Multiple Representations, Problem-Solving Behavior and Educational Outcomes

Ryan Carlson
Language Technologies
Institute
Carnegie-Mellon University
Pittsburgh, PA 15213
rcarlson@cs.cmu.edu

Konstantin Genin
Department of Philosophy
Carnegie-Mellon University
Pittsburgh, PA 15213
kgenin@andrew.cmu.edu

Helga Caballero
School of Public and
International Affairs
University of Pittsburgh
Pittsburgh, PA 15260
hec33@pitt.edu

Martina Rau
Human-Computer Interaction
Institute
Carnegie-Mellon University
Pittsburgh, PA 15213
marau@cs.cmu.edu

Richard Scheines
Department of Philosophy
Carnegie-Mellon University
Pittsburgh, PA 15213
scheines@cmu.edu

Clark Glymour
Department of Philosophy
Carnegie-Mellon University
Pittsburgh, PA 15213
cg09@andrew.cmu.edu

ABSTRACT
We analyze log-data generated by an experiment with Mathtutor, an intelligent tutoring system for fractions. The experiment was performed to compare the educational outcomes of students presented with single and multiple graphical representations. For each student we extract low-level, computationally efficient features characterizing error and hint-seeking propensities. We use latent class analysis to cluster the students by their problem-solving behaviors, using extracted features as proxies. We then explore how these behaviors interact with the mode of representation and post-test outcomes.

We find that while experimental condition and outcomes are unconditionally dependent, latent class membership screens off experimental condition from outcome for all but one class. That is, we identified the single cohesive class of students (grouped by their behaviors) where the experimental condition affected learning outcomes. The behaviors that compose this group provide a window into the types of students who most benefit from multiple representations. Furthermore, our methodology can be leveraged by future tutoring system designers to create dynamic systems that are personalized to the student.

1. INTRODUCTION
Multiple Graphical Representations (MGRs) are used extensively in middle-school fraction instruction. Fractions can be alternately presented as pie and rectangle graphs, number lines, or discrete sets of objects. The educational psychology literature suggests that multiple-representations support learning in a variety of ways, though the experimental results are not univocal [1]. Because of the significant log data that intelligent tutoring systems (ITSs) generate from user interactions, these systems are well-suited for conducting experiments on the effect of multiple representations on learning outcomes. The influx of data also suggests that machine learning methods can successfully discern the effectiveness of MGRs and the factors mediating their success.

Prior work conducted on middle-school students working with ITSs found that multiple representations, in conjunction with self-explanation prompts, contribute to improved learning outcomes [8]. Subsequent studies examining error-rate, hint-use and time-spent in ITS logs failed to identify variables that mediate the effectiveness of multiple representations [7]. The mechanisms by which multiple representations improve learning outcomes remain poorly understood.

We conjecture that previous efforts to identify mediating factors were frustrated by heterogeneity in the problem-solving habits and behaviors of the student population under investigation. Using latent class analysis, a mixture modeling technique, we cluster students by their patterns of interaction with the tutor and examine how multiple representations affect post-test outcomes in the resulting sub-populations. Confirming previous results, we find that experimental condition correlates with post-test outcome. Furthermore, we establish independence between cluster membership and experimental condition (i.e., students are randomly distributed into clusters across conditions). Most interestingly, we discover that if we condition on cluster membership, all but one sub-population of students shows conditional independence between experimental condition and outcome. That is, we have identified the group of students where multiple representations correlates with post-test scores.

The rest of this paper is structured as follows. Section 2 describes the initial experiment and elaborates on the dif-

ferent graphical representation conditions. We then provide some background in Section 3. We describe our feature extraction process and the mixture model we chose in Section 4. Section 5 covers the results of the model and additional experiments we ran using the clusters created by the model, and we offer concluding remarks and future directions in Section 6.

