Week 7 Video 6

Knowledge Inference:
Q-Matrix

What is the Q-Matrix?

- (Has nothing to do with Keanu Reeves)

What is the Q-Matrix?

- A table
- Where rows are items
- And columns are skills

- (Tatsuoka, 1983; Barnes, 2005)

- Also called a KC [knowledge component] Model
- Or a skill-item mapping

What is the Q-Matrix? (Tatsuoka, 1983; Barnes, 2005)

	Skill1	Skill2	Skill3	Skill4
Item1	1	0	0	0
Item2	1	1	0	0
Item3	1	0	1	0
Item4	0	0	0	1
Item5	0	0	1	1
Item6	0	1	0	0

Example

	Add	Subtract	Multiply	Divide
7 + 3 + 2	1	0	0	0
7 + 3 - 2	1	1	0	0
(7 + 3) * 2	1	0	1	0
7 / 3 / 2	0	0	0	1
7 * 3 / 2	0	0	1	1
7 - 3 - 2	0	1	0	0

How do we get a skill-item mapping?

- Automatic model discovery
- Hand-development and refinement
- Hybrid approaches

How do we get a skill-item mapping?

- Automatic model discovery
- Hand-development and refinement
- Hybrid approaches

Automated Model Discovery

- Learn the mapping between items and skills solely from data

One popular algorithm

- Barnes, T., D. Bitzer, & M. Vouk. (2005). Experimental analysis of the q-matrix method in knowledge discovery. *Proceedings of the 15th International Symposium on Methodologies for Intelligent Systems.*

Not the only approach

- Non-negative matrix factorization
- Lots of linear algebra
- Desmarais, M.C. (2011) Conditions for effectively deriving a Q-Matrix from data with Non-negative Matrix Factorization. *Proceedings of the 4th International Conference on Educational Data Mining,* 41-50.

First question

- How many skills should we use?

- This is determined empirically
  1. Try 1 skill
  2. Try 1 more skill than previous model (e.g. 2,3,4,5…)
  3. Does the new model do better than the previous model?
     If so, go to step 2.
     If not, quit and use the previous model.

After that: Follow pseudocode

```
Set MinError = LargeNumber;
For Starts= 1 to NumStarts
    Randomly initialize Q[NumCon][NumQues];
    Set Q* = Q; Set CurrError = Error(Q) ;
    For Iter = 1 to NumIter;
      For c= 1 to NumCon
        For q= 1 to NumQues
          Q*[c][q] = Q[c][q] + Delta;
          If (Error(Q*) < CurrError)
            Do
              Set Q=Q*; Set CurrError = Error(Q*);
              Q*[c][q] = Q[c][q] + Delta;
            While (Error(Q*) < CurrError);
          Else
            Q*[c][q] = Q[c][q] – Delta;
            While (Error(Q*) < CurrError)
                Set Q=Q*; Set CurrError = Error(Q*);
                Q*[c][q] = Q[c][q] - Delta;
     If (CurrError < MinError)
      Set BestQ = Q; Set MinError = CurrError;
```

Any questions?

Let’s Break That Down

- For each number of skills, the algorithm will be run a certain number of times, with a different (random) initial assignment of items to skills

- This avoids local minima

First Random Version

	Skill 1	Skill 2	Skill 3	Skill 4
7 + 3 + 2	0	0	0	0
7 + 3 - 2	1	1	0	1
(7 + 3) * 2	1	1	0	0
7 / 3 / 2	0	1	0	1
7 * 3 / 2	1	0	1	1
7 - 3 - 2	0	1	0	1

Second Random Version

	Skill 1	Skill 2	Skill 3	Skill 4
7 + 3 + 2	1	1	1	0
7 + 3 - 2	1	1	1	1
(7 + 3) * 2	0	0	0	1
7 / 3 / 2	0	1	0	1
7 * 3 / 2	1	1	1	1
7 - 3 - 2	1	0	1	0

Next…

- Take a set of passes through the table
- Systematically look at whether flipping each 1 to 0 (and each 0 to 1)
- Produces a better model

- Continue this process a predetermined number of times, or until a pass results in no changes

How do we know if it’s a better model?

