = \left[\sum_{t=1}^n (q_{k,t} - rC_t)^2 \right]^{1/2} \quad (6)$$

Although the application of Euclidean distance is quite intuitive and common among teachers, it seems to be rather lenient to compare the performance of a Euclidean distance-based measure in (6) with that of a probabilistic heuristic based on (3). A more challenging distance-based measure is the Mahalanobis distance (cf. Tatsuoka and Tatsuoka, 1987; Duda et al., 2001).

$$d_4(R, g_k) = [(R - \mu_k) \Sigma_k^{-1} (R - \mu_k)^T]^{1/2} \quad (7)$$

In (7), μ_k and Σ_k are, respectively, the mean and the variance-covariance matrix of the competence patterns of students in subgroup g_k , and $(R - \mu_k)^T$ is the transpose of the row vector $(R - \mu_k)$. Recall that $\{C_1, C_2, \dots, C_n\}$ is the set of concepts of interest, so μ_k and Σ_k are, respectively, an $n \times 1$ row vector and an $n \times n$ matrix. The advantages of the Mahalanobis distance come at some extra costs, and our system has to learn statistics about μ_k and Σ_k from student' test records with standard statistical methods.

Note that it is possible that R has the same distances to the competence patterns of multiple subgroups, no matter what distance measures that we might use. Since *distance* is the only measure for this approach, we have no extra basis to prefer one subgroup to another. Hence, each such a subgroup, when they exist, will be considered equally likely.

A Bayesian Network-Based Simulation Environment

Figure 4 shows major components of the simulation environment. Simulation administrators need to provide a command file that describes the *simulation scenario*. Given the command file, the simulator generates a Bayesian network that models the learning domain, and uses this network to create simulees for further applications. In current simulations, the concept nodes are dichotomous, meaning that we assume that a student is either competent or not competent in a concept. Similarly, we assume that the item nodes are dichotomous, meaning that each student responds to items either correctly or incorrectly. The final output of the simulator is a list of records of simulees' item response patterns along with their groups.

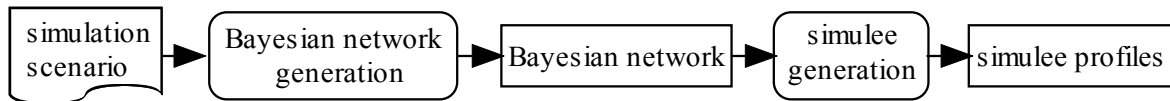


Figure 4. Major steps for creating simulees in the simulations

Generating the Bayesian Networks

This following BNF grammar summarizes how we describe the setups for simulations in the command files. The semantics of the grammar will become clear in the following elaboration.

- The BNF grammar for our simulations
- $\langle \text{sim} \rangle \rightarrow \langle \text{pgroup} \rangle \langle \text{concept} \rangle^+ \langle \text{pitem} \rangle^* \langle \text{params} \rangle$
- $\langle \text{pgroup} \rangle \rightarrow \text{group-name number-of-group} \langle \text{subgroup} \rangle^+$
- $\langle \text{subgroup} \rangle \rightarrow \text{subgroup-name subgroup-probability}$
- $\langle \text{concept} \rangle \rightarrow \langle \text{bconcept-type} \rangle \mid \langle \text{dconcept-type} \rangle$
- $\langle \text{bconcept-type} \rangle \rightarrow \text{bconcept concept-name}$

- <dconcept-type> → dconcept concept-name number-of-parents <parent-concept-name>+
- <pitem> → item item-name parent-concept-name
- <params> → Q-matrix <p1> <p2> <p3> <p4>
- <p1> → guess value-of-guess
- <p2> → slip value-of-slip
- <p3> → gError1 value-of-gError1
- <p4> → gError2 value-of-gError2

As we described earlier, major ingredients of the problems that we plan to explore include the set of student types \mathcal{G} , the set of concepts \mathcal{C} , and the set of test items \mathcal{I}_j for each C_j in \mathcal{C} , and the *Q-matrix*. The non-terminal nodes <pgroup>, <concept>, <pitem>, and <params> and their further derivations, respectively, specify details about student groups, concepts, test items, and system parameters for the simulation. In addition, we need to provide more details before the simulation can better mimic the uncertainty in the real world using a Bayesian network similar to those shown in Figure 2.

The current simulator allows us to specify the distribution over the student groups. Since the group node is a discrete and probabilistic, simulation administrators need to specify the prior probability of each student type, i.e., $\Pr(\text{group}=g_i)$ for all g_i in \mathcal{G} . We can control the probability distributions over the student groups by manipulating values in the subgroup-probability field.

For convenience, we use only 1s and 0s in specifying the *Q-matrix* in Table 1, and take the risk of giving an illusion of our introducing deterministic relationships between the student types and their competence patterns. We compensate this by requiring the simulation administrators to specify two parameters in commands gError1 and gError2. These parameters control the probability of how students of each type will deviate from the stereotypical behaviors that are specified in the *Q-matrix*: gError1 controls the maximum degree a variable will deviate from a positive value, and gError2 controls the maximum degree a variable will deviate from a negative value. When $q_{g,c}=1$ for a student type g and a basic concept c , the conditional probability $\Pr(c|g)$ will be sampled uniformly from the range $[1 - \text{value-of-gError1}, 1]$. When $q_{g,c}=0$, $\Pr(\bar{c}|g)$ will be sampled uniformly from the range $[0, \text{value-of-gError2}]$. At this moment, we rely on the default random number generator rand() in Microsoft Visual C++ for the sampling task.

