Using Data Mining Results to Improve Educational Video Game Design

Deirdre Kerr
National Center for Research on Evaluation, Standards, and Student Testing[1]
University of California, Los Angeles
dkerr@ets.org

This study uses information about in-game strategy use, identified through cluster analysis of actions in an educational video game, to make data-driven modifications to the game in order to reduce construct-irrelevant behavior. The examination of student strategies identified through cluster analysis indicated that (a) it was common for students to pass certain levels using incorrect mathematical strategies and (b) throughout the game a large number of students used order-based strategies to solve problems rather than strategies based on mathematics, making measurement of their mathematical ability difficult. To address the construct irrelevant variance produced by these issues, two minor changes were made to the game and students were randomly assigned to either the original version or the revised version. Students who played the revised version (a) solved levels using incorrect mathematical strategies significantly less often and (b) used order-based strategies significantly less often than students who played the original version. Additionally, student perception of the revised version of the game was more positive than student perception of the original version, though there were no significant differences in either in-game or paper-and-pencil posttest performance. These findings indicate that data mining results can be used to make targeted modifications to a game that increased the interpretability of the resulting data without negatively impacting student perception or performance.

1. INTRODUCTION

In educational video games or simulations, relevant features of student performance must be extracted from the log files that are automatically generated by the game or simulation as students play [Kim, Gunn, Schuh, Phillips, Pagulayan, and Wixon 2008]. These log files store complete student answers to the problems given in the game, including information regarding the strategies and mistakes each student employed while attempting to solve each problem [Merceron and Yacef 2004], in order to capture information about student strategies that can be impossible to capture in a more traditional test [Rahkila and Karjalainen 1999]. Logging student actions allows the researcher to record students' exact learning behavior [Romero and Ventura 2007] that is not always captured in written or verbal explanations of their thought processes [Bejar 1984], without interrupting the flow of student work [Kim et al. 2008; Mostow, Beck, Cuneao, Gouvea, Heiner, and Juarez 2011].

Though log data are more comprehensive and more detailed than most other forms of assessment data, the inclusion of such fine-grained detail presents a number of problems for analysis [Garcia, Romero, Ventura, de Castro, and Calders 2011; Mostow et al. 2011]. First, log files contain large quantities of information, typically consisting of thousands of pieces of information on each subject [Romero, Gonzalez, Ventura, del Jesus, and Herrera 2009], with a

[1] Deirdre Kerr is now at Educational Testing Service.

single subject able to generate over 3,000 actions in just half an hour of game play [Chung et al. 2010]. Second, the data are at such a small grain size (e.g., "selected Tool A", "moved game character to the left", etc.) that there is often no known theory to help identify precisely which actions are important to the construct being measured [National Research Council 2011]. Third, examining fine-grained behaviors in highly unstructured environments such as educational video games and simulations creates some inherent problems [Amershi and Conati 2011], in that there is often little overlap in the thousands of actions produced by one subject and the thousands of actions produced by a second subject. Therefore, log data are sparse (in that any given subject produces a lot of actions, but any given action may only be produced by a few subjects), noisy (in that irrelevant actions can vastly outnumber relevant ones, and relevant actions are not usually identifiable a priori), and so large that it is prohibitively costly to examine the data by hand.

In datasets too large to analyze by hand, data mining techniques are ideal for automatically identifying and describing meaningful patterns despite the noise surrounding them [Bonchi et al. 2001; Frawley, Piateski-Shapiro, and Matheus 1992]. This automatic extraction of implicit, interesting patterns from large, noisy datasets can lead to the discovery of new knowledge about how students solve problems in order to identify interesting or unexpected learning patterns [Romero et al. 2009)] and can allow questions to be addressed that were not previously feasible to answer [Romero, Ventura, Pechenizkiy, and Baker 2011].

One of the more common educational data mining techniques is cluster analysis [Castro, Vellido, Nebot, and Mugica 2007]. Cluster analysis is a density estimation technique for identifying patterns within user actions reflecting differences in underlying attitudes, thought processes, or behaviors [Berkhin 2006; Romero et al. 2009] through the analysis of either general correlations or sequential correlations [Bonchi et al. 2001].

Cluster analysis has been used in computerized educational environments to identify aspects of student motivation [Hershkovitz and Nachmias 2011], collaborative vs. solitary work patterns [Rodrigo, Anglo, Sugay, and Baker 2008], processes of inquiry [Buckley, Gobert, and Horwitz 1999], and approaches to teaching and learning [Trigwell, Prosser, and Waterhouse 1999]. Cluster analysis has also been used to identify different player types [Etheredge, Lopes, and Bidarra 2013], specific usage patterns [Harpstead et al. 2013; Romero et al. 2009] and error patterns [Sison, Numao, and Shimura 2000], circumstances under which students accessed help [Nilakant and Mitovic 2005], and suboptimal learning behaviors [Amershi and Conati 2011].

While educational data mining techniques such as cluster analysis have been used to provide new insights on how students behave in a number of different computerized educational environments, fewer studies have capitalized on those insights in order to redesign the environment [Cen, Koedinger, and Junker 2007]. For example, Kim et al. [2008] modified a first-person shooter to address areas in the game with abnormally high rates of player death and found that the modifications increased both player performance and player engagement. Baker et al. [2006] modified an intelligent tutoring system to reduce gaming behavior that was circumventing learning and found that the modifications reduced gaming behavior but did not significantly impact learning. Beck and Rodrigo [2014] modified an intelligent tutoring system to reduce wheel spinning behaviors and found that the modifications did not successfully reduce the behavior. Cen et al. [2007] modified an intelligent tutoring system to reduce overpractice of skills that students had already learned and found that the modifications led to a significantly reduced amount of time necessary to learn the same amount of information.

More such studies are necessary if games are to be used as assessments, because game level design is often not as purposeful as test item design and standard methods of determining game level quality from a measurement standpoint have not yet been developed. There is currently

much less knowledge about the impact of game design decisions on the measurement properties of a given educational game than there is about the impact of test item design decisions on the measurement properties of a given exam. This is a particularly important concern not just because measurement properties of games need to be known if they are to be used as stand-alone assessments of student knowledge, but also because students behave differently in games than they do in standard testing formats.