- Several definitions

Barnes et al.'s definition

- Better models have the property that if a student knows skill X
- And item 1 and item 2 both have skill X
- Then a student who gets item 1 right will be more likely to get item 2 right
  - And item 1 wrong → item 2 wrong
  - And item 2 right → item 1 right
  - And item 2 wrong → item 1 wrong

Barnes et al.’s definition

- Given a skill-item mapping, you can predict, for each combination of skills whether a student should get each item correct or not

- A model’s degree of error is based on how many item-student pairs the prediction gets wrong

Subtlety

- Is skill conjunctive? (as in Barnes)
  - You need all relevant skills to get an item right
- Or is it compensatory? (Pardos et al., 2008)
  - Any relevant skill leads to getting an item right

Assumption

- The exact approach in Barnes et al. assumes no learning

Alternate Test of Model Goodness

- Look at student improvement over time
- Fit a model like PFA or BKT from Week 4, and see how well it fits data, given the skill-item mapping

- More on this in a sec

How do we get a Q-Matrix?

- Automatic model discovery
- Hand-development and refinement
- Hybrid approaches

Hand Development and Refinement

- The original way that Q-Matrices were created

- A domain expert creates the Q-Matrix using knowledge engineering

Hand Development and Refinement

- What kind of data can we use to guide refinement?

- Some slides adapted from a talk in my class by John Stamper

Strategies for Q-Matrix Refinement

- Try to smooth learning curves
- Look for skills with no apparent learning
- Look for problems with unexpected error rates

Tool for doing this

- Pittsburgh Science of Learning Center DataShop

- https://pslcdatashop.web.cmu.edu/

Possible to look at learning curves for different skill models

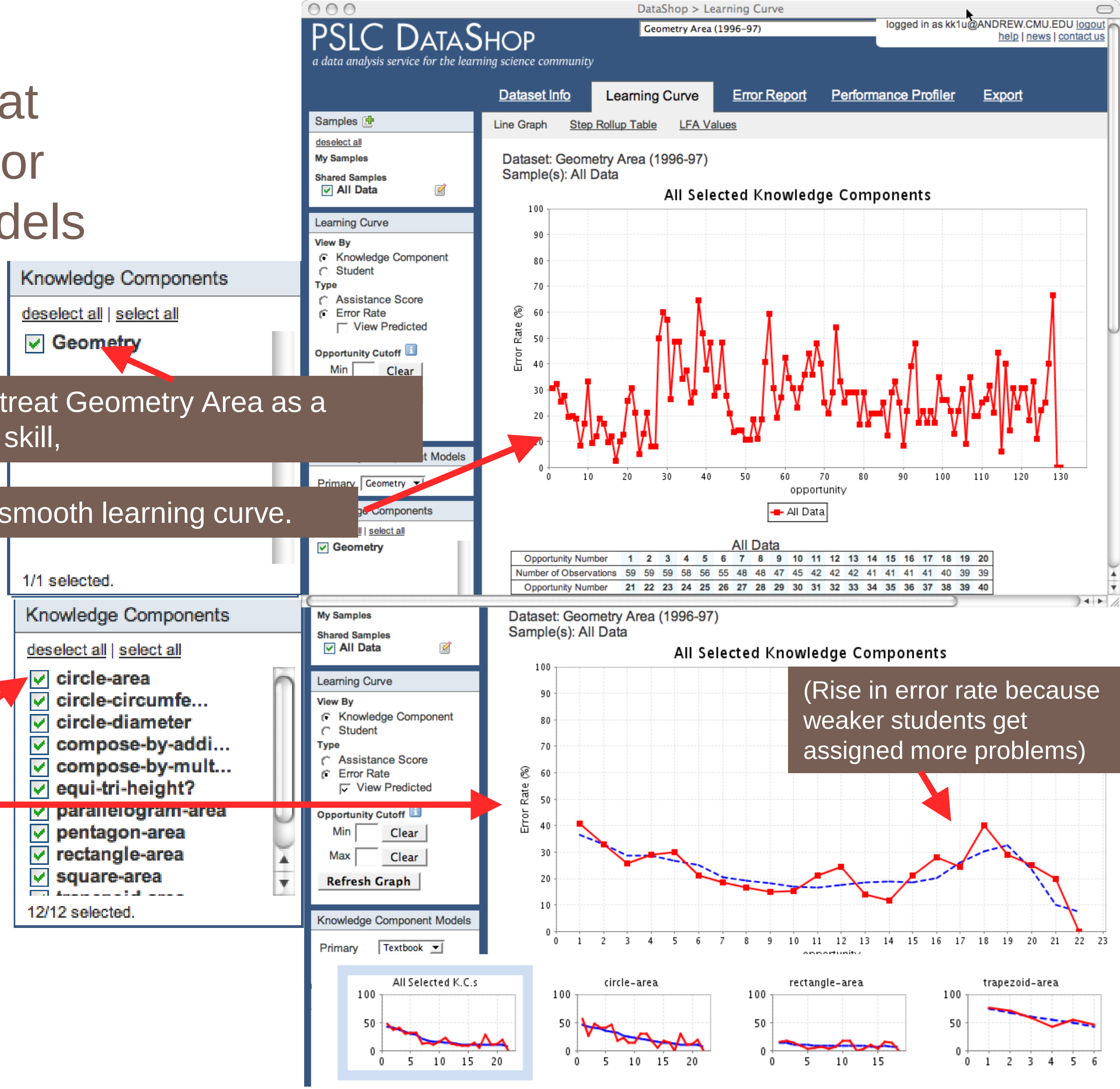
If you treat Geometry Area as a single skill,