The task for creating the CPTs for the composite concepts is more complex. Recall that both types of students and competence in parent concepts of the composite concepts influence the competence in the composite concepts. Hence, if a dichotomous composite concept has k dichotomous parent concepts, the simulator must determine $s \times 2^k$ parameters for this composite concept, where s is the cardinality of \mathcal{G} . Although this is not impossible for a simulator to do so, doing so would be impractical and perhaps unnecessary. Take *dAB* for example. Using a logical way of thinking, a student must be competent in its parent concepts, and be able to integrate its parent concepts so that s/he can be competent in *dAB*. Namely, there are three main factors that simultaneously affect the student's competence in *dAB*. This is clearly an example of the "AND" concept in logics, and there is an extension of the "logic-AND" concept in Bayesian networks. We choose to employ a simplified version of "noisy-AND" nodes (Pearl, 1988) in Bayesian networks, and apply a probabilistic version of AND nodes for modeling how competences in basic concepts influence competences in composite concepts. Adopting noisy-AND nodes is not an uncommon practice for work that relies on Bayesian networks in student modeling (e.g., Conati et al., 2002).

Take the second student type in Table 1 for example. We need to obtain the influences from the basic concepts *cA* and *cB* to *dAB* for setting the values for $\Pr(\text{dab} | \overline{ca}, \overline{cb}, g_2)$. Because *cA* is positive, we sample the influence of *cA* uniformly from $[1 - \text{value-of-gError1}, 1]$, and because *cB* is negative, we sample the influence of *cB* uniformly from $[0, \text{value-of-gError2}]$. The influence of being a student in g_2 will be sampled uniformly from $[1 - \text{value-of-gError1}, 1]$ because a stereotypical student of g_2 is capable of integrating *cA* and *cB*. After obtaining these three random numbers, we set $\Pr(\text{dab} | \overline{ca}, \overline{cb}, g_2)$ to their product, and $\Pr(\overline{\text{dab}} | \overline{ca}, \overline{cb}, g_2)$ to $1 - \Pr(\text{dab} | \overline{ca}, \overline{cb}, g_2)$. We set the parameters for other parent configurations of *dAB* using an analogous method. Simulation administrators control the assignment of the CPTs for the item nodes by choosing values for *slip* and *guess*, which control the probabilities that students make slipping and guessing, respectively. For any concept *C* and any of its test items, we set $\Pr(i | c)$ to a number sampled uniformly from $[1 - \text{value-of-slip}, 1]$, and $\Pr(i | \bar{c})$ to a number sampled uniformly from $[0, \text{value-of-guess}]$. We then set $\Pr(\bar{i} | c)$ to $1 - \Pr(i | c)$ and $\Pr(\bar{i} | \bar{c})$ to $1 - \Pr(i | \bar{c})$.

Generating the Simulees

Once we create a Bayesian network according to the directions given in the command file, we are ready to create simulees using the generated Bayesian network. Figure 2(a) shows one of such generated Bayesian networks, and we can easily use it to simulate how simulees respond to test items in examinations.

We determine whether a simulee respond to a test item correctly or incorrectly with the help of random numbers. For a simulee that belongs to the g -th student type, we can calculate the conditional probability of answering a test item I correctly, $\Pr(i|g)$, with the Bayesian network. In our simulations, we assume that simulees always respond to test items, so the results of their responses must be categorized as either correct or incorrect. To this end, we sample a random number ρ uniformly from the range $[0,1]$ to determine whether a particular simulee responds to the item correctly or not. We record that the simulee answers an item I incorrectly if $\rho > \Pr(i|g)$ and correctly otherwise. In the current work, the correctness of response to each item is determined independently. More specifically, the responses to items I_u and I_v are determined by two independently sampled random numbers, $\Pr(i_u | g)$, and $\Pr(i_v | g)$.

We apply a similar procedure to assign a type to each simulee. Based on the probability provided in the command file, we let each student type occupy an interval in the range of $[0,1]$. We sample a random number ρ uniformly from $[0,1]$, and assign the simulee the student type whose interval includes ρ .

In the simulations, we create simulees one at a time, and record their types and their item response patterns in the output file. Further experiments are then conducted with the recorded data.

Simulation-Based Evaluation

We ran simulations for the Q -matrix listed in Table 1. We have discussed one possible way of realizing this scenario with the Bayesian network shown in Figure 2. For examining the effects of different numbers of groups, we removed the eighth and ninth subgroups from Table 1 in some of the experiments. In the current experiments, each concept was prepared three test items. Hence there were 21 test items for the network shown in Figures 2(a). For comparing effects of different network structures, we also tried another network that made $dABC$ a composite concept of cC and dAB . That was tentative to show the belief that students learn $dABC$ by integrating cC and dAB , rather than directly from the three basic concepts.

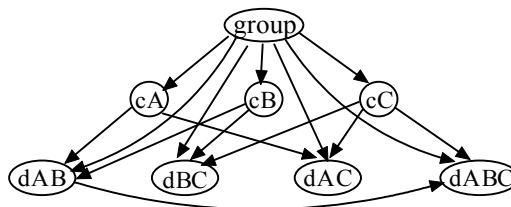


Figure 5. Another partial Bayesian network for encoding data in Table 1

We looked into how parameters of the simulation scenarios influenced performances of the evaluated classifiers and item-selection strategies. We examined the influences of *guess*, *slip*, *gError2*, and *gError1*. (For simplicity, we use *guess*, *slip*, *gError2*, and *gError1*, respectively, in places of *value-of-guess*, *value-of-slip*, *value-of-gError1*, and *value-of-gError2* henceforth.) These parameters affected how fuzzy the item-response patterns of the simulees can be. The number of student groups and the structures of the networks also affected the difficulties of the classification tasks for the classification mechanisms and item-selection strategies.

We used a total of 20000 simulees in each experiment. Each experiment consisted of 10 smaller-scaled experiments that involved 2000 simulees. Each of the smaller-scaled experiments used a particular simulation scenario, which is explained near Figure 4. In each of these smaller-scaled experiments, the CPTs of the Bayesian networks were re-sampled for the structure and parameters specified in the simulation scenario. We applied the Bayesian networks to create simulees. Half of these simulees, i.e., 1000 simulees, were used as the training data for learning parameters about the generated simulees, and the other half were used as the test data (Mitchell, 1997). We then applied the learned models, which can be either a Bayesian network or the parameters for computing the Mahalanobis distance, to select test items and classify simulees in the test data.