2. EXPERIMENT

In the Spring of 2010, Rau conducted an experiment wherein 290 4th and 5th grade students worked with an interactive fractions tutor for about 5 hours of their mathematics instruction. Students were randomly assigned to one of five experimental conditions. Conditions varied by the frequency with which they would be presented with a new fraction representation. Students in the SINGLE representation condition worked exclusively with either a number line, a circle or a rectangle. Students in the FULLY INTERLEAVED condition would see a different representation with every new problem. Students in the intermediate conditions would go longer before they were presented with a different representation.
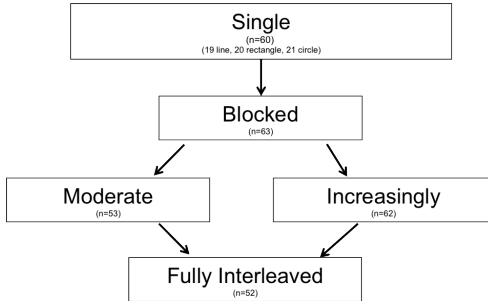

Figure 1: A partial ordering of experimental conditions by the frequency with which a new representation is presented.

When interacting with different graphical representations of fractions, students were able to drag-and-drop slices of a pie graph, for example, into separate areas. They were also able to experiment with changing the number of subdivisions in each graphical representation. Students received a pre-test on the day before they began working with the tutor and an immediate post-test on the day after they finished. Students also took a delayed post-test a week after the first. Previous investigation found that students in the multiple representation conditions significantly outperformed students in the single representation condition on the delayed post-test [7].

3. RELATED WORK

Latent Class Analysis and other clustering tools have commonly been used to analyze educational data. LCA can be used either for explanatory or confirmatory purposes. An application of clustering for educational data is presented in [5], where students were clustered by types of mistakes. The authors suggested that identifying the different types of learners could help teachers apply remedial methods. Similarly, [6] applied clustering techniques to group teams of students by the amount of work they performed. With LCA techniques they were able to reduce seven groups to

three clusters. [11] provided a comparative analysis of simulated data for Rule Assessment Methodology and LCA. They found five classes with conditional probabilities close to 0.95. They also found that the five classes worked better with RAM than the previous class definitions. Finally, [10] proposed a four-phase learning portfolio mining (LPM) approach, which uses sequential pattern mining, clustering and decision tree creation to find features from the portfolio and predict which group a new learner belongs to. Additional techniques used in educational data mining can be found in [9].

4. METHOD

Our analysis proceeds in three stages. Extracting features characterizing error and hint-seeking behavior, we transform the longitudinal log data into a cross-sectional form, with one observation per student. We then run Latent Class Analysis to identify sub-populations of students, using AIC and BIC to select the number of latent classes.

Once we have clustered our students, we investigate the interaction between the latent classes and their experimental conditions. We construct a contingency table binning the experimental conditions into the clusters estimated by the latent class model. We then run a Chi-squared test for independence between experimental condition and latent class. Chi-squared tests are also run to investigate dependence between latent class and post-test outcome and the conditional dependence of outcome and experimental condition, given latent class membership.

4.1 Extracting Features

The Cognitive Tutor captures a detailed log of each student's interactions with the tutor. It stores a time series of correct and incorrect answers, hint requests, interface selections and durations between interactions. Previous analysis (Scheines, Rau, 2012) extracted the average number of errors made per step, the average number of hints requested per step, and the average time spent per step from the log data. Similarly, we include the average number of hints requested ($hints\_req$) and number of errors (num_errors) made per *problem* by each student. We also extract the average number of bottom-out hints ($NumBOH$) per student per problem – this is the average number of times a student exhausts the available hints in a given problem. We also note that it is not always the average of these features that best characterizes a student. For example, examination of the distribution of hints requested per step across experimental condition, shows a telling picture.

