Book Title
Book Editors
IOS Press, 2003

Comparing apples and oranges: Computational methods for evaluating student and group learning histories in intelligent tutoring systems

Carole Beal [a,1], Paul Cohen [a]
[a] *USC Information Sciences Institute*

Abstract.
Intelligent tutoring systems customize the learning experiences of students. Because no two students have precisely the same learning history, traditional analytic techniques are not appropriate. This paper shows how to compare the learning histories of students and how to compare groups of students in different experimental conditions. A class of randomization tests is introduced and illustrated with data from the AnimalWatch ITS project for elementary school arithmetic.

Keywords.
Randomization Comparing student progress Statistical methods

Interacting with an intelligent tutoring system is like conversing with a car salesperson: No two conversations are the same, yet each goes in roughly the same direction: the salesperson establishes rapport, finds out what you want, sizes up your budget, and eventually makes, or doesn't make, a sale. Within and between dealerships, some salespeople are better than others. Customers also vary, for example, in their budget, how soon they intend to purchase, whether they have decided on a particular model, and so on. Of the customers who deal with a salesperson, some fraction actually purchase a car, so one can compare salespeople with a binomial tests or something similar. Indeed, any number of sound statistical comparisons can be drawn between the *outcomes* of dealing with salespeople: total revenues, distributions of revenues over car model classes, interactions between the probability of sale and model classes, and so on.

Similarly, one can evaluate intelligent tutoring systems on outcome variables: the number of problems solved correctly, or the fraction of students who pass a posttest, and so on. Consider the AnimalWatch tutoring system for arithmetic. Students between the ages of 10 and 12 worked on customized sequences of word problems about endangered species. They were provided with multimedia help when they made errors [1]. The word problems provided instruction in nine topics, including addition, subtraction, multiplication and division of integers, recognizing the numerator and denominator of a fraction, adding and subtracting like and unlike fractions and mixed numbers, and so on. Previous analyses focused on outcome measures such as topic mastery estimates maintained by

[1]Correspondence to: Carole Beal, USC Information Sciences Institute Tel.: 310 448 8755; E-mail: cbeal@isi.edu.

the student model component of the AnimalWatch ITS. These analyses indicated that students who received rich multimedia help when they made errors (the Heuristic condition) had higher topic mastery scores than peers who worked with a text only version of the ITS which provided only simple text messages (e.g., "try again") [2].

Outcome variables can provide evidence of learning from an ITS. However, they tell us nothing about the individual student's experience while working with the tutor. Students might reach similar outcome points via quite different sequences of problems, or learning trajectories, some of which might be more effective, efficient or well-matched to particular students. Thus, if our interest is in the process of learning, then we should evaluate the efficacy and other attributes of sequences of problem-solving interactions. The challenge is that, by definition, each student's learning experience with an ITS is unique. For example, the AnimalWatch ITS includes more than 800 word problems, most of which can be customized in real time to individual students. Those who worked with AnimalWatch took unique paths through an extremely large problem space, and each step in their trajectories depended on their prior problem solving history [3].

One approach to evaluating student progress and performance while working with an ITS has been to examine the reduction in the number of errors across sequences of problems involving similar skills [4,5]. Unfortunately, the utility of this approach is often limited due to the lack of sufficient problems of the same type and difficulty that can be used to form meaningful sequences. A more serious problem is that the elements of interactions in a problem sequence are not independent; the next problem a student sees depends on his or her unique learning history. This means that we cannot treat the student's experience as a sample of independent and identically distributed problems, nor can we rely on traditional statistical methods (analysis of variance; regression) that treat it as such [6].

In this paper, we present alternative methods to compare the learning experiences of students, and experimental groups of students. We illustrate these methods with student problem solving data from the AnimalWatch project; however, they are general.