Because they are less predictable environments than most standard testing formats [Rupp, Gushta, Mislevy, and Shaffer 2010], students often engage in behaviors in educational video games that are difficult to interpret in the context of the content being assessed by the game. An evaluation of the instructional procedures used in the game is often necessary to identify the source of such behavior in order to gain a more accurate measure of student knowledge [Jitendra and Kameenui 1996]. The alternative of adding additional assessment items to the game to increase interpretability of student behavior is often not feasible because it may disrupt the flow of the game or produce constraints that are unnatural in the game environment [Rupp et al. 2010].

Therefore, purposeful modification of the design of an educational game based on the results of previous analyses may be necessary in order to increase the degree to which the game encourages students to think about the targeted educational content [Dolmans, Gijselaers, Schmidt, and Van Der Meer 1993; Fisch 2005], and thus the degree to which student in-game behavior is interpretable in the context of the content being assessed by the game. To address this need, this study examines the impact of modifications to an educational video game on the interpretability of student in-game behavior.

2. PURPOSE

In order to determine whether data-driven modifications to an educational video game resulted in increased interpretability of student in-game behavior, this study compared two versions of the educational video game *Save Patch*: the game as it was originally intended to be distributed to students and a revised version of the game that incorporated small changes suggested by data mining results from a previous study of the original version. Students were randomly assigned within-class to either the original version or the revised version to determine the impact of the data-driven revisions. The primary research question addressed in this study was:

1. Does revising an educational video game based on data mining results reduce the identified construct irrelevant in-game behavior?

Even though the data driven revisions were designed solely to reduce specific construct irrelevant behavior, it is possible that the revisions might have secondary effects. Reducing construct irrelevant behavior could indirectly increase learning, since students might be forced to use more mathematical thinking in their game play. On the other hand, student perception of the game might be negatively impacted since certain game behaviors would no longer be allowed, particularly if the targeted behaviors were part of what made the game enjoyable in the original version. Therefore, the following secondary research questions were also addressed in this study:

2. Does revising an educational video game based on data-mining results positively impact student performance either in-game or on paper-and-pencil posttest measures?
3. Does revising an educational video game based on data-mining results negatively impact student perception of the game?

3. METHODS

3.1. GAME DESIGN

The educational video game used in this study is *Save Patch*, a game designed by the National Center for Research on Evaluation, Standards, and Student Testing (CRESST) at the University of California, Los Angeles, and the Game Innovation Lab at the University of Southern California[2]. The activity selection process for *Save Patch* was driven by the findings of the National Mathematics Advisory Panel that fluency with fractions is critical to performance in algebra [National Mathematics Advisory Panel 2008], which is, in turn, of central importance to performance and participation in science, technology, engineering, and math courses and careers [Malcom, Chubin, and Jesse 2004]. Additionally, the understanding of fractions is one of the most difficult mathematical concepts students learn before algebra [National Council of Teachers of Mathematics 2000; Siebert and Gaskin 2006] and misconceptions about the meaning of fractions are not only very common but are also associated with subsequent difficulty understanding and applying advanced mathematical concepts [Carpenter, Fennema, Franke, Levi, and Empson 2000; McNeil and Alibali 2005].

Table 1: Knowledge Specifications for *Save Patch*

1.0. Does the student understand the meaning and importance of the whole unit?
1.1. The size of a rational number is relative to how the whole unit is defined.
1.2. In mathematics, a whole unit is understood to be of some quantity.
1.3. The whole unit can be represented as an interval on the number line.
2.0. Does the student understand the meaning of addition as applied to fractions?
2.1. To add quantities, the units or parts of units must be identical.
2.2. Identical units can be added to create a single numerical sum.
2.3. Dissimilar quantities cannot be represented as a single sum.
3.0. Does the student understand the meaning of the denominator in a fraction?
3.1. The denominator of a fraction represents the number of identical parts in a whole unit.
3.2. As the denominator gets larger, the size of each fractional part gets smaller.
3.3. As the fractional part size gets smaller, the number of pieces in the whole gets larger.
4.0. Does the student understand the meaning of the numerator in a fraction?
4.1. The numerator of a fraction represents the number of identical parts that are combined.
4.2. If the numerator is smaller than the denominator, the fraction represents less than a whole.
4.3. If the numerator equals the denominator, the fraction represents a whole unit.
4.4. If the numerator is larger than the denominator, the fraction represents more than a whole.

A number line representation of fractions, rather than the area model, was chosen for the game because, even though students initially have more difficulty understanding this representation, using a number line representation while teaching fractions is thought to lead to a more complete mental model for students [Bright, Behr, Post, and Wachsmuth 1988; Saxe et al. 2007]. The most important concepts involved in fractions knowledge in this model were analyzed and distilled into a set of knowledge specifications delineating precisely what students were expected to learn in the game [Vendlinski, Delacruz, Buschang, Chung, and Baker 2010].

[2] Save Patch is available online at: http://cats.cse.ucla.edu/games/

The four main concepts to be addressed in the game included the meaning of the unit, the meaning of addition as applied to fractions, the meaning of the denominator, and the meaning of the numerator. Each of these concepts was broken down into further specifications of what understanding of that concept would entail, as seen in Table 1.

These knowledge specifications were the driving force behind game design decisions. For instance, the game area was represented as a line in one-dimensional levels and a grid in two-dimensional levels to reinforce the idea that a unit can be represented as one whole interval on a number line (Knowledge Specification 1.3). Units were represented as blue ropes on a dirt path with small red posts indicating the fractional pieces the unit was broken into (see Figure 1). Students were given ropes in the resource bin on the left side of the game screen labeled "Path Options" and had to break the ropes they were given into the fractional pieces indicated in the level and place the correct number of unit fractions (fractions with a numerator of one) on each sign to guide their character safely to the cage to unlock the prize.

Figure 1: Screen shot from *Save Patch*.

A successful solution to the level shown in Figure 1 would proceed as follows: First the student would click on the down arrow next to one of the whole unit ropes in the resource bin labeled "Path Options" on the left side of the screen. The first click would change the whole unit rope to a rope consisting of two 1/2 pieces, and the second click would change the whole unit rope to a rope consisting of three 1/3 pieces. Then the student would select one of the 1/3 pieces and drag it to the leftmost sign on the game grid. This would result in a value of 1/3 being displayed on the sign. The student would then drag an additional 1/3 piece to the sign, resulting in a value of 2/3, and then drag a final 1/3 piece to the sign, resulting in a value of 3/3. The student would then click on the down arrow next to one of the other whole unit ropes twice to change the whole unit to three 1/3 pieces, select one of the 1/3 pieces and drag it to the second

sign, and select another 1/3 piece and drag it to the third sign. This would result in a value of 3/3 on the first sign, 1/3 on the second sign, and 1/3 on the third sign.

