Using DataShop to discover a better knowledge component model of student learning

Contents

Square brackets in text either indicate actions to show on-screen in the video or summarize the section.

Introduction

This video illustrates the use of DataShop to perform exploratory analysis of DataShop data, generate a theory for optimizing a cognitive model, and test that theory both visually and statistically within DataShop. It also illustrates, more broadly, how educational technology data can be used to gain insights into student thinking and learning.

View and discuss learning curves for old model [Learning Curves]

We start this analysis by exploring learning curves of knowledge components for the dataset “Geometry Area 1997-97”. One of the things we notice is that we have some nice learning curves for skills like circle-area—the error rate’s high and then goes down—and similarly for pentagon-area, and trapezoid. Parallelogram-area is low and stays low, so there’s not so much learning there, but there’s no need for learning so it’s not too worrisome. One curve that’s a little bit odd, however, is the compose-by-addition curve because you can see that it’s flat here and there doesn’t seem to be any learning going on. You also might wonder What *is* “compose-by-addition”? So that’s one reason to go look at those problems and see how they’re coded. We also want to become familiar with the domain a little bit, and see what the tutor problems looked like to students who worked with them.

View and discuss the coding in the tutor for two problems [Hypothesis generation; mental composition]

This pogs problem is an example of one of the problems that students saw, and when they see it in the tutor these first three columns are initially present. Given the radius of the pog and the sides on this square, they are asked ”How many square inches of the scrap metal are remaining?”, so they have to find that. Of course, the way to find that is to find the area of the square and the area of the circle and subtract the two but they don’t have to do it in their head, they can add columns—these columns here—and put the area of the square and the area of the pog there.

There are some other problems like “Building a sidewalk”, where similar things are going on: here, they need to find the area of a sidewalk by finding the area of this swimming pool and the area of this larger rectangle, the yard, and then subtract. But in *this* problem, the key columns that you have to subtract--“area of pool and area of yard”--are already present in the beginning so the student doesn’t have to so much think of setting those as sub-goals; they are there in the first place.

It turns out that in the original cognitive model, finding the area of the sidewalk—this kind of subtraction of that additive structure, if you will—and the subtraction in the “pogs” problem were treated the same: they’re both part-whole structure. And the skill in that original model is called “Compose by addition”.

We can hypothesize here that maybe the compose-by-addition curve isn’t going down because in fact there are different kinds of composition in the harder one. [*Go back to DataShop*] Maybe what’s causing these blips here is an increase in those kinds of problems where you have to come up with the sub-goals yourself and add corresponding columns as opposed to being given them.

We then need to figure out which of these problems containing compose-by-addition actually require a mental decomposition because the columns weren’t there and which of them didn’t. We know that POGS requires the mental composition, so we might call that knowledge component “Decompose”; the SIDEWALK problem will stay as compose-by-addition. There are a few other problems that I’ve looked up and determined whether or not they also require this mental composition.

Export “Textbook New” KC Model and make changes in Excel [Dataset is Geometry Area (1996-1997)]

We’ll now implement the new knowledge-component model in a format DataShop can read and import it into DataShop. We first need to export the existing knowledge component model, or KC model, called “textbook new”.

Opening this KC model file in Excel, we want to zero-in on just those compose-by-addition rows and particularly the problems that we care about. The first thing we do is turn the auto filter on. Now we see a list of our 10 KC’s here. We just want to look at compose-by-addition rows, so I deselect all and just select compose-by-addition. Now we see a limited number of problems here. Some of them, like “Building-a-sidewalk”, are going to remain as “compose-by-addition”. Some of them, like POGS, we’re going to change to “decompose”.

In the existing column here, I enter the KCs for the new KC model. I can make changes to the existing column because DataShop will prompt me to rename the KC model when I import it.

Before recording this video, I discovered that there were four problems where the extra columns were not given to the students at the beginning of the problem. We'll label these problems with the "decompose" KC. The rest we'll keep the same.