1. Comparing Experiences

The first step is to create a multidimensional representation of the student's experience as a sequence of dependent interactions. For instance, the student might attempt problem 1, fail, get a hint, fail again, get another hint, succeed, and then move onto problem 17, which the tutor judges to the best next problem, given the observed sequence of interactions. Let $S_i = x_1, x_2, \ldots, x_n$ be the sequence of interactions for student i. In general the set of interaction types is quite large; for instance, the AnimalWatch tutor includes 807 problems, each of which is instantiated with a variety of operands; and 47 distinct hint types. Interactions have attributes in addition to their type. They take time, they are more or less challenging to the student, they succeed or fail, and so on. In fact, interaction x_i is a vector of attributes like the one in Figure 1. This is the 5th problem seen by student `x32A4EE6`, it involves adding two integers, it is moderately difficult, it required 142 seconds and one hint to solve correctly, and so on. The experience of a student is represented by a sequence of structures like this one. While our examples all focus on information about problems (topic, difficulty, time), the approach can be generalized to other characterizations of students' experience, such as the frequency and

```
PROBLEM-ID: 675 , NUMBER: 5 , STUDENT: #<STUDENT x32A4EE6> ,
TOPIC: ADDINTEGERS ,  OP1: 8155 , OP2: 2937, DIFFICULTY: 4.33
NUMSKILLS: 2 , TIME-REQUIRED: 142 , NTHINTOPIC: 3 ,
HINTS: (<HINT x3646D76>)
```

Figure 1. A single problem instance presented to a student by AnimalWatch

content of hints. That is, we identity aspects of interaction with the ITS that we want to consider in an evaluation and represent these in the vector x_i.

Although the problem instance in Figure 1 is unique, it belongs to several *problem classes*; for instance, it belongs to the class of `ADD-INTEGERS` problems with `DIFFICULTY = 4.33`. Such *class attributes* define problem classes. Another example is the number of different math skills required to solve problems in the class. Other class attributes are derived from the problem instances in the class. An important derived attribute is *empirical difficulty*, which we define as the number of problems in a class answered incorrectly divided by the total number of attempted problems in that class. In Section 6 we will see that empirical difficulty often differs from a priori estimates by the ITS developers of the difficulty of problems.

Once we have created vectors to represent the elements of interest of the student's interaction with the ITS, we can compare students. We want to perform several kinds of analysis:

- Compare two students' experiences; for example, assess whether one student learns more quickly, or is exposed to a wider range of topics, than another.
- Form clusters of students who have similar experiences; for example, cluster students according to the rates at which they proceed through the curriculum, or according to the topics they find particularly difficult.
- Compare groups of students to see whether their experiences are independent of the grouping variables; for example, tutoring strategies are different if students have significantly different experiences under each strategy.

2. General Method

These kinds of analysis are made possible by the following method. We will assume that each problem instance x seen by a student is a member of exactly one problem class χ.

1. Re-code each student experience $S_i = x_1, x_2, \ldots x_n$ as a sequence of problem classes $\sigma_i = \chi_i, \chi_j, \ldots \chi_m$.
2. Derive one or more functions $\phi(\sigma_i, \sigma_j)$ to compare two problem class sequences (i.e., two students' experiences). Typically, ϕ returns a real-valued number.
3. Students may be grouped into empirical clusters by treating ϕ as a similarity measure. Groups of students (e.g., those in different experimental conditions) can be compared by testing the hypothesis that the variability of ϕ within groups equals the variability between groups.

Expanding on the last step, let G_i be a group comprising n_i sequences of problem classes (one sequence per student), so there are $C_i = (n_i^2 - n_i)/2$ pairwise comparisons of sequences. If we merge groups G_i and G_j, there are $C_{i \cup j} = ((n_i + n_j)^2 - (n_i + n_j))/2$ pairwise comparisons of all sequences.

Let

$$\delta(i) = \sum_{a,b \in G_i} \phi(a,b) \tag{1}$$

be the sum of all pairwise comparisons within group G_i. If groups G_i and G_j are not different, then one would expect

$$\Delta(i,j) = \frac{(\delta(i) + \delta(j))/(C_i + C_j)}{\delta(i \cup j)/C_{i \cup j}} = 1.0 \tag{2}$$

This equation generalizes to multiple groups in the obvious way: If there are no differences between the groups then the average comparison among elements in each group will equal the average comparison among elements of the union of all the groups.