The student would then click on the "Go" button and the character would read the first sign and walk a distance of 3/3 to the right, as indicated on the sign. Since that action would place the character at the second sign, the character would read that sign and walk a distance of 1/3 to the right. This would place the character at the third sign, so the character would read that sign and walk another 1/3 distance to the right. This would place the character at the cage, which would open and release the prize, allowing the student to progress to the next level.

If, after walking the distance indicated on a given sign, the character did not arrive at either another sign or the cage containing the prize it would result in a failure state and the student would be required to retry the level. Note that any equivalent distance on a given sign would result in the same outcome, so using a whole unit rope on the first sign instead of three 1/3 ropes would also result in a correct solution.

Successful game play was intended to require students to determine the unit size for a given grid as well as the size of the fractional pieces making up each unit. The distance the character moved was a function of the number and size of rope pieces placed on each sign, where one whole rope represented a whole unit on the grid and each whole rope could easily be broken into fractional pieces of the desired size by clicking on the arrows next to the rope in the resource bin on the left side of the game screen. Therefore, a successful solution to a given level should indicate a solid understanding of the knowledge specifications underlying the game presentation.

This design allowed students to demonstrate knowledge of the meaning of the denominator of a fraction (Knowledge Specification 3.0) by choosing which fractional pieces to break each whole unit rope into and the meaning of the numerator of a fraction (Knowledge Specification 4.0) by choosing how many unit fractions to place on each sign. Additionally, a number of levels in the game were designed to represent more than one unit, allowing students to demonstrate knowledge of the meaning and importance of the whole unit (Knowledge Specification 1.0).

Game play was constrained so that it was not possible to add two numbers with different denominators (Knowledge Specifications 2.1 and 2.3), rather than allowing the students to make the addition and having the game calculate the resulting distance. This meant that the game did not allow students to add 1/2 to 1/3, instead forcing students to scroll the 1/2 rope to 3/6 and the 1/3 rope to 2/6 before allowing them to be added together. For the same reason, the game did not allow the creation of mixed numbers (e.g., 1½), forcing players to convert the whole number portion of the mixed number into the appropriate fractional representation (e.g., 3/2) before adding the fractional portion to the whole number portion.

In order to scaffold the knowledge specifications and provide a logical progression through the game [Rupp et al. 2010], *Save Patch* was broken into six stages. The first stage was designed to introduce students to the game mechanics in a mathematical setting they were comfortable with, and therefore included only whole number distances between signs. The second stage introduced fractions via unit fractions, requiring students to identify the denominator while restricting the numerator to one (i.e., the distance between signs was always 1/x, never 2/x, 3/x, etc.). The third stage combined concepts from the first two stages, with at least one distance in each level representing a unit fraction (1/x) and at least one other distance representing a whole unit (x/x). The fourth stage was similar to the third stage, except that the distance representing a whole unit did not start and end on unit markers. Instead, the whole unit distance spanned a unit marker (e.g., extending from 1/3 to 4/3). The fifth stage was when students were first asked to identify the numerator as well as the denominator of a fraction and dealt with proper fractions

wherein the numerator was greater than one but smaller than the denominator. The sixth stage completed the identification of fractions concepts by asking students to identify improper fractions, wherein the numerator was greater than the denominator.

3.2. SUMMARY OF THE DATA MINING PROCESS

The cluster analysis process that was used to identify the data-driven suggestions for revisions to *Save Patch* was run over data from a previous study and followed the process explained in detail in Kerr and Chung [2012] in an earlier issue of this journal. A summary of that process follows.

To record student actions reflecting understanding of the knowledge specifications, each action taken by a student while playing the game was automatically recorded and stored in the form of a structured log written to a tab-delimited text file. As delineated in Chung and Kerr [2012], each entry in the log file included general information about the type of action performed, specific information about the exact parameters of the action, and relevant contextual information.

The actions in each level of the game were transformed into a sparse binary matrix wherein each row represented a single student's attempt to solve the level (where an attempt was defined as the set of actions beginning immediately after either a new level load or a level reset and ending at either a level reset or a level completion) and each column represented a possible action that could be taken in that level. 1's indicated actions made in a given attempt and 0's indicated actions not made in that attempt. Since most students made multiple attempts to solve a given level, most students were represented in multiple rows in each matrix. The actions in each matrix were then clustered using the fuzzy cluster analysis algorithm *fanny* [Maechler 2012] in *R* [R Development Core Team 2010] to determine which actions frequently co-occurred.

Fuzzy clustering was chosen because data from *Save Patch* was known apriori to be fuzzy. In cluster analysis, data is considered to be "fuzzy" if a number of points belong to more than one cluster. For example, if one were to cluster the features of shapes (e.g., the number of sides, degree of angles, etc.), a number of the features that belong to the "square" cluster would also belong to the "rectangle" cluster because those shapes (and, therefore, their clusters) are similar. Fuzzy clustering allows for overlapping clusters by assigning each point a probability of belonging to each cluster. If standard cluster analysis algorithms are used on fuzzy data, one of two things will happen. Either the fuzzy clusters will be merged into a single, well-defined cluster (e.g., the "square/rectangle" cluster) or the fuzzy points will be subsumed into whichever cluster has a larger *n* (e.g., the "square" cluster), resulting in a poor fit for the larger cluster and a smaller cluster of the remaining non-overlapping points (e.g., a "rectangle without all the features" cluster).

The level pictured in Figure 1 can be used to demonstrate why data from *Save Patch* was known apriori to be fuzzy. Because the first distance represented in the level could be seen as either a whole unit or a fractional distance, there were two different valid solutions to this level: the fractional solution of 3/3, 1/3, 1/3 and the whole unit solution of 1/1, 1/3, 1/3. The actions necessary to complete the level using either the fractional solution or the whole unit solution are listed in Table 2, along with indicators of which actions were used in which solution. There were a total of 13 actions in the fractional solution and 8 actions in the whole unit solution. However, 5 of these actions were used in both solutions, indicating that the data (at least in regards to the two solution strategies known apriori) was quite fuzzy. It was, therefore, considered likely that different strategies for solving levels in *Save Patch* would frequently share common actions,

and as a result fuzzy clustering was used to allow for the differentiation of such strategies and to assure that the whole unit solution would form a cluster of its own, separate from the fractional solution.