[Make changes to the KCs for the four problems below]

```
ONE_CIRCLE_IN_SQUARE   decompose
POGS                   decompose
TWO_CIRCLES_IN_CIRCLE  decompose
TWO_CIRCLES_IN_SQUARE  decompose
```

Import the new model [KC Model Import]

[Not shown in video.]

Look at AIC/BIC [KC Model info / LFA computes AIC/BIC] [Statistical evidence for a better model]

Now let's look at a statistical measure of goodness of fit for the new model.

We see lower AIC and BIC values for our new model compared to "Textbook New", which indicates that the model with the extra knowledge component better fits the data. It even has a better BIC value after being penalized for the extra [KC] parameter, which BIC takes into account.

Look at compose-by-addition with old model, max opp cutoff to 20

[Compose-by addition, primary is original "textbook new"] In this original model, not only is it flat but the error rate is substantial at points, starting at 27%, almost 30. Here it's 50% at 12 opportunities which is still 24 students, not a small piece of the sample as opposed to out here, where this is just one student with an error rate of 100%. Here we can apply a feature of DataShop to filter out data points for students who had more than some number of opportunities with this KC. We'll use this "Opportunity Cutoff" and set the maximum number of opportunities to "20": this eliminates data for all students who had more than 20 opportunities to use this knowledge component.

That looks a little better. Now let's switch to our new model for this KC, which will renumber the X-axis, "opportunity".

Look at compose-by-addition and decompose with the new model [KC Model switching] [Evidence for a better model]

[Change to compose-by-addition KC; set primary to Textbook New Decompose (best model); max opp. cutoff to 12]

Remember those peaks we saw before like that 50% in peak? It's pretty much taken out here; everything is below 30%. This new model shows the "smoothing" of the learning curve. We can further smooth it by setting the max opportunity cutoff to 12.

For a student, it's probably not the subtraction of those two areas that's hard—just because the columns are given to you doesn't mean you know which ones to subtract. But you at least don't have to think "Oh, I need to figure out the areas of these two. What's the formula for a circle?" That's a complex process that the previously overly-general KC requires that this one doesn't. This one at least requires you to think "which two numbers do I need to use and what do I need to do with them—subtract. Now I have to do the subtraction". A 20% error rate is probably more accurate for what that piece of the puzzle is, whereas the other piece is much harder.

[Change to decompose KC; set Primary to Textbook New Decompose; max cut-off at 12]
Let's take a look at our new KC, decompose. It's a bit noisy—we don't have a lot of data so the number of observations here is somewhat limited—but it looks pretty reasonable that there is real improvement.

Add the new model as a secondary curve [Primary/Secondary KC model views; two models on one curve]

The curve is still pretty bumpy. Further investigation led to hypothesizing a new knowledge component, and using that model further yields a better fit. If we set the "Secondary" KC model to this new model, "DecomposeArith", we see that the latest model tracks the decompose KC even better.

Conclusion

In this video, we hypothesized a new knowledge component based on a unusual learning curve and explored that hypothesis in DataShop. We started by looking at a flat learning curve for a knowledge component, examined the problems that students saw that contained that knowledge component, and hypothesized that the original knowledge component was too general, covering two dissimilar cases. We split the knowledge component into two, creating a new knowledge component model and imported it into DataShop, where we discovered it better fit the data both statistically and visually. Our analysis also revealed some new insights into student thinking and learning.

We saw how DataShop can aid the discovering of important "hidden skills", like knowing when and how to break a problem down. Such hidden skills are not discussed in typical instruction or described by textbooks, so their discovery can lead to course improvements that can significantly enhance student learning.

Conclusion bullets:

- Hypothesized a "decompose" knowledge component
    - Unusual curve
    - Looked at problems that students saw
    - Split the existing knowledge component
    - Imported the new knowledge component model

- Discovered a “hidden” skill not present in typical instruction
    - New knowledge [component] model is a better fit to the data
    - It better accounts for student error rate across different problem types
    - It better explains learning – changes in the error rate over time