Learning Model Parameters

We did not use exactly the same Bayesian network that was used in creating simulees to classify simulees in the test data. We used the simulees in the training data to learn Bayesian networks for the classification task. When conducting the task of learning Bayesian network, the network structure that was used to generate the training data was provided to the learning procedure, but the conditional probability tables of the original network was not. The network structure and the training data were then used to learn the CPTs of the network that will be used in the classification task.

The learning procedure was proposed by Lauritzen (1995) and implemented in Hugin (<http://www.hugin.dk>). The `h_domain_learn_tables` function in Hugin applies an expectation-maximization approach for learning the CPTs of Bayesian network from the training data. The network that was used to create the simulees was used as the network given to `h_domain_learn_tables`. The experience counts for CPTs of this given network were set to 10, thereby reducing the influence of the initial settings on the CPTs to be learned and allowing the training data to dominate the results of the learning task. It is noted that the fact that the initial settings of the CPTs came directly from the simulator might give advantages to the learned Bayesian networks. Nevertheless, in practice, test administrators should have the capability to collect ample amount of students' data, and provide initial settings of good quality as well, so the way we assigned the initial values of the CPTs is not unreasonable. If we can collect students' data for a long period, the collected data should provide good hints about how the Bayesian networks should be initialized.

Similarly, in evaluating the classifiers that employed the Mahalanobis distance, we needed to collect statistics for calculating quantities given in (7). The statistics μ_k and Σ_k , for all k , were computed based on the training data using standard statistical methods.

Measurement for Quality of Classification

We used the average accuracy of the classification for measuring the system performance. When we used the original MI-ADAPT, simulees would be assigned to the group that has the largest conditional probability at step 2. When we used the distance-based heuristics given in both (6) and (7) for classification, simulees would be assigned to the group that was closest to the simulees' competence patterns R . If there were f subgroups that had the same, closest distance with the simulee, each of these f subgroups would get $1/f$ credit. The accuracy of classifying the j -th simulee when we administered k items, denoted $m_{j,k}$, was the credit that was assigned to the correct subgroup of the simulee when we administered k items. Let σ be the total number of simulees in the test data in an experiment. The average accuracy of an experiment is defined as follows.

$$accuracy(k) = \frac{1}{\sigma} \sum_{j=1}^{\sigma} m_{j,k}$$

We collected statistics about the performance of the classification after administering every test item. The results were then averaged over the smaller-scaled experiments, and used to plot the performance profile of a particular experiment.

Experimental Results

Although many factors influence the accuracy of the classification task, we can report on influences of these factors only in a limited number of experiments. We have set `guess` and `slip` to the same value, denoted α , and `gError2` and `gError1` to the same value, denoted β .

We employ tags for identifying experiments conducted under different setups. The tags of experiments consist of 7 parts. The first part indicates how we classify simulees, and can take **Bn**, **ED**, or **MD** as its value. **Bn** means that we use Bayesian networks to compute the probability distribution over `group` to select the most probable subgroup, **ED** that we rely on the Euclidean distance-based measure in (6) to guess the simulee's subgroup, and **MD** that we rely on the Mahalanobis distance-based measure in (7) to guess the simulee's subgroup. The second part indicates how we select items to administer, and can take **Mi**, **HMi**, **Dist**, and **Rand**. **Mi** means that we use the exact conditional mutual information, **HMi** that we use both the distance-based measure in (5) for selecting concepts and the heuristic mutual information in (3) for selecting items, **Dist** that we use only the distance-based measure in (5) for selecting concepts but randomly select the item for the selected concept, **Rand** that items are

randomly selected for randomly chosen concepts. The third part indicates the number of concepts, the fourth part the number of student group, the fifth part α , and the sixth part β . We use “2” and “5”, respectively, to indicate 0.2 and 0.05 for α and β . The choice for 0.2 and 0.05 was arbitrary. We used 0.05 to represent the situation when there is small chance of deviation from stereotypical behaviors, and 0.2 to represent a relatively large chance. For this problem, researchers had chosen to use different values, e.g., 0.1, 0.2, and 0.3, in their work (Collins, 1996; VanLehn et al., 1998). We put an “a” as the seventh part if we use the alternative network shown in Figure 5. There will not be an “a” in the tag if we used the network shown in Figure 2. Recall that, when we discuss the application of (5) to choose concepts for the **HMi** and **Dist** methods, we reset U to an empty set when an item for each concept is administered in order to give a flavor of content balance in our item selection. We follow this principle in **Rand**, so **Rand** did not choose concepts in an absolutely random manner.

A valid tag for an experiment looks like **BnMi7925**, for instance. The tag indicates that simulees are classified based on the probability distributions computed with a Bayesian network, that the exact mutual information was used to select test items, and that there are 7 concepts and 9 possible student groups in the problem. The fact that α is 2 indicates that we set *guess* and *slip* to 0.2. The last number, 5, is the value for β , indicating that we set *gError2* and *gError1* to 0.05. This tag does not have the seventh part, so the experiment would have been conducted with the network shown in Figure 2.

Notice that some combinations of the methods for item selection and student classification are impractical, and will be included for comparison purposes. For instance, it is very unlikely that one would use mutual information for item selection and the Euclidean distance-based measure for student classification as we will in **EDMi**

Influences of the Simulation Parameters

The charts in Figure 6 show the simulation results of using different setups in the experiments. The experiments that belong to the **BnMi** family employed the MI-ADAPT procedure directly. As just been explained, the **EDMi** family used a similar procedure, except that the Euclidean distance-based heuristic in (6) was used to determine simulees’ groups.