Table 2: Actions Present in the Factional Solution and Whole Unit Solution

Possible Correct Actions	Present in Fractional Solution	Present in Whole Unit Solution
Cut 1/1 into 3/3	✓	
Select 1/3	✓	
Drag 1/3 to Sign 1 Resulting in 1/3	✓	
Select 1/3	✓	
Drag 1/3 to Sign 1 Resulting in 2/3	✓	
Select 1/3	✓	
Drag 1/3 to Sign 1 Resulting in 3/3	✓	
Select 1/1		✓
Drag 1/1 to Sign 1 Resulting in 1/1		✓
Cut 1/1 into 3/3	✓	✓
Select 1/3	✓	✓
Drag 1/3 to Sign 2 Resulting in 1/3	✓	✓
Select 1/3	✓	✓
Drag 1/3 to Sign 3 Resulting in 1/3	✓	✓
Submit 3/3, 1/3, 1/3	✓	
Submit 1/1, 1/3, 1/3		✓

The final number of clusters for each level was chosen by inspection, iterating the number of clusters until actions known to be part of the fractional solution or whole unit solution began appearing in clusters in which they did not belong. This process resulted in the identification of frequent action sets in each game level representing distinct, nameable strategies. These strategies included multiple valid solution strategies, distinct mathematic misconceptions, and misapplications of game mechanics, as detailed in Kerr and Chung [2012].

For example, one of the clusters identified for the level pictured in Figure 1 included the actions “Drag 1/2 to Sign 1 Resulting in 3/2”, “Drag 1/2 to Sign 2 Resulting in 1/2”, “Drag 1/2 to Sign 3 Resulting in 1/2”, and “Submit 3/2, 1/2, 1/2”. This cluster was determined to represent a *partitioning error*, since the actions making up the cluster were consistent with what students would do if they were incorrectly partitioning the fraction by counting the number of dividing marks, rather than the number of spaces, to determine the denominator of the represented fraction. Further evidence that the cluster represented a *partitioning error* was found in Kerr [2014], wherein a significant number of students who were determined to have made a partitioning error specifically stated in their written explanation of their answer that they counted marks or lines to determine the denominator.

3.3. IDENTIFYING DATA-DRIVEN SUGGESTIONS

While most of the identified strategies allowed inferences (either positive or negative) to be drawn regarding students' levels of mathematical understanding and/or understanding of game mechanics, two strategies consisted entirely of construct irrelevant behavior. These strategies provided no information about students' understanding of the mathematical concepts covered in the game, nor did they provide indications that students were having difficulty grasping the mechanics of the game.

The first such strategy occurred whenever students used the *order-based* strategy to attempt to solve levels in the game. In this strategy, instead of using mathematical reasoning to determine the value of ropes to place on each sign, students placed the ropes in the order in which they were presented in the resource bin. For example, if the level shown in Figure 1 had a 1/2 rope, followed by a 1/3 rope, followed by two 1/5 ropes as resources in the resource bin (rather than the three whole units pictured), students who were using this strategy would place 1/2 on the first sign, 1/3 on the second sign, and 2/5 on the third sign. Similar to students filling in bubbles in a multiple-choice test to form a smiley face or other such pattern, this strategy provided no information about student mathematical understanding.

The second such strategy was the result of a flaw in the design of the game, and only occurred when a specific (unfortunately common) strategy was combined with a specific type of level design. The strategy involved was the strategy indicating a *partitioning error*, described earlier, wherein students determined the denominator of a fraction by counting the number of dividing marks rather than the number of spaces. In certain levels, the final destination was on a whole unit marker and there was another sign one space to the left, as in the example shown in Figure 2.

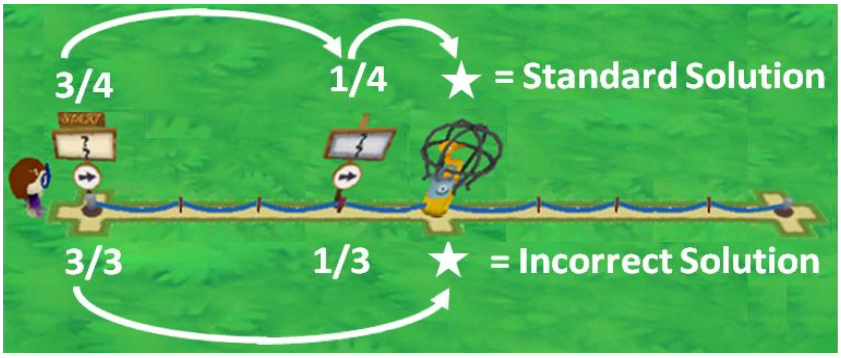

Figure 2: An example of a level that can be solved using incorrect mathematical strategies.

Students using the *partitioning error* strategy in the level represented in Figure 2 would see the level as being in thirds rather than fourths and place 3/3 on the first sign and 1/3 on the second sign. Even though this is incorrect, 3/3 on the first sign would result in the character walking right past the second sign and going directly to the prize located at 4/4, since 3/3 and 4/4 are equivalent distances. Note that the placement of 1/3 on the second sign indicates that these students were probably assuming that the game character would stop at the second sign, providing some evidence that they were operating under this specific mathematical misconception rather than deliberately providing an equivalent fraction.

This oversight in game design meant not only that these student were allowed to solve certain game levels using a mathematically incorrect strategy, but that they were provided with

congratulatory feedback and advanced to the next level, despite their error. Advancing to the next level meant that these students didn't get a chance to learn from their mistakes, and may actually have reinforced misconceptions they already had. It also essentially created missing information for that student in that level, because a student solving the level incorrectly with this strategy was not forced to replay the level until it was solved correctly (as was required for all other students).

3.4. STUDY DESIGN

To minimize the in-game behavior that produced construct irrelevant invariance, two important changes were made to the game (see Figure 3). First, keys were added to some signs in the game. This made the affected signs mandatory, so that students could not solve levels incorrectly by skipping over those signs. Secondly, available resources at the start of each level were changed from fractional pieces to whole unit pieces. This was intended to make the *order-based* strategy a less attractive option, since the desired fraction was clearly not already listed in the resource bin (as the bin no longer contained any fractional values). The revised version of the game differed from the original version of the game only in these two respects. All other aspects of the game, including level design and in-game feedback, were identical in both versions.