3. Hypothesis Testing by Randomization

We introduce randomization testing for two groups, though it generalizes easily to multiple groups. In the previous section we introduced a test statistic $\Delta(i,j)$ and its expected value under a null hypothesis, but not its sampling distribution. The sampling distribution of a statistic under a null hypothesis H_0 is the distribution of values of the statistic if H_0 is true. Typically H_0 is a statement that two things are equal, for instance, $H_0 : \Delta(i,j) = 1$. If the test statistic has an improbable value according to the sampling distribution then H_0 probably is not true. We reject H_0 and report the probability of the test statistic given H_0 as a *p value*.

Suppose one has a statistic that compares two groups i and j, such as $\Delta(i,j)$ (Eq. 2). Under the null hypothesis that the groups are not different, an element of one group could be swapped for an element of the other without affecting the value of the statistic very much. Indeed, the elements of the groups could be thoroughly shuffled and re-distributed to *pseudosamples* i^* and j^* (ensuring that the pseudosamples have the same sizes as the original samples i and j) and the statistic could be recomputed for the pseudosamples. Repeating this process produces a distribution of *pseudostatistics* which serves as the sampling distribution against which to compare the test statistic.

Randomization is non-parametric, it makes no assumptions about the distributions from which samples are drawn; and it can be used to find sampling distributions for any statistic.

The hypothesis testing procedure for comparing two groups, i and j, of students, then, is to derive the test statistic $\Delta(i,j)$ as described earlier, then throw all the students into a single group, shuffle them, draw pseudosamples i^* and j^*, compute $\Delta^*(i^*,j^*)$ and increment a counter c if $\Delta^*(i^*,j^*) > \Delta(i,j)$. After repeating the process k times, the p value for rejecting the null hypothesis that the groups are equal is c/k.

3.1. About the Implementation

Comparing each student to every other is quadratic, repeating the process for each pseudosample adds a linear factor. Note also that the denominator of Eq. 2 is calculated only once; only the numerator changes when we draw pseudosamples. In practice, one can make the procedure run very fast by not actually drawing pseudosamples from the orig-

inal sample but, rather, shuffling pointers into the original sample. This requires little more space than it takes to store the original samples and keeps the space complexity of the algorithm very low. The analyses in the examples below involve a few dozen students in each of two samples and 1000 pseudosamples, and none takes more than two minutes on a Macintosh G4.

4. Example: Comparing the progress of students in different conditions

Suppose we want to assess the distribution of topics encountered by a student after ten, twenty, ... problems, and compare students to see whether they progress through the topics in the same way. As noted earlier, AnimalWatch presented nine topics. Let $s_{i,t} = n_1, n_2, ...n_9$ represent the number of problems on each of nine topics encountered by student i at time t. Said differently, we imagine the progress of the student at time t as a point in nine-dimensional space. If we measure the progress of the student at regular intervals, we get a trajectory through nine-dimensional space. Two students may be compared by summing the Euclidean distances between corresponding points in this space:

$$\phi(\sigma_a, \sigma_b) = \sum_{t=0,10,20,...} \sqrt{\sum_{i=1,2,...9} (n_{i,a} - n_{i,b})^2} \tag{3}$$

We used the randomization method to compare progress for students in the Text and Heuristic experimental conditions, described earlier. We looked at each student after 10, 20, ..., 90 problems and recorded how many problems on each of nine topics the student solved. Students were compared with the function ϕ in Eq 3. The test statistic $\Delta(Text, Heuristic) = 0.981$ was rejected only twice in 1000 randomization trials, so we can reject the null hypothesis that progress through the nine-topic problem space is the same for students in the Text and Heuristic conditions, with $p = .002$.

It is one thing to test whether student in different experimental groups are different, another to visualize *how* they are different. In the previous example the trajectories are in a nine-dimensional space. However, the progress of each student through this space may be plotted as follows: Let $\mathcal{P}(s, t, c)$ be the proportion of problems in problem class c solved correctly by student s in the first t problems seen by that student. For instance, $\mathcal{P}$(1,30,addintegers)$= .6$ means that of the addintegers problems in the first 30 problems seen by student 1, 60 % were solved correctly. Let $\mathcal{N}(s, t, p)$ denote the number of problem classes for which $\mathcal{P}(s, t, c) > p$. For example, $\mathcal{N}(1, 30, .5) = 2$ means that in the first 30 problems, student 1 encountered two problem classes for which she solved 50% of the problems correctly. Let $V_{\mathcal{N}}(s, p) = [\mathcal{N}(s, 10, p), \mathcal{N}(s, 20, p), \mathcal{N}(s, 30, p)...]$, that is, the sequence of values of $\mathcal{N}$ for student s after $10, 20, 30...$ problems. Such a sequence represents progress for a student in the sense that it tells us how many classes of problems a student has solved to some criterion p after $10, 20, 30...$ problems.