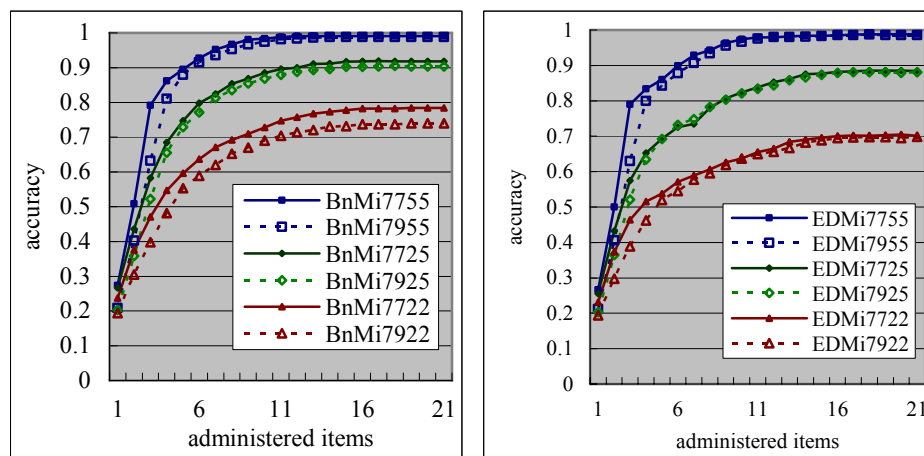


Figure 6. Influences of number of groups, guess, slip, *gError2*, and *gError1*

The curves in these charts unambiguously support the intuition that it becomes increasingly more difficult to correctly classify simulees as we increase the values of α (*guess* and *slip*) and β (*gError2* and *gError1*). The performance profiles of both the **BnMi** and the **EDMi** families drop significantly when we increase α from 0.05 to 0.2 and β from 0.05 to 0.2. For instance, after administering 10 test items, performance profiles for **BnMi7755**, **BnMi7725**, and **BnMi7722** are 10% apart in accuracy. Although the differences are much smaller, the same support occurs in situations when we increase the number of possible student groups from 7 to 9 in the experiments. The difference between **BnMi7722** and **BnMi7922** is more extreme, reaching almost 5% in accuracy.

The results of exploring the influences of using different networks are shown in Figure 7. The curves could be shown in the corresponding charts in Figure 6, but doing so would reduce the readability of the charts as a whole. As noted above, the curves whose tags end with “a”s come from results of experiments that we used the network shown in Figure 5. The charts in Figure 7 suggest that using a more complex model degraded the

performance of classifiers that used either the BnMi or the EDMi approaches. The degradation is more salient for EDMi than for BnMi, suggesting that the extra costs of computing the probability distribution over group in the Bayesian networks can be worthwhile.

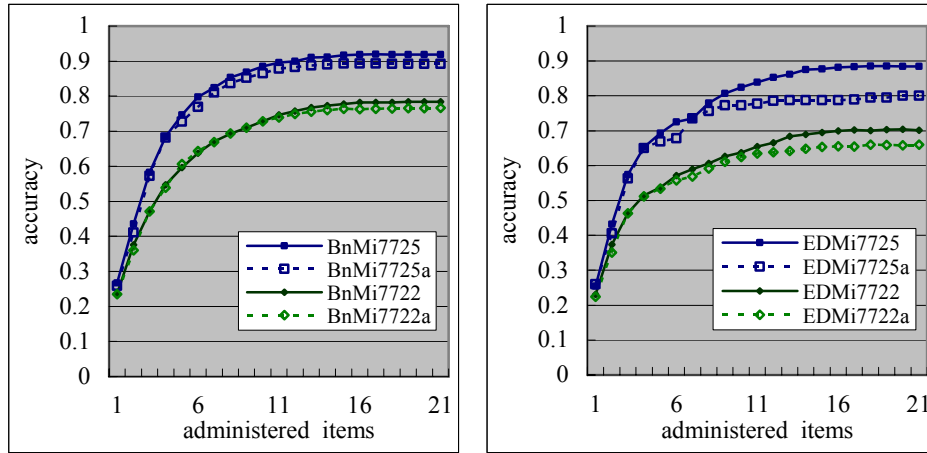


Figure 7. Effects of using different networks in Figures 2 and 5

Effects of the Heuristics

The charts in Figure 8 show the results of comparing the effects of different ways of deciding simulees' groups. Using the probability distributions computed in Bayesian networks offered better performance profiles than using the Euclidean distance-based heuristic in (6) under all different setups. The differences occurred not just when we used up all 21 test items, but also when we used just a few test items. For some examples, it took BnMi7725 and EDMi7725, respectively, 6 and 9 test items to achieve 80% in accuracy. When all 21 test items were used up, there was a noticeable gap between the profiles of BnMi7922 and EDMi7922.

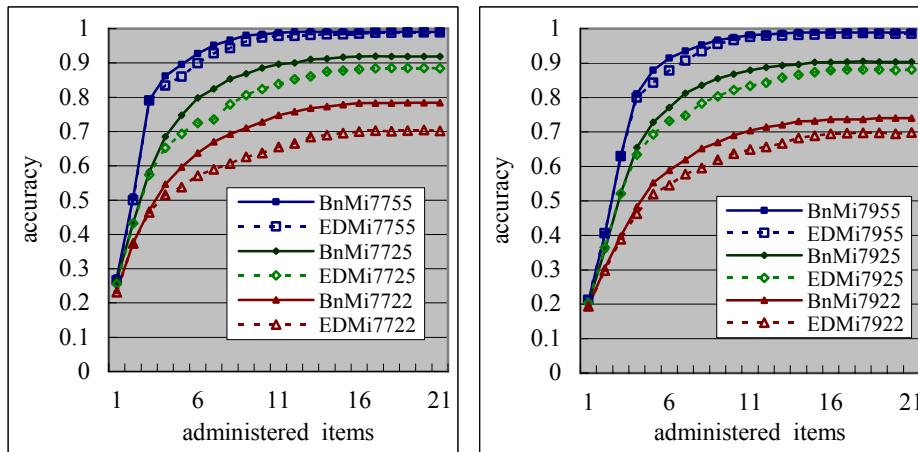


Figure 8. Using Bayesian networks for classification outperforms using Euclidean distances

The charts in Figure 9 show results of comparing different strategies for item selection. The left chart shows the profiles for using the **Mi**, **HMi**, **Dist**, and **Rand** strategies while we used the probability distributions computed in Bayesian networks for classifying students. The curves for the **Dist** and the **Rand** strategies almost overlapped throughout the experiments. This phenomenon occurred in other experiments as well, so the curves for **Dist** are not shown in the right chart and other following charts. The curves support the viability of the heuristic proposed in (3), although the differences in using **HMi** and **Rand** appear to be smaller when α and β are large. No matter whether we used **Bn** or **ED** for classifying simulees, using **HMi** provided better performance profiles than using **Dist** and **Rand**. It took 6 test items for **BnMi7725** to achieve 80% in correct classification, 9 test items for **BnHMi7725**, and 12 test items for **BnRand7725**. Similarly, it took 9 test items for **EDMi7725** to achieve 80% in correct classification, 14 test items for **EDHMi7725**, and 16 test items for **EDRand7725**.