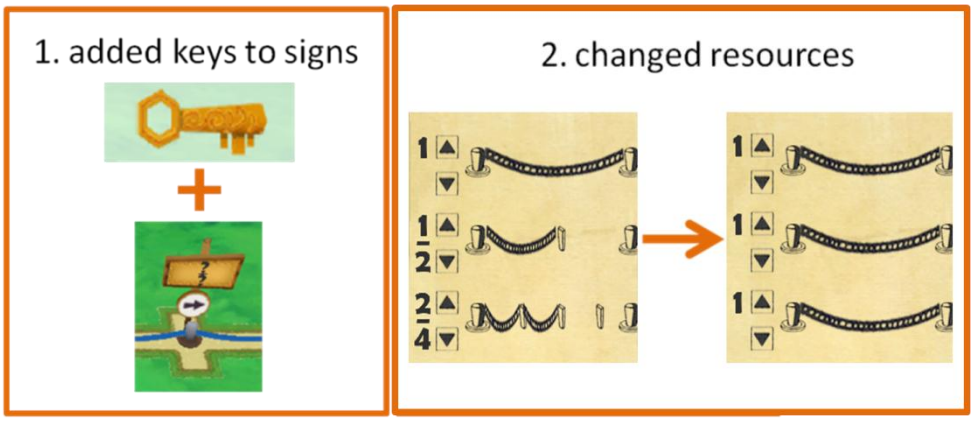

Figure 3: Data-driven revisions to *Save Patch*.

In this study, 62 sixth-grade students from two suburban schools in southern California were randomly assigned within class to either the original version of the game (if they received an odd-numbered ID) or the revised version of the game (if they received an even-numbered ID), resulting in 31 students in each condition. The students were fairly randomly distributed in terms of gender, with 24 males, 30 females, and 8 unreported.

Students took a fractions pretest on the first day of the study. Following the pretest, students played *Save Patch* approximately 40 minutes a day for four subsequent days. Students took an immediate posttest on the last day of game play and a delayed posttest approximately three weeks later. The immediate posttest also included a survey regarding student perception of the game.

3.5. DETERMINING THE IMPACT OF THE REVISIONS

To determine the impact of the replacement of fractional resources with whole unit resources and the addition of keys as a mechanic on construct irrelevant in-game behavior,

student performance, and student perception, three MANCOVAs were run. All analyses used pretest score as the independent variable and game version as the factor. The analysis examining the effect on construct irrelevant behavior used the number of levels solved incorrectly and the percentage of attempts utilizing guessing strategies as dependent variables.

The analysis examining the effect on student performance used two in-game measures of performance (the number of attempts per level and the percentage of first attempts that were solutions) and two paper-and-pencil measures of performance (immediate posttest score and delayed posttest score) as dependent variables. Both paper-and-pencil measures included some near-transfer items (e.g., identifying a position on a number line) and some far-transfer items (e.g., adding two fractions together).

The analysis examining the effect on student perception used a measure of student perception of the game (see Table 3) developed internally from earlier studies on student perception of a variety of educational video games.

Table 3: Measure of Student Perception

	I Disagree	I Disagree A Little	I Agree A Little	I Agree
The game was boring	0	1	2	3
I got into the game	0	1	2	3
I wanted to play the game longer	0	1	2	3
Beating levels in the game made me feel good	0	1	2	3
I learned from the game	0	1	2	3

4. RESULTS

Revising the game significantly reduced construct irrelevant in-game behavior (see Table 4). Controlling for pretest score, students playing the revised version of the game solved fewer levels using mathematically incorrect strategies, with students in the original version solving an average of 1.33 levels incorrectly and students in the revised version solving only an average of 0.08 levels incorrectly ($p < .001$, Cohen's $d = 1.18$). Students playing the revised version also used order-based strategies instead of mathematical strategies less often, with students in the original version using those strategies 16% of the time and students in the revised version using them only 10% of the time ($p = .001$, Cohen's $d = 0.77$).

The revisions to the game had no impact on student performance either in the game or on posttest measures of fractions knowledge (see Table 4). Controlling for pretest scores, students playing the revised version did not solve levels in fewer attempts ($p = .316$) or solve levels on their first attempt more often ($p = .419$) than students playing the original version. They also did not have significantly higher immediate posttest scores ($p = .576$) or delayed posttest scores ($p = .725$) than students playing the original version.

However, when broken out by stage, the mean number of attempts per level dropped from 4.19 in the original version to 2.38 in the revised version in Stage 4 ($p = .024$, Cohen's $d = 0.58$). The mean number of attempts per level also dropped from 3.07 to 2.14 in Stage 5 ($p = .022$, Cohen's $d = 0.61$). Additionally, students in the revised version solved an average of 2.45 levels of the four levels in Stage 5 on their first attempt while students in the original version only

solved an average of 1.56 levels on their first attempt (p = .005, Cohen's d = 0.75). There were no significant differences in other stages in either the number of levels solved on the first attempt or the number of attempts per level.

Table 4: Impact of Data-Driven Revisions

Research question	Measures	Original	Revised	p	d
Do data-driven revisions to an educational video game reduce construct irrelevant in-game behavior?	Levels solved incorrectly	1.33	0.08	< .001	1.18
	Percent order-based	16%	10%	.001	0.77
Do data-driven revisions to an educational video game positively impact student performance?	Attempts per level	2.88	2.62	.316	na
	Solved on first attempt	11%	12%	.419	na
	Immediate posttest	3.83	3.97	.576	na
	Delayed posttest	8.05	7.61	.725	na
Do data-driven revisions to an educational video game negatively impact student perception of the game?	Game was boring	1.69	0.91	.011	0.67
	Got into the game	1.43	2.26	.012	0.72
	Wanted to play longer	1.42	2.10	.048	0.56
	Felt good beating levels	2.01	2.23	.426	na
	Learned from the game	1.94	2.44	.105	na

Revising the game did, however, have a significant impact on student perception of the game, though not in the direction anticipated (see Table 4). Controlling for pretest scores, students playing the revised version of the game thought the game was less boring (p = .011, Cohen's d = 0.67), got into the game more (p = .012, Cohen's d = 0.72), and were more likely to want to continue playing the game after four days of game play (p = .048, Cohen's d = .056). The revisions did not significantly impact how good beating levels in the game made students feel (p = .426) or how much students felt they learned from the game (p = .105).