To visualize the progress of a student one may simply plot $V_{\mathcal{N}}(s, p)$, and to compare groups of students one may plot the mean $V_{\mathcal{N}}(s, p)$ for students within groups. This is done in Figure 2. The vertical axis is mean $\mathcal{N}(s, t, p)$ averaged over students in a group, the horizontal axis is t, the number of problems attempted by the students. Here, t ranges from 10 to 100 problems. The higher of the two lines corresponds to the Heuristic

condition, the lower to Text. One sees that on average, a student in the Heuristic condition masters roughly five topics to the criterion level of 50% in the first 100 problems, whereas students in the Text condition master only 3.5 topics to this level in the same number of attempts. These curves also can be compared with our randomization procedure, and are significantly different.

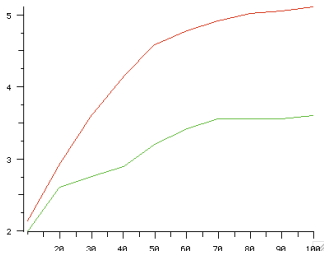

Figure 2. Mean number of problem classes mastered to the 50% criterion level as a function of the number of problems attempted by the students. Upper curve is Heuristic condition, lower is Text.

5. Example: Comparing the distribution of problems seen by students in different conditions

We will use data from the AnimalWatch project to illustrate the approach. Students were taught about nine arithmetic topics. Each student can therefore be represented as a vector of nine numbers, each representing the number of problems on a given topic that the student solved correctly, ordered on the basis of our empirical difficulty measure derived above (although the vector might represent other attributes of interest).

Let $\sigma_m(i)$ be the ith value in the vector for student m. Two students may be compared by

$$\phi(\sigma_m, \sigma_n) = \sum \mathrm{abs}(\sigma_m(i) - \sigma_n(i)) \tag{4}$$

that is, the sum of the absolute differences in the numbers of problems solved correctly on each topic.

In this example, we will compare the learning experiences of students who worked with two different versions of the AnimalWatch ITS: Some students worked with a version that provided only minimal, text-based help in response to errors (Text). Other students worked with a version that provided students with rich, multimedia hints and explanations (Heuristic). Figure 3 shows the mean number of problems on each topic solved by students in the Text and Heuristic conditions, with 95% confidence intervals around the means. One might be tempted to run a two-way analysis of variance on these data with Topic and Condition as factors, but remember that the problems seen by a student are not independent, the tutor constructed a unique sequence of problems for each student, and the cell sizes are quite unequal, all of which violate assumptions of the analysis of variance. The randomization method makes no such assumptions. We compared the Text and Heuristic conditions with the randomization procedure described earlier. The test statistic

$\Delta(Text, Heuristic) = .963$ was exceeded in every one of 1000 randomization trials, so we can reject the null hypothesis that the conditions are equal with $p < .001$. Thus, we conclude that, even though students had unique experiences with the ITS, those who received multimedia help in response to errors solved more problems correctly, across all topics, relative to students who received only limited, text-based help.

The total number problems solved by students was not the same in the Text and Heuristic conditions. This might account for the significant result. We can run the analysis differently, asking of each student what fraction of the problems she saw in each problem class she answered correctly. In this case we are comparing probabilities of correct responses, not raw numbers of correct responses. Repeating the randomization procedure with this new function for comparing students still yields a significant result, albeit less extreme: The test statistic $\Delta(Text, Heuristic) = .973$ was exceeded in 950 of 1000 trials, for a p value of 0.05.

By contrast, the p value for a comparison of girls and boys was 0.49, there is no reason to reject the null hypothesis that girls and boys correctly solved the same numbers of problems on all topics.