Note that students who received only one representation start out requesting the fewest hints, but students in the moderate condition eventually need fewer. Such considerations motivated our interest in the temporal distribution of hint behavior at the student level. We fit geometric distributions to the number of steps taken before the first hint request ($firstHintGeom$) and to the number of errors before the first hint ($stubbornGeom$). The estimated parameter is used to characterize the student's hint-seeking propensity in general and hint-seeking propensity when faced with adversity. For example, students in the first quintile of $stubbornGeom$ seek help soon after making a mistake, whereas students in the fifth quintile don't change their hint-seeking behavior
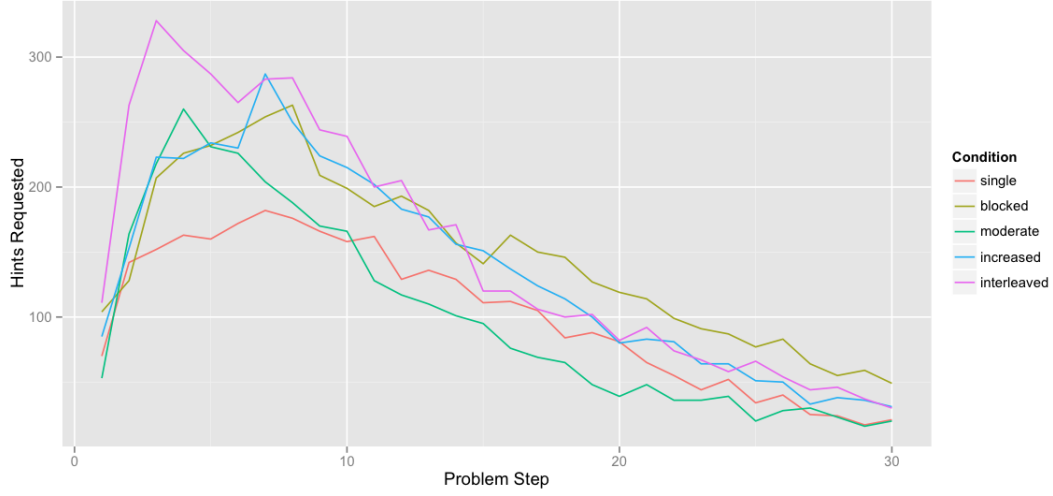
Figure 2: The x-axis represents the n_{th} interaction with the tutor across all problems. The y-axis is the total number of hints requested at the n_{th} step.

Table 1: Summary Statistics for Variables Used in Clustering

	mean	sd	median	min	max	20%	40%	60%	80%	100%
hints_req	0.78	1.27	0.34	0	11.22	0.06	0.19	0.5	1.31	11.22
num_errors	2.21	1.27	1.92	0.34	8.39	1.15	1.7	2.18	3.19	8.39
firstHintGeom	0.35	0.27	0.27	0.04	1	0.13	0.2	0.33	0.57	1
stubbornGeom	0.36	0.21	0.31	0.07	1	0.19	0.27	0.38	0.47	1
NumBOH	0.04	0.08	0	0	0.62	0	0	0.01	0.05	0.63

even after making a large number of errors. Students in the first quintile of *firstHintGeom* are likely to request hints early in a problem, whereas students in the fifth quintile are unlikely to request hints at any point.

4.2 Latent Class Analysis

Latent Class Analysis (LCA) is a modeling technique that determines subtypes based on multinomial distributions. We use LCA to categorize students into *latent classes* using discretized versions of the features described above. Table 1 shows summary statistics and cut-off points for the extracted features. The model maps a set of observed categorical variables onto a set of inferred latent variables.

We note that the categorical nature of the model has the potential to add some noise, since we must select numeric cutoffs to transform our variables into nominals. However, categorical models can offer greater interpretability by allowing us to organize our data into a small set of variables, which forms the basis for categorizing students into a small set of meaningful homogenous groups. Furthermore, it is not unreasonable to suspect that our variables are in some sense "truly" categorical.

The formal representation of LCA begins with $j = 1 \ldots J$ observed variables, where each such variable j has a set of response variables $r_j = 1, \ldots, R_j$. Each student has a distinct response pattern $y = (r_1, \ldots, r_j)$.