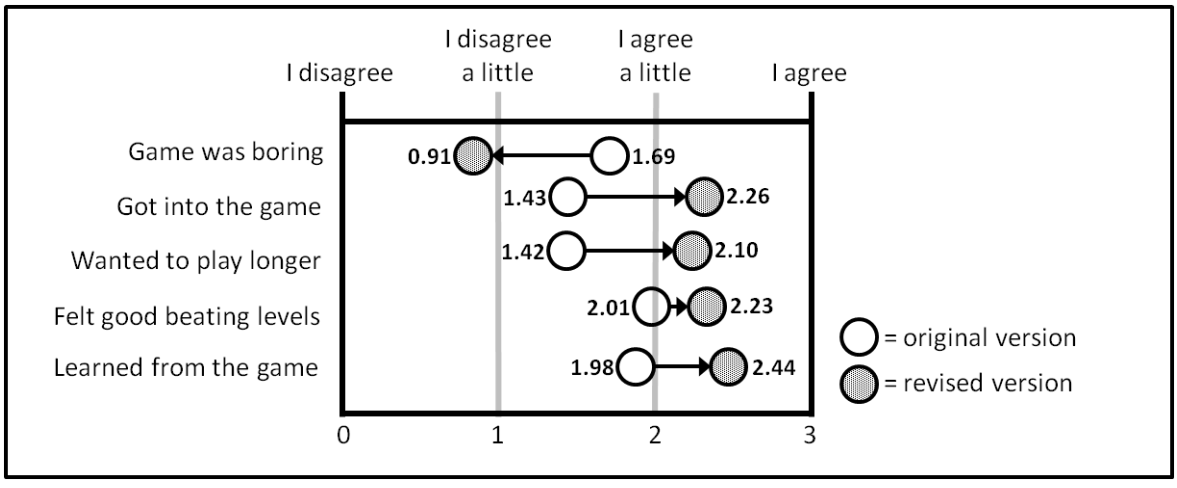

Figure 4: Differences in perception between the original and revised versions.

These differences were substantively meaningful, not just statistically significant. As can be seen in Figure 4, the mean values for all perception measures for the original version of the game were ambivalent, lying between "I disagree a little" and "I agree a little," whereas the mean values for all perception measures for the revised version lie between "I agree a little" and "I agree" (or "I disagree a little" and "I disagree" for the negatively phrased item).

In sum, revising the game based on data mining results significantly decreased construct irrelevant in-game behaviors that were reducing the interpretability of student actions, and did so without negatively impacting student perception of the game. In fact, student perception of the revised version of the game was significantly more positive than student perception of the original version, even though the revisions were designed to curtail gaming behavior that students may have found entertaining. However, the decrease in construct irrelevant in-game behavior did not, in and of itself, lead to a corresponding increase in either in-game or paper-and-pencil posttest performance.

5. DISCUSSION

These results indicate that revising an educational video game based on data mining results can significantly decrease construct irrelevant in-game behavior, making it easier to measure student understanding of the targeted content. The two minor revisions to the game in this study almost completely eliminated students' ability to solve levels using incorrect mathematical strategies and reduced guessing strategies by 42%. Together these two changes reduced the percentage of game data that consisted of construct irrelevant information from 23% to 10%. Interestingly, the changes also had a positive impact on student perception on almost all measures, with students reporting a more positive perception when their ability to game the system was reduced.

However, there was no overall difference in student performance between the two versions of the game. Similar to the findings of Baker et al. [2006], revisions that reduced gaming behavior were not found to significantly increase learning. This may be because students who are prone to gaming behavior in a specific environment do so because they do not have a good understanding of the underlying educational content. For such students, the reduction of gaming behavior would result in a corresponding increase in the use of strategies corresponding to specific mathematical misconceptions (which would not result in students solving levels in fewer attempts). Only in the case of students who had a good understanding of the underlying educational content, but chose to use gaming strategies despite knowing the math, would one see a corresponding increase in the use of correct solution strategies (which would result in students solving levels in fewer attempts).

After some time spent in the game, one might posit that students who were forced to use mathematical strategies (even if they were incorrect) rather than gaming strategies might begin to learn more from the game than they would have if they had been allowed to continue using non-mathematical strategies. This study's findings that students in the revised version outperformed students in the original version only in Stage 4 and Stage 5 indicates that this might be the case (though it is then notable that there was no increase in performance in Stage 6). Additionally, the increased understanding of content only covered in later stages of the game likely would not translate to an increase on paper-and-pencil posttest scores in this case because the posttest was not designed to be diagnostic and therefore no items on the posttest corresponded solely to the content covered in those stages. Had the posttest been more diagnostic in nature, it might have been possible to test this theory.

Additionally, the game did not adapt to specific student errors. If information about the specific misconception (e.g., a partitioning error) being displayed by a specific student at a specific time had been fed back into the computer to generate targeted feedback (e.g., "I think you're supposed to count spaces to determine the denominator."), it is possible that reducing the amount of construct irrelevant behavior would have increased in-game learning as students would have received helpful advice more often. Without the targeted feedback, students who did not understand the content well often simply flailed around in both versions of the game, switching back and forth between different errors or guessing strategies.

In summary, data mining techniques led to the identification of two specific revisions to an educational video game that were posited to reduce the occurrence of construct irrelevant behavior. As anticipated, this study found that students playing the revised version used significantly fewer non-mathematical strategies to solve problems in the game than students playing the original version. Additionally, student perception of the revised version of the game was more positive than student perception of the original version, despite concerns that the revisions might negatively affect perception. However, there were no corresponding differences in performance.

These results support the use of data mining techniques to identify specific game features for revision, and demonstrate the utility of comparing the original and revised versions after revisions have been made. The results also indicate that modifications that decrease construct irrelevant ("gaming") behavior in the game do not necessarily negatively impact student perception of the game, as may have been expected. While this study did not show an increase in performance corresponding to the observed decrease in construct irrelevant behavior, further research in this area is necessary to accurately determine the effect of such revisions on in-game learning.

ACKNOWLEDGEMENT

The work reported herein was completed at the National Center for Evaluation, Standards, and Student Testing (CRESST) at UCLA and was supported under the Educational Research and Development Centers Program, PR/Award Number R305C080015. The findings and opinions expressed here do not necessarily reflect the positions or policies of the National Center for Education Research, the Institute of Education Sciences, or the U.S. Department of Education.