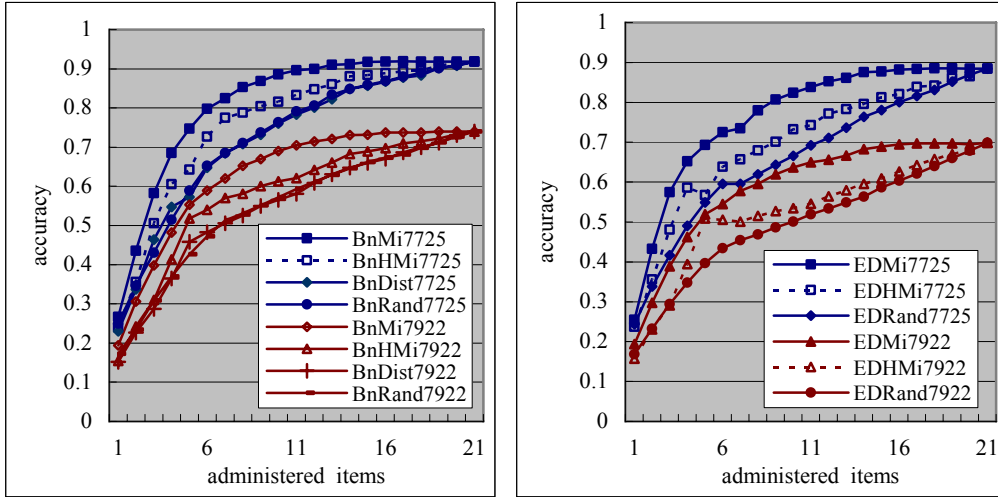


Figure 9. The heuristic designed based on the discovered theorem provides good guidance

Extending to Mahalanobis Distance

The following charts depict the classifier’s performance when we used the Mahalanobis distance-based heuristic in (7) for classifying simulees. Qualitatively, the results are not very different from those shown in Figures 6 and 7. Increasing the values of α , β , and the number of possible groups of simulees decreased the accuracy, as suggested by the curves in the left chart. Curves in the right chart indicate that making the network more complex decreased the accuracy as well.

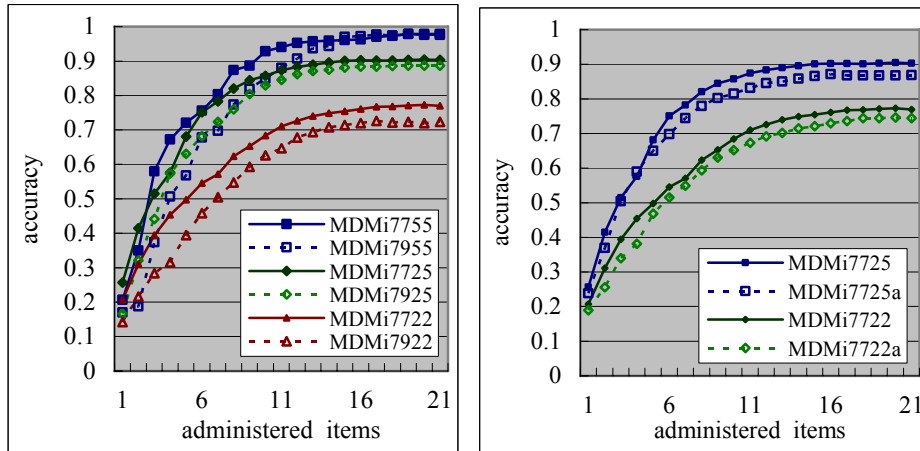


Figure 10. Parameters of simulation have similar influence on Mahalanobis distance-based heuristics

The charts in Figure 11 show how the performance of the classifier changed when we used the Mahalanobis distance-based heuristic in step 2 of MI-ADAPT for classifying simulees. In all direct comparisons, using probability distributions that were computed with the Bayesian networks provided better performance profiles. Comparing the charts in Figure 11 and Figure 8 reveals interesting insights into a drawback of how we applied the concept of Mahalanobis distance. Take the left charts in Figures 11 and 8 for example. When more test items were administered, using the Mahalanobis distance-based heuristic provided better performance than using the Euclidean distance-based heuristic. When few test items were administered, the advantages went to the Euclidean distance-based heuristic. We would not jump to the conclusion that the Euclidean distance-based heuristic is better than the Mahalanobis distance-based heuristic, if we recall the definition of R in (6) and (7). Each component of the competence pattern R , say, rC_m , is the ratio of the student’s correct responses to administered items for C_m . When no or only one item was tested for C_m , the quantity rC_m will not be a very reliable measure, so won’t R . This unreliable R happens to have stronger influence on the performance of the Mahalanobis distance-based heuristic in the chosen experiments.