Now we need to consider the latent classes. Let L be a latent variable with latent classes $c = 1, \ldots, C$. Furthermore, let γ_c be the probability of membership in class c. Note that latent classes are exhaustive and mutually exclusive, so each student is a member of exactly one latent class. We also need to define the item-response probability $\rho_{j,r_j|c}$, which is the probability of response r_j to observed variable j, conditional on membership in latent class c. Each student provides exactly one response alternative to variable j. Given these constraints, note that

$$\sum_{c=1}^{C} \gamma_c = 1, \quad \sum_{r_j=1}^{R_j} \rho_{j,r_j|c} = 1.$$

Now that we have defined key variables, we can define the probability of observing a particular response vector based on the γ's and ρ's:

$$P(Y = y) = \sum_{c=1}^{C} \gamma_c \prod_{j=1}^{J} \prod_{r_j=1}^{R_j} \rho_{j,r_j|c}^{I(y_j=r_j)} \qquad (1)$$

where the indicator function $I(y_j = r_j)$ equals 1 when the response variable $j = r_j$. The parameters γ_c and $\rho_{j,r_j|c}$ are estimated by EM. Since EM is sensitive to starting probabilities, we pick the maximum likelihood over twenty-five runs. LCA is very similar to other EM-based algorithms. In fact, LCA is an application of multivariate mixture estimation using categorical variables with an additional local independence assumption, meaning that the observed variables are independent of each other conditional on the latent variable. This is a simplifying assumption similar to the one made in Naive Bayes; without it Equation 1 would have to be much

more complicated. There is some work on relaxing this independence assumption [4]. To run latent class analysis, we used poLCA, a freely available R package[1].

Note that unlike some common clustering algorithms (e.g., k-means), LCA produces "fuzzy" clusters–probability distributions over features for each class. To cluster students we identify their most likely class:

$$\arg\max_c P(Y = y \mid L = c) = \arg\max_c \gamma_c \prod_{j=1}^{J} \prod_{r_j=1}^{R_j} \rho_{j,r_j|c}^{I(y_j=r_j)}$$
$$(2)$$

We still need to fix C, the number of latent classes. We use two complexity-penalized log-likelihood scores to select an appropriate C: Akaike information criterion (AIC) and Bayesian information criterion (BIC). Plotting these statistics as we increment the number of latent classes, we look for a "knee" where both statistics either bottom-out or level off to identify the optimal value of C.

5. RESULTS

Figure 3 shows the parameter selection process described in section 4.2. Note that we chose to model four latent classes because BIC bottoms out and AIC levels off at that point.
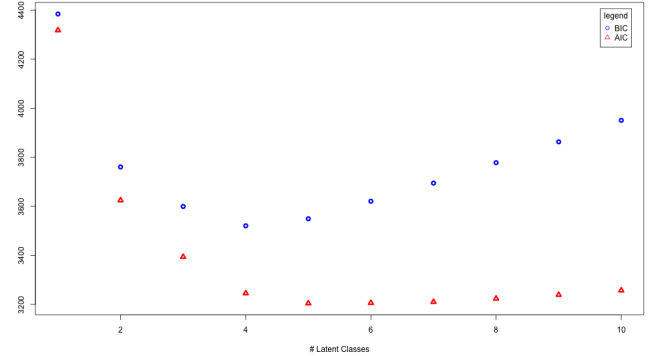
Figure 3: AIC and BIC over increasing number of latent classes. BIC bottoms out and AIC levels off at four classes, so we conclude that four latent classes best fits the data.

5.1 Exploring the Latent Classes

After selecting the appropriate C parameter, we extract membership probabilities for the individual students. Given a latent class, we know the probability distribution over each feature, and use Equation 2 to identify the most likely class for each student.