REFERENCES

AMERSHI, S., AND CONATI, C. (2011). Automatic recognition of learner types in exploratory learning environments. In *Handbook of Educational Data Mining*, C. ROMERO, S. VENTURA, M. PECHENIZKIY, AND R. S. J. D. BAKER, Eds. CRC Press, Boca Raton, FL, 213-230.

BAKER, R. S. J. D., CORBETT, A. T., KOEDINGER, K. R., EVENSON, S., ROLL, I., WAGNER, A. Z., NAIM, M., RASPAT, J., BAKER, D. J., AND BECK, J. E. 2006. Adapting to when students game an intelligent tutoring system. In *Intelligent Tutoring Systems*, M. IKEDA, K. ASHLEY, AND T.-W. CHAN, Eds. Springer, Berlin, Germany, 392-401.

BECK, J., AND RODRIGO, M. M. T. 2014. Understanding wheel spinning in the context of affective factors. In *Intelligent Tutoring Systems*, S. TRAUSAN-MATU, K. E. BOYER, M. CROSBY, AND K. PANOURGIA, Eds. Springer, Berlin, Germany, 162-167.

BEJAR, I. I. 1984. Educational diagnostic assessment. *Journal of Educational Measurement, 21*, 2, 175-189.

BERKHIN, R. 2006. A survey of clustering data mining techniques. In *Grouping Multidimensional Data*, J. KOGAN, C. NICHOLAS, AND M. TEBOULLE, Eds. Springer, New York, NY, 25-72.

BONCHI, F., GIANNOTI, F., GOZZI, C., MANCO, G., NANNI, M., PEDRESCHI, D., RENSO, C., AND RUGGIERI, S. 2001. Web log data warehouses and mining for intelligent web caching. *Data & Knowledge Engineering*, *39*, 165-189.

BRIGHT, G. W., BEHR, M. J., POST, T. R., AND WACHSMUTH, I. 1988. Identifying fractions on number lines. *Journal for Research in Mathematics Education*, *19*, 3, 215-232.

BUCKLEY, B. C., GOBERT, J. D., AND HORWITZ, P. 1999. Using log files to track students' model-based inquiry. *Journal of Management*, *25*, 1, 1-27.

CARPENTER, T. P., FENNEMA, E., FRANKE, M. L., LEVI, L. W., AND EMPSON, S. B. 2000. *Cognitively Guided Instruction: A Research-Based Teacher Professional Development Program for Elementary School Mathematics*. National Center for Improving Student Learning and Achievement in Mathematics and Science, Madison, WI.

CASTRO, F., VELLIDO, A., NEBOT, A., AND MUGICA, F. 2007. Applying data mining techniques to e-learning problems. In *Evolution of Teaching and Learning Paradigms in Intelligent Environments, Studies in Computational Intelligence (SCI) Volume 62*, L. C. JAIN, R. A. TEADMAN, AND D. K. TEDMAN, Eds. Springer, Berlin, Germany, 183-221.

CEN, H., KOEDINGER, K. R., AND JUNKER, B. 2007. Is Over Practice Necessary? Improving Learning Efficiency with the Cognitive Tutor through Educational Data Mining. *Frontiers in Artificial Intelligence and Applications*, *158*, 511-518.

CHUNG, G. K. W. K., BAKER, E. L., VENDLINSKI, T. P., BUSCHANG, R. E., DELACRUZ, G. C., MICHIUYE, J. K., AND BITTICK, S. J. 2010. Testing instructional design variations in a prototype math game. In *Current perspectives from three national R&D centers focused on game-based learning: Issues in learning, instruction, assessment, and game design.* Structured poster session at the annual meeting of the American Educational Research Association, Denver, CO, April, 2010, R. ATKINSON, Chair.

CHUNG, G. K. W. K., AND KERR, D. 2012. *A primer on data logging to support extraction of meaningful information from educational games: An example from Save Patch.* CRESST Report 814. National Center for Research on Evaluation, Standards, and Student Testing, University of California, Los Angeles, CA.

DOLMANS, D. H. J. M., GIJSELAERS, W. H., SCHMIDT, H. G., AND VAN DER MEER, S. B. 1993. Problem effectiveness in a course using problem-based learning. *Academic Medicine*, *68*, 207-213.

ETHEREDGE, M., LOPES, R., AND BIDARRA, R. 2013. A generic method for classification of player behavior. In *Proceedings of the Second AIIDE Workshop on Artificial Intelligence in the Game Design Process*, M. J. NELSON, A. M. SMITH, AND G. SMITH, Eds. AAAI Press, Palo Alto, CA.

FISCH, S. M. 2005. Making educational computer games "educational." In *Proceedings of the 4th International Conference for Interaction Design and Children.* Boulder, CO, June 2005, ACM Press, New York, NY, 56-61.

FRAWLEY, W. J., PIATESKI-SHAPIRO, G., AND MATHEUS, C. J. 1992. Knowledge discovery in databases: An overview. *AI Magazine*, *13*, 3, 57-70.

GARCIA, E., ROMERO, C., VENTURA, S., DE CASTRO, C., AND CALDERS, T. 2011. Association rule mining in learning management systems. In *Handbook of Educational Data Mining*, C. ROMERO, S. VENTURA, M. PECHENIZKIY, AND R. S. J. D. BAKER, Eds. CRC Press, Boca Raton, FL, 93-106.

HARPSTEAD, E., MACLELLAN, C. J., KOEDINGER, K. R., ALEVEN, V., DOW, S. P., AND MYERS, B. A. 2013. Investigating the solution space of an open-ended educational game using conceptual feature extraction. In *Proceedings of the 6th International Conference on Educational Data Mining (EDM 2013),* S. K. D'MELLO, R. A. CALVO, AND A. OLNEY, Eds. Memphis, TN, July 2013, International Educational Data Mining Society, 51-58.

HERSHKOVITZ, A., AND NACHMIAS, R. 2011. Log-based assessment of motivation in online learning. In *Handbook of Educational Data Mining*, C. ROMERO, S. VENTURA, M. PECHENIZKIY, AND R. S.J.D. BAKER, Eds. CRC Press, Boca Raton, FL, 389-416, 287-297.

JITENDRA, A., AND KAMEENUI, E. J. 1996. Experts' and novices' error patterns in solving part-whole mathematical word problems. *Journal of Educational Research, 90,* 1, 42-51.