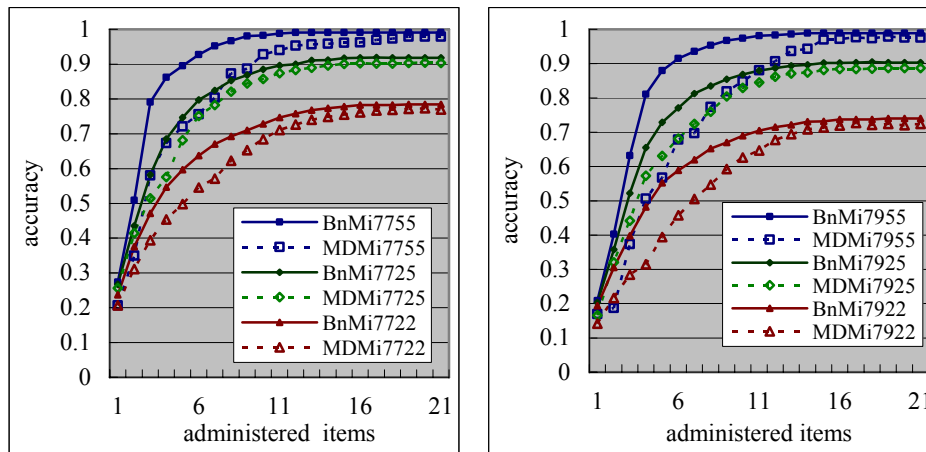


Figure 11. Using Bayesian networks for classification outperforms using Mahalanobis distances

Similar to the results shown in Figure 9, the chart in Figure 12 indicates that the mutual information-based heuristic helps to select the test items that are more effective for classifying simulees. The crossing of **MDMi7922** and **MDHMi7922** may be surprising initially, but the crossing is not impossible. Given the previous explanation on R and the fact using **Mi** and **HMi** may select different items in the tests, using **MDMi** does not guarantee to provide better performance profiles than using **MDHMi**.

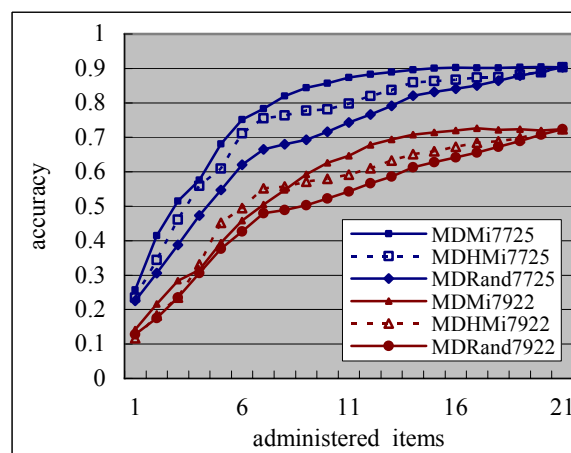


Figure 12. The theorem-based heuristic provides good guidance for MD-based classification as well

Overall Evaluation

Due to the randomness in generating the simulees, a certain percentage of simulees showed typical behavior of other subgroups, and were impossible to be correctly classified. The percentage of such wildly behaving students reflects the difficulty levels for the classification tasks. For instances, for the scenarios that had 7 possible student groups and $(\square, \square)=(0.2, 0.05)$, about 6% of the simulees had this type of problem. For the scenarios that had 7 possible groups and $(\square, \square)=(0.2, 0.2)$, the percentage climbed to 20%. The curves for **BnMi7725** and **BnMi7722** in Figure 6 indicate that, under these constraints, MI-ADAPT was able to offer very high accuracy when all 21 test items were used. In contrast, using either the Mahalanobis or Euclidean distance-based heuristics offered accuracy only near 70% in the latter scenario. On the other hand, when there were seven and nine possible student subgroups, a blind guess should hit the correct answer about 14.3% and 11.1% of the time, respectively. All of the studied methods did better than this baseline even when we used the results obtained by administering only the very first item.

Comparing all the charts, it should be clear that the **BnMi** curves dominated all other curves, both in terms of the achieved accuracy and the administered number of items necessary for achieving a particular degree of accuracy. Although simulation results cannot establish sound basis for accepting our proposed methods, these results do

support viability of MI-ADAPT. Also based on the simulation results shown in the charts in Figure 9, **BnHMi** provided a pretty good alternative for **BnMi**, when computing mutual information at run time was a concern. When we did not have a complete Bayesian network, but had the competence patterns of subgroups and $PR(X|C)$ for all items X and their parent concepts C , the heuristic designed based on Theorem 2 and its corollaries may help in selecting good test items, as shown in charts in Figures 9 and 12.

A Comparison with the Item Response Theory

Item Response Theory (cf. Hambleton, 1991) is such a dominant theory for educational assessment that we have to compare our models and IRT models in more details. There are three IRT models, each including different number of factors in the model. The three-parameter model considers item discrimination a_i , item difficulty b_i , and the guess parameter c_i . The model prescribes that a simulee with competence θ will respond to item I correctly with probability provided in (8), where k_i is a constant for normalization.

$$\Pr(i | \theta) = c_i + \frac{1 - c_i}{1 + e^{k_i a_i (\theta - b_i)}} \quad (8)$$

For grading simulees, it is common to assume that the probabilities of correct responses to different items are independent given a particular θ . Assume that $\mathcal{S} = \{i_1, i_2, \dots, i_t\}$ is the set of items administered in the test. Applying IRT, we estimate the competence Θ of the simulee using the following formula.

$$\Theta = \arg \max_{\theta} \Pr(\theta | \mathcal{S}) = \arg \max_{\theta} \Pr(\mathcal{S} | \theta) \Pr(\theta) = \arg \max_{\theta} \prod_{i_j \in \mathcal{S}} \Pr(i_j | \theta) \Pr(\theta) \quad (9)$$

The second equality in (9) is based on the independent assumption for responses to test items given students' competence levels. It should be clear that formula (9) is a realization of the naïve Bayes (**NB**) models (Mitchell, 1997). Although formula (8) is significantly more complex than typical formula used in NB models, there is no essential difference between NB models and IRT models when we use (9) for grading simulees.