The feature distributions over each class are represented graphically in Figure 4. Note that each feature is listed along the horizontal x-axis, the value each variable takes takes is along the front-to-back y-axis, and the probability that the feature takes that value is given along the vertical z-axis. For example, consider the *hints_req* feature (average hints

---

[1] `http://userwww.service.emory.edu/~dlinzer/poLCA/`

requested per problem) in Class 2. In that class, with high probability, students requested many hints (i.e., the highest categorical value for hints) per problem on average. As another example, students in Class 1 are more likely to make a moderate number of errors, though other error levels also occur with nontrivial probabilities. Note that lower values of *firstHintGeom* and *stubbornGeom* indicate a steep geometric slope, corresponding to a higher hint-seeking propensity.

How do we interpret latent class membership? Students in Class 1 are "Moderate", they ask for a moderate number of hints, make a moderate number of errors, and are moderately responsive to the interface. Students in Class 2 are "Interactive", they make a lot of errors, but respond by requesting many hints. These students are proactive in asking for help and are not shy about using the resources the cognitive tutor makes available. Students in Class 3 are "Confident", they don't ask for hints, but they don't seem to need them. Finally, the students in Class 4 are "Stubborn", they are fairly mixed in error-profile but they don't respond to mistakes with hint-requests. These students are not using all the resources that the cognitive tutor makes available.

5.2 Condition and Outcome
We construct a measure of student improvement and knowledge consolidation using the pre-test and the delayed post-test:

$$\text{Adjusted Delayed Post-Test Score} = \frac{\text{post\_test} - \text{pre\_test}}{1 - \text{pre\_test}}$$

We then construct terciles of the Adjusted Delayed Post-Test Score and run a Chi-squared test for independence of outcome from experimental condition. Confirming previous results, we reject independence at a *p*-value of .024 (See Table 3).

5.3 Cluster Membership and Outcome
We run a Chi-squared test for independence of latent class membership from outcome on the delayed post-test score. We reject independence at a *p*-value of .0075 (See Table 2). The problem-solving behaviors encoded by latent class membership are highly relevant to a student's learning outcome.

Table 2: Latent Class by Tercile of Adjusted Delayed Post-Test Score

	33%	66%	99%
LC 1	20	35	29
LC 2	33	26	14
LC 3	13	15	22
LC 4	31	20	32

$\mathcal{X}^2 = 17.52$, df = 6, p-value = **0.0075**

5.4 Condition and Cluster Membership
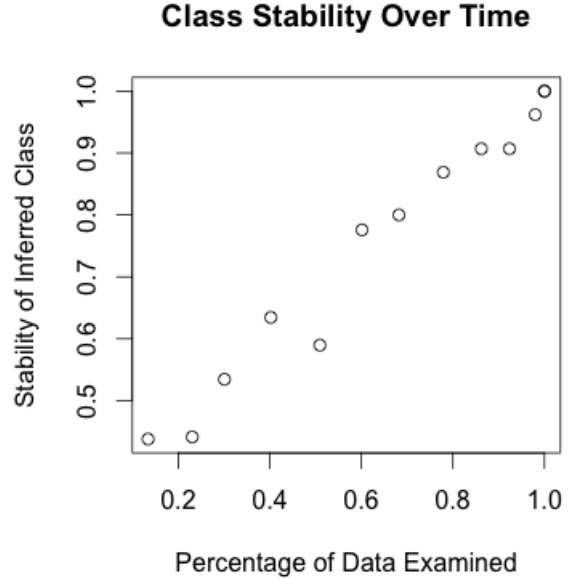

Class Stability Over Time

Figure 5: TODO

We may worry that experimental condition is inducing latent class membership. If this were the case, we would not be detecting pre-existing student profiles, but detecting an artifact of the experimental design. Using the Chi-squared test, we fail to reject independence at a *p*-value of .38 (See Table 3). It is therefore likely that our latent classes are detecting genuinely different student profiles, independent of experimental condition.