Not a smooth learning curve.

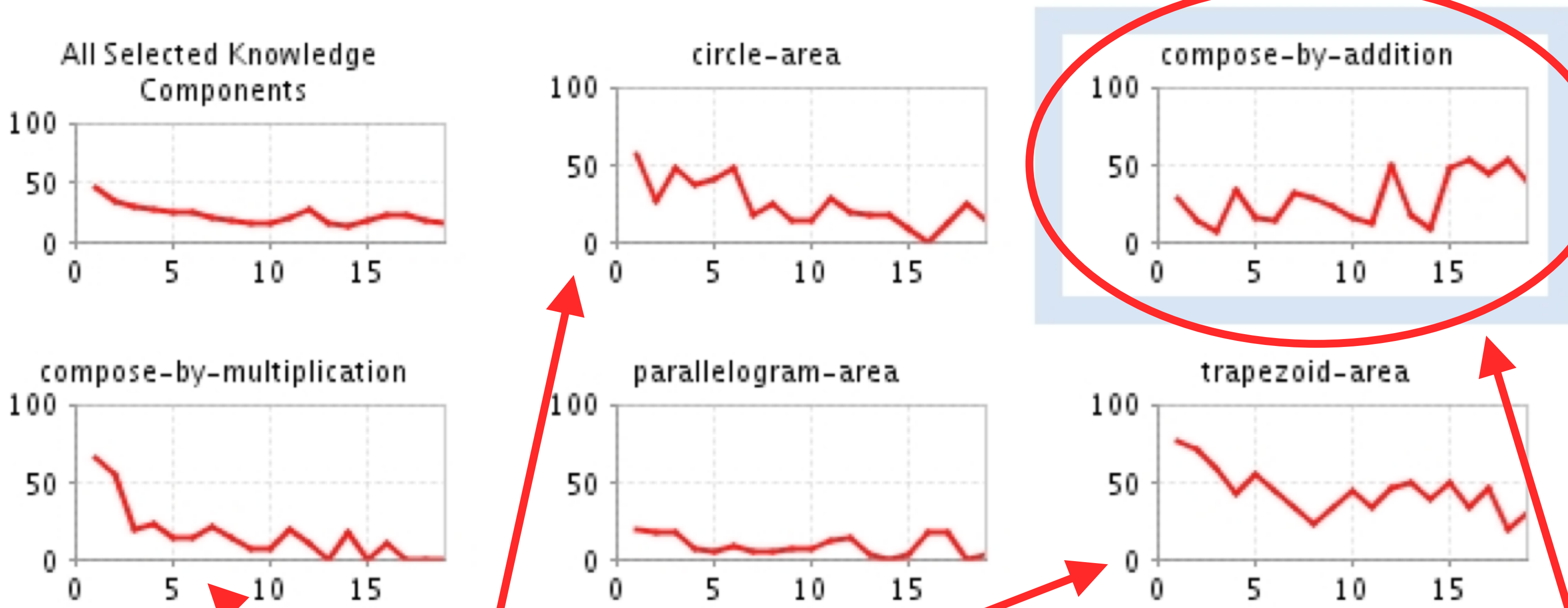
But if you split it into 12 skills

You get a smooth learning curve.