From this perspective, we can easily see that the models we build, e.g., those shown in Figures 2 and 5, are more complex than the IRT models. Given that we know a simulee's type, say g , the probability of correctly responding to different items, e.g., I and J , remains *dependent* in our models. More specifically, unlike IRT models, the equality in (10) is not guaranteed in our models. Moreover, the equality will hold only if the parent concepts of the test items are independent given the testee's identity, which generally does not hold in our simulations and in reality.

$$\Pr(i, j | g) = \Pr(i | g) \Pr(j | g) \quad (10)$$

The dependence between responses to test items should be common, in practice. However, dependence relationship is not as simple as a yes/no problem, and the strength of dependence is more of a concern. In realistic reasoning systems, it may be fine to ignore weak dependence between variables to trade for computational efficiency. Consider an extreme incarnation of the network shown in Figure 2. Let cA , cB , and cC represent basic arithmetic competences, and let dAB , dBC , and dAC represent concepts that integrate the basic concepts. The responses to items designed for the composite concepts will remain dependent given the identity of the student, i.e., *group*. Among all mutual dependences among the responses, some are stronger than others. It is possible that ignoring weak dependent relationships may not have a detrimental impact on the final outcomes of the reasoning system. Due to the author's limited experience in the education domain, the preceding analysis reflects what one can see from an abstract, rather than a practical, viewpoint. Whether the mutual dependent relationships matters in practice depends on the specific details for individual applications.

In summary, there are three major differences between our and the IRT models. Firstly, the responses to different test items may remain dependent given the identity of the simulee in our models. Secondly, students are classified into types not competence levels in our work, although we may design a conversion mechanism between these two criteria. Thirdly, because we are assuming that all random variables are dichotomous in this paper, our current simulations use only two parameters for each item, which is not as expressive as the 3-parameter IRT model. The function of c_i is undertaken by the parameter *guess*, and the functions of a_i and b_i are undertaken by the parameter *slip* in our work. It should be clear that the MI-ADAPT procedure allows more complex models than the dichotomous ones. Allowing the variables that represent the mastery statuses of

concepts to take more than two possible values, we will have more expressive power to catch the concepts of discrimination and difficulty of test items. Paying for the gains in expressiveness, it would become harder to compare test items purely based on their parameters.

In an attempt to compare our approach and the NB-based approach, we have begun to compare the effectiveness of using the network shown in Figure 2 and using a comparable NB model for classifying students. We look forward to reporting the results to the research community in the next months to come.

Concluding Remarks

The main contribution of this paper is the theoretical foundation for comparing the effectiveness of test items based on mutual information. Theorem 2 turns out to be a good vehicle for explaining some intuitions for item comparison, and provides a basis for item comparison. In addition, the theorem and its corollaries allow us to design heuristics when computing exact values of mutual information online is considered too costly. Although simulated experiments cannot establish decisive conclusions for viability of mutual information-based heuristics for item selection, the current results are definitely encouraging.

Successful applications in computer assisted learning must do well in inferring about students' internal statuses from their external behaviors. Important functions such as adaptive testing and course sequencing relies this core technologies. Hence, the literature has seen an abundance of research tackling this issue from different perspectives. This paper only skims through a handful of related work from the literature, and a (hopefully) broader survey is provided in (Liu, in press).

A major flaw of the current evaluation procedure is that we employed only simulated students. Although it is easy to categorize concepts and draw dependent relationships among concepts in imagination, it may not be easy to realize the postulates in real life. The author would hypothesize that the proposed idea may be more readily useful for science education than language learning. Intuitively, it is relatively easy to define basic concepts in Mathematics and Physics than in English and Chinese. We would not be able to make convincing comparison of our work with other researchers' approaches without taking real students into the evaluation procedure.

As an anonymous reviewer points out, it is questionable to use the same environment, particularly the same Bayesian network structure, for both creating simulees and evaluating the strategies for item selection. The current evaluation procedure has followed standard machine learning steps. The parameters for the Bayesian networks, which were used in the evaluation, were learned from training data, so we have allowed the resulting Bayesian network to be different from the Bayesian network that was used to generate simulees. It is not deniable that our environment might give advantages to the Bayesian networks-based approaches, but a fair judgment may require the incorporation of real students into the evaluation procedure. Learning the network structure and parameters completely from data is possible (Heckerman, 1999), but few, if any, have tried this possibility for real world applications of computer assisted learning.

Another obvious problem of the evaluation method appears at the first step of MI-ADAPT. At that step, we always chose the test item that has the largest mutual information with the target variable, i.e., *group*. This choice will not work in real life, as every student will learn the answer to this particular test item very quickly. This design choice in MI-ADAPT was partially because MI-ADAPT was not designed for realistic testing and partially because our imaginary item bank contained only 21 test items. If we do have a large item bank, we can choose one test item from a reasonable amount of test items for starting the test procedure, and avoid repeatedly using the same item for all tests.

There is plenty of room for more future work. For instance, including mutual information in a decision theory-based system is clearly an option, e.g., (Mayo, 2001). What structure of the Bayesian network should be used to realize the competence patterns in Table 1 deserves a lengthy discussion. Different structures of the network imply different learning patterns of students. The charts shown in Figures 7 and 10 suggest that network structures influence the classification accuracy. We have begun our investigation in this regard (Wang and Liu, 2004), and hope to produce a more complete report on this front shortly. The contents of the *Q-matrix* must also have strong influence on the quality of classification, and we have changed the contents of the *Q-matrix* in Table 1 by removing two subgroups in some of the experiments. However, we have not thoroughly explored issues in this direction. Correct classification will become more difficult when types between different groups become more similarity which is computed based on the competence patterns among the students' groups. We have also begun our investigation in this direction (Liu and Liu, 2004), and hope to report more results soon.

Acknowledgements

The author would like to thank many anonymous reviewers for their invaluable comments on earlier versions of this paper. This research was supported in part by Grants 92-2213-E-004-004 and 93-2213-E-004-004 from the National Science Council of Taiwan.