5.5 Condition, Outcome and Cluster Membership
Finally, we ask whether we still detect a dependence of outcome on condition within the latent classes we have defined. Interestingly, we find that latent class membership screens off condition from outcome for all classes but the fourth (See Table 4). Recall that these students rarely requested hints, even when they encountered difficulty. Our results suggest that since these students are not taking full advantage of the other resources the tutor makes available, it is the multiple representation condition that most strongly determines their learning outcome. The problem-solving behaviors of students in the other classes makes them insensitive to experimental condition.

5.6 Class Stability
See Figure 5

6. CONCLUSION & FUTURE WORK
We estimated a latent class model to classify students into four groups based on their error-rates and hint-seeking behaviors. We detected dependence of experimental condition and post-test outcome only in the class of students characterized by high-error rate and low hint-seeking propensity. That is, students who did not take full advantage of the resources that the Mathtutor offered were the ones most
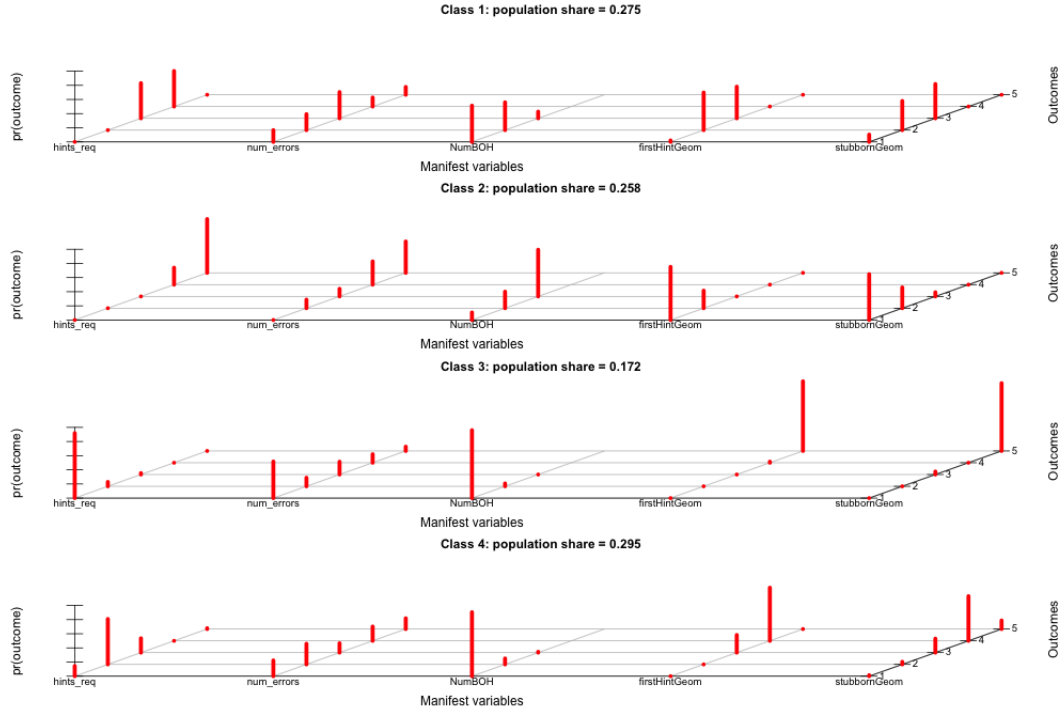
Figure 4: Visualization of feature distributions for each latent class. The left-to-right x-axis identifies each feature, the front-to-back y-axis identifies which value that feature takes, and the top-to-bottom z-axis describes the probability that the feature takes the value. Thus, given a feature and a class, the z-axis also describes the probability distribution over that feature in that class.

Table 3: Condition by Tercile of Adjusted Delayed Post-Test Score (left) and Latent Class (right).