KERR, D. 2014. *Identifying common mathematical misconceptions from actions in educational video games.* CRESST Report 838. National Center for Research on Evaluation, Standards, and Student Testing, University of California, Los Angeles, CA.

KERR, D., AND CHUNG, G. K. W. K. 2012. Identifying key features of student performance in educational video games and simulations through cluster analysis. *Journal of Educational Data Mining*, *4*, 144-182.

KIM, J. H., GUNN, D. V., SCHUH, E., PHILLIPS, B. C., PAGULAYAN, R. J., AND WIXON, D. 2008. Tracking real-time user experience (TRUE): A comprehensive instrumentation solution for complex systems. In *Proceedings of the 26th annual SIGCHI Conference on Human Factors in Computing Systems*. Florence, Italy, April 2008, ACM Press, New York, NY, 443-452.

MAECHLER, M. 2012. *cluster: Cluster analysis extended Rousseeuw et al.* R package version 1.14.3. Retrieved from http://cran.r-project.org/web/packages/cluster/index.html

MALCOM, S. M., CHUBIN, D. E., AND JESSE, J. K. 2004. *Standing our ground: A guidebook for STEM educators in the Post-Michigan Era.* American Association for the Advancement of Science, Washington, DC.

MCNEIL, N. M., AND ALIBALI, M. W. 2005. Why won't you change your mind? Knowledge of operational patterns hinders learning and performance on equations. *Child Development*, *76*, 4, 883-899.

MERCERON, A., AND YACEF, K. 2004. Mining student data captured from a web-based tutoring tool: Initial exploration and results. *Journal of Interactive Learning Research*, *15*, 319-346.

MOSTOW, J., BECK, J. E., CUNEAO, A., GOUVEA, E., HEINER, C., AND JUAREZ, O. 2011. Lessons from Project LISTEN's session browser. In *Handbook of Educational Data Mining*, C. ROMERO, S. VENTURA, M. PECHENIZKIY, AND R. S. J. D. BAKER, Eds. CRC Press, Boca Raton, FL, 389-416.

NATIONAL COUNCIL OF TEACHERS OF MATHEMATICS. 2000. *Principles and standards for school mathematics.* Reston, VA.

NATIONAL MATHEMATICS ADVISORY PANEL. 2008. *Foundations for success: The final report of the National Mathematics Advisory Panel.* U.S. Department of Education, Washington, DC.

NATIONAL RESEARCH COUNCIL. 2011. *Learning science through computer games and simulations.* National Academies Press, Washington, DC.

NILAKANT, K., AND MITOVIC, A. 2005. Application of data mining in constraint-based intelligent tutoring systems. In *Proceedings of the 2005 Conference on Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology*, C.-K. LOOI, G. I. MCCALLA, B. BREDEWEG, AND J. BREUKER, Eds. Amsterdam, Netherlands, July 2005, IOS Press, Amsterdam, Netherlands, 896-898.

R DEVELOPMENT CORE TEAM. 2010. *R: A language and Environment for Statistical Computing*. Retrieved from http://www.R-project.org

RAHKILA, M., AND KARJALAINEN, M. 1999. Evaluation of learning in computer based education using log systems. In *Proceedings of 29th ASEE/IEEE Frontiers in Education Conference (FIE '99)*. San Antonio, TX, October 2009, IEEE, Piscataway, NJ, 16-22.

RODRIGO, M. M. T., ANGLO, E. A., SUGAY, J. O., AND BAKER, R. S. J. D. 2008. Use of unsupervised clustering to characterize learner behaviors and affective states while using an intelligent tutoring system. In *Proceedings of the 16th International Conference on Computers in Education,* T.-W. CHAN, G. BISWAS, F.-C. CHEN, S. CHEN, C.CHOU, M. JACOBSON, KINSHUK, F. KLETT, C.-K. LOOI, T. MITROVIC, R. MIZOGUCHI, K. NAKABAYASHI, P. REIMANN, S. SUTHERS, S. YANG, AND J.-C.

YANG, Eds. Taipei, Taiwan, October 2008, Asia-Pacific Society for Computers in Education, Taipei, Taiwan, 49-56.

ROMERO, C., GONZALEZ, P., VENTURA, S., DEL JESUS, M. J., AND HERRERA, F. 2009. Evolutionary algorithms for subgroup discovery in e-learning: A practical application using Moodle data. *Expert Systems with Applications*, *39*, 1632-1644.

ROMERO, C., AND VENTURA, S. 2007. Educational data mining: A survey from 1995 to 2005. *Expert Systems with Applications*, *35*, 135-146.

ROMERO, C., VENTURA, S., PECHENIZKIY, M., AND BAKER, R. S. J. D. 2011. *Handbook of Educational Data Mining*. CRC Press, Boca Raton, FL.

RUPP, A. A., GUSHTA, M., MISLEVY, R. J., AND SHAFFER, D. W. 2010. Evidence centered design of epistemic games: Measurement principles for complex learning environments. *The Journal of Technology, Learning, and Assessment*, *8*, 4. Retrieved from http://ejournals.bc.edu/ojs/index.php/jtla/article/view/1623/1467

SAXE, G. B., SHAUGHNESSY, M. M., SHANNON, A., LANGER-OSUNA, J. M., CHINN, R., AND GEARHART, M. 2007. Learning about fractions as points on a number line. In *The learning of mathematics: Sixty-ninth yearbook*, M. E. STRUTCHENS AND G. W. MARTIN, Eds. National Council of Teachers of Mathematics, Reston, VA, 221-237.

SIEBERT, D., AND GASKIN, N. 2006. Creating, naming, and justifying fractions. *Teaching Children Mathematics*, *12*, 8, 394-400.

SISON, R., NUMAO, M., AND SHIMURA, M. 2000. Multistrategy discovery and detection of novice programmer errors. *Machine Learning*, *38*, 157-180.

TRIGWELL, K., PROSSER, M., AND WATERHOUSE, F. 1999. Relations between teachers' approaches to teaching and students' approaches to learning. *Higher Education*, *37*, 57-70.

VENDLINSKI, T. P., DELACRUZ, G. C., BUSCHANG, R. E., CHUNG, G. K. W. K., AND BAKER, E. L. 2010. *Developing high-quality assessments that align with instructional video games*. CRESST Report 774. National Center for Research on Evaluation, Standards, and Student Testing, University of California, Los Angeles, CA.