(Rise in error rate because weaker students get assigned more problems)



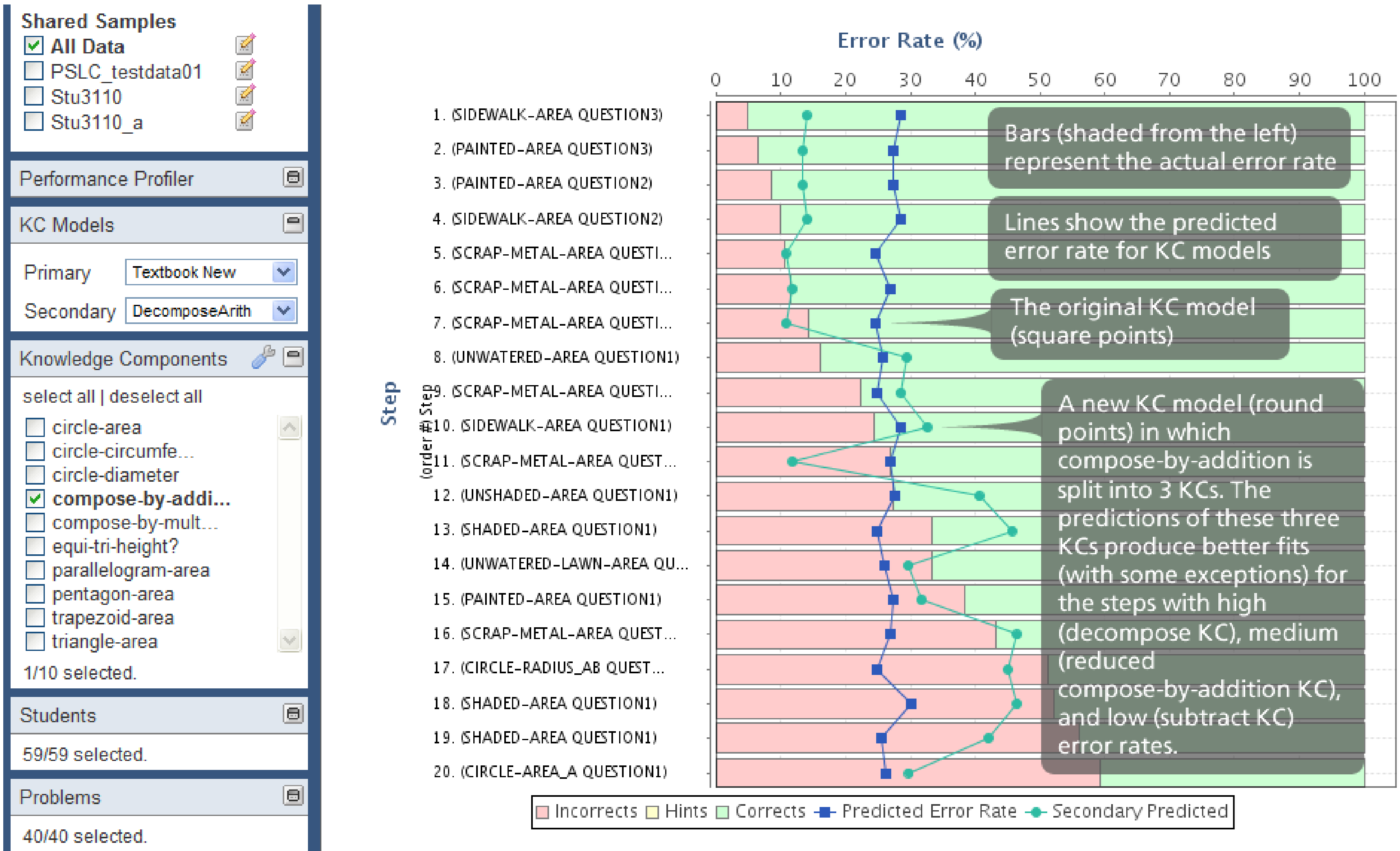
You can inspect curves for individual skills



Many curves show a reasonable decline (e.g. less errors over time)

Some do not => Opportunity to improve model!

Also look for problems with unexpected error rates



DataShop can apply model for you!

- Applies a mathematical model called LFA (similar to PFA) to data
- Can give AIC and BIC goodness measures for different skill-item mappings

Decompose — **15 KCs** — export
created by system on 2008-11-21 12:48:05.0
mapping type: correct-transaction-to-kc — status: ready to use
LFA values — AIC: 14697.59 **BIC: 15237.72**
14875 observations labeled with KCs
show model details

Textbook — **13 KCs** — export
created by system on 2008-11-21 12:48:05.0
mapping type: correct-transaction-to-kc — status: ready to use
LFA values — AIC: 14865.38 **BIC: 15375.07**
14875 observations labeled with KCs
show model details

Next Up

- Knowledge Structure Inference: Hybrid Approaches and Models with Prerequisites and Hierarchy