	33%	66%	99%
blocked	14	29	20
increased	22	20	20
interleaved	13	21	18
moderate	18	13	22
single	30	13	17

	LC 1	LC 2	LC 3	LC 4
blocked	13	15	10	25
increased	21	16	10	15
interleaved	17	18	7	10
moderate	18	10	12	13
single	15	14	11	20

$\mathcal{X}^2 = 17.65$, df = 8, p-value = **0.024** $\mathcal{X}^2 = 12.85$, df = 12, p-value = 0.38

Table 4: Condition and Tercile of Adjusted Delayed Post-Test Score, by Latent Class

LC 1	33%	66%	99%
blocked	2	8	3
increased	4	9	8
interleaved	4	9	4
moderate	4	5	9
single	6	4	5

$\mathcal{X}^2 = 8.08$, df = 8, p-value = 0.43

LC 2	33%	66%	99%
blocked	7	6	2
increased	9	5	2
interleaved	5	8	5
moderate	7	2	1
single	5	5	4

$\mathcal{X}^2 = 6.95$, df = 8, p-value = 0.54

LC 3	33%	66%	99%
blocked	0	5	5
increased	3	3	4
interleaved	2	2	3
moderate	3	4	5
single	5	1	5

$\mathcal{X}^2 = 7.41$, df = 8, p-value = 0.49

LC 4	33%	66%	99%
blocked	5	10	10
increased	6	3	6
interleaved	2	2	6
moderate	4	2	7
single	14	3	3

$\mathcal{X}^2 = 17.4837$, df = 8, p-value = **0.025**

strongly affected by experimental condition. These students may not have the meta-cognitive skills required to know when to seek hints [2]. Our methods could be used by intelligent tutoring systems designers to detect students with this profile in real time. Tutoring systems could then intervene to target these students with multiple representations and to scaffold their hint-seeking behaviors. Future research into the mediating mechanisms of multiple representations could leverage our results to identify the relevant student sub-populations to investigate.

7. REFERENCES

[1] S. Ainsworth. The functions of multiple representations. *Computers and Education*, 33(2-3):131–152, September 1999.

[2] V. Aleven, B. McLaren, I. Roll, and K. R. Koedinger. Toward meta-cognition tutoring: A model of help seeking with a cognitive tutor. *International Journal of Artificial Intelligence in Education*, 2006.

[3] L. M. Collins and S. T. Lanza. *Latent Class and Latent Transition Analysis: With Applications in the Social, Behavioral, and Health Sciences*. Wiley Publishing, 2009.

[4] J. A. Hagenaars. Latent structure models with direct effects between indicators: Local dependence models. *Sociological Methods and Research*, 1988.

[5] A. Merceron and K. Yacef. Clustering students to help evaluate learning. *Technology Enhanced Learning*, pages 31–42, 2005.

[6] D. Perera, J. Kay, I. Koprinska, K. Yacef, and O. R. Zaiane. Clustering and sequential pattern mining of online collaborative learning data. *Knowledge and Data Engineering, IEEE Transactions*, 21(6):759–772, 2009.

[7] M. Rau and R. Scheines. Searching for variables and models to investigate mediators of learning from multiple representations. *International Conference on Educational Data Mining*, 2012.

[8] M. A. Rau, V. Aleven, and N. Rummel. Intelligent tutoring systems with multiple representations and self-explanation prompts support learning of fractions. *International Conference of Artificial Intelligence in Education*, pages 441–448, 2009.

[9] C. Romero and S. Ventura. Educational data mining: A survey from 1995 to 2005. *Expert Systems with Applications*, 33(1):135–146, 2007.

[10] J.-M. Su, S.-S. Tseng, W. Wang, J.-F. Weng, J. T. D. Yang, and W. Tsai. Learning portfolio analysis and mining for scorm compliant environment. *Journal of Educational Society and Society*, 9(1), 2006.

[11] H. L. J. van der Maas and M. Straatemeier. How to detect cognitive strategies: commentary on âĂŸdifferentiation and integration: guiding principles for analyzing cognitive changeâĂŹ. *Developmental Science*, 11(4):449–453, 2008.