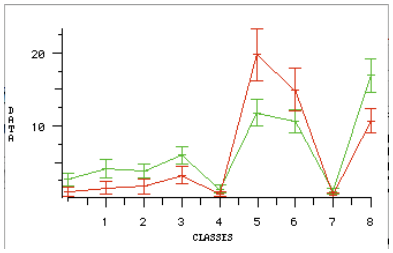

Figure 3. Mean correct number of problems for Heuristic and Text conditions.

6. Example: Change in Empirical Difficulty

As a final example of methods for comparing student experiences, we return to the idea of empirical difficulty, introduced in Section 1. We define the empirical difficulty of a problem as the number of unsuccessful attempts to solve it divided by the total number of attempts to solve it. Figure 4 shows the empirical difficulty of the nth problem for the Heuristic and Text groups. That is, the horizontal axis represents where a problem is encountered in a sequence of problems, the vertical axis represents the proportion of attempts to solve that problem which failed. Regression lines are shown for the Heuristic and Text groups. It appears that the empirical difficulty of problems in the Heuristic group is lower than that of the Text group, or, said differently, Heuristic students solved a higher proportion of problems they encountered. This appears to be true wherever the problems were encountered during the students' experience.

We can test this hypothesis easily by randomizing the group to which students belong to get a sampling distribution of mean empirical problem difficulty. This result is highly significant: In 1000 randomized pseudosamples the mean difference in problem difficulty between Heuristic and Text, 0.094, was never exceeded. One also can random-

ize the group to which students belong to get a p value for the difference between the slopes of the regression lines. This p value is $.495$, so there is no reason to reject the hypothesis that the regression lines have equal slope. In other words, the change in empirical problem difficulty as a function of when the problem is encountered, a slightly positive relationship, is the same for Heuristic and Text students.

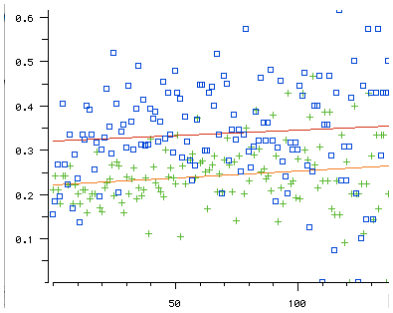

Figure 4. Empirical problem difficulty as a function of when problems are encountered.

In conclusion, we demonstrated that students' experiences with an ITS are sequences of multidimensional, dependent observations, and yet they are not beyond the reach of statistical analysis. We showed how to represent students' learning trajectories and how to test hypotheses about them with randomization methods.

Acknowledgments

We thank Dr. Ivon Arroyo for her work on constructing the AnimalWatch dataset. We also thank the students, parents, and staff of the schools that participated in the AnimalWatch project. The original AnimalWatch project was supported by National Science Foundation HRD 9714757. Preparation of this paper was supported by National Science Foundation REC 0411886 and HRD 0411532.

References

[1] Beal, C. R., & Arroyo, I. (2002). & The AnimalWatch project: Creating an intelligent computer mathematics tutor. In S. Calvert, A. Jordan, & R. Cocking (Eds.), Children in the digital age (pp. 183-198).

[2] Beck, J., Arroyo, I., Woolf, B., & Beal, C. R. (1999). An ablative evaluation. In Proceedings of the 9th International Conference on Artificial Intelligence, pp. 611-613, Paris: ISO Press.

[3] Beck, J. E., Woolf, B. P., & Beal, C. R. (2000). Learning to teach: A machine learning architecture for intelligent tutor construction. Proceedings of the Seventeenth National Conference on Artificial Intelligence, Austin TX.

[4] Arroyo, I. (2003). Quantitative evaluation of gender differences, cognitive development differences, and software effectiveness for an elementary mathematics intelligent tutoring system. Doctoral dissertation, University of Massachusetts at Amherst.

[5] Mitrovic, A., Martin, B., & Mayo, M. (2002). Using evaluation to shape ITS design: Results and experiences with SQL Tutor. Using Modeling and User Adapted Instruction, 12, 243-279.

[6] Cohen, P. R. (1995). Empirical methods for artificial intelligence. Cambridge MA: MIT Press.