References

- Beck, J. E. (2002). Directing development effort with simulated students. *Lecture Notes in Computer Science*, 2363, 851–860.
- Birenbaum, M., Kelly, A. E., Tatsuoka, K. K., & Gutvitz, Y. (1994). Attribute mastery patterns from rule space as the basis for student models in algebra. *International Journal of Human-Computer Studies*, 40 (3), 497–508.
- Collins, J. A., Greer, J. E., & Huang, S. X. (1996). Adaptive assessment using granularity hierarchies and Bayesian nets. *Paper presented at the Third International Conference on Intelligent Tutoring Systems*, June 12-14, 1996, Montreal, Canada.
- Conati, C., Gertner, A., & Vanlehn, K. (2002). Using Bayesian networks to manage uncertainty in student modeling. *User Modeling and User-Adapted Interaction*, 12, 371–417.
- Cooper, G. F. (1990). The computational complexity of probabilistic inference using Bayesian networks. *Artificial Intelligence*, 42, 393–405.
- Cover, T. M., & Thomas J. A. (1991). *Elements of Information Theory*, New York, NY: John Wiley & Sons.
- Dagum, P., & Luby, M. (1993). Approximating probabilistic inference in Bayesian belief networks is NP-hard. *Artificial Intelligence*, 60, 141–153.
- Dayton, C. M. (1991). Educational applications of latent class analysis. *Measurement and Evaluation in Counseling and Development*, 24 (3), 131–141.
- Hambleton, R. K., Swaminathan, H., & Rogers, H. J. (1991). *Fundamentals of Item Response Theory*, London: SAGE.
- Heckerman, D. (1999). A tutorial on learning with Bayesian networks. In Jordan, M. I. (Ed.), *Learning in Graphical Models*, Cambridge, MA: MIT Press, 301-355.
- Jensen, F. V. (2001). *Bayesian Networks and Decision Graphs*, Heidelberg, Germany: Springer.
- Lauritzen, S. L. (1995). The EM algorithm for graphical association models with missing data. *Computational Statistics & Data Analysis*, 19 (2), 191–201.
- Leung, C.-K., Chang, H.-H., & Hau, K.-T. (2003) Incorporation of content balancing requirements in stratification designs for computerized adaptive testing. *Educational and Psychological Measurement*, 63 (2), 257-270.
- Liu, C.-L. (2004). Using mutual information for adaptive student assessments. In Kinshuk, Looi C.-K., Sutinen E., Sampson D., Aedo I., Uden L. & Kähkönen E. (Eds.), *Proceedings of the 4th IEEE International Conference on Advanced learning Technologies 2004*, Los Alamitos, CA: IEEE Computer Society, 585–589.
- Liu, C.-L. (In Press). Using Bayesian networks for student modeling, Alkhalifa, E. M. (Ed.), *Cognitively Informed Systems: Utilizing Practical Approaches to Enrich Information Presentation and Transfer*, accepted to appear, Idea Group Inc.
- Liu, C.-L., Wang, Y.-T., & Liu, Y.-C. (2004). A Bayesian network-based simulation environment for investigating assessment issues in intelligent tutoring systems. *Paper presented at the International Computer Symposium 2004*, December 15-17, 2004, Taipei, Taiwan.

- Liu, C.-L. (2005). Some theoretical properties of mutual information for student assessments in intelligent tutoring systems. *Paper presented at the 15th International Symposium on Methodologies for Intelligent Systems*, May 25-28, 2005, Saratoga Springs, New York, USA.
- Liu, Y.-C., & Liu, C.-L. (2004). Some simulated results of classifying students using their item response patterns. *Proceedings of the 9th TAAI Conference on Artificial Intelligence and Applications*, CD-ROM. (in Chinese)
- Mayo, M., & Mitrovic, A. (2001). Optimising ITS behaviour with Bayesian networks and decision theory. *International Journal of Artificial Intelligence in Education*, 12, 124–153.
- Millán, E., Pérez-de-la-Cruz, J. L., & Suárez, E. (2000). Adaptive Bayesian networks for multilevel student modeling. *Paper presented at the 5th International Conference on Intelligent Tutoring Systems*, June 19-23, 2005, Montréal, Canada.
- Mislevy, R. J., Almond, R. G., Yan, D., & Steinberg, L. S. (1999). Bayes nets in educational assessment: Where do the numbers come from? *Paper presented at the 15th Conference on Uncertainty in Artificial Intelligence*, July 30 - August 1, 1999, Stockholm, Sweden.
- Mislevy, R. J., & Gitomer, G. H. (1996). The role of probability-based inference in an intelligent tutoring system. *User Modeling and User-Adapted Interaction*, 5, 253–282.
- Mislevy, R. J., Steinberg, L. S., Breyer, F. J., & Almond, R. G. (1998). *A Cognitive Task Analysis, with Implications for Designing a Simulation-based Performance Assessment*, CSE Tech. Report 487, UCLA, USA.
- Mitchell, T. M. (1997). *Machine Learning*, New York: McGraw-Hill.
- Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, San Mateo, CA: Morgan Kaufmann.
- Tatsuoka, K. K. (1983). Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of Educational Measurement*, 20, 345–354.
- Tatsuoka, K. K., & Tatsuoka, M. M. (1987). Bug distribution and pattern classification. *Psychometrika*, 52, 193-206.
- VanLehn, K., & Martin, J. (1997). Evaluation of an assessment system based on Bayesian student modeling. *International Journal of Artificial Intelligence in Education*, 8 (2), 179–221.
- VanLehn, K., Niu, Z., Siler, S., & Gertner, A. (1998). Student modeling from conventional test data: A Bayesian approach without priors. *Lecture Notes in Computer Science*, 1452, 434–443.
- VanLehn, K., Ohlsson, S., & Nason, R. (1994). Applications of simulated students: An exploration. *International Journal of Artificial Intelligence in Education*, 5 (2), 135–175.
- Wang, Y.-T., & Liu, C.-L. (2004). An exploration of mapping the learning processes of composite concepts. *Proceedings of the 9th TAAI Conference on Artificial Intelligence and Applications*, CD-ROM. (in Chinese)
- Welch, R. E., & Frick, T. W. (1993). Computerized adaptive testing in instructional settings. *Educational Technology Research and Development*, 41 (3), 47–62.
- Wellman, M. P. (1990). Fundamental concepts of qualitative probabilistic networks. *Artificial Intelligence*, 44 (3), 257–303.
- Yan, D., Almond, R. G., & Mislevy, R. J. (2003). Empirical comparisons of cognitive diagnosis models. *Paper presented at the Annual meeting of the National Council on Measurement in Education*, April 24-26, 2003, Chicago, IL, USA.