

# LEARNING COGNITIVE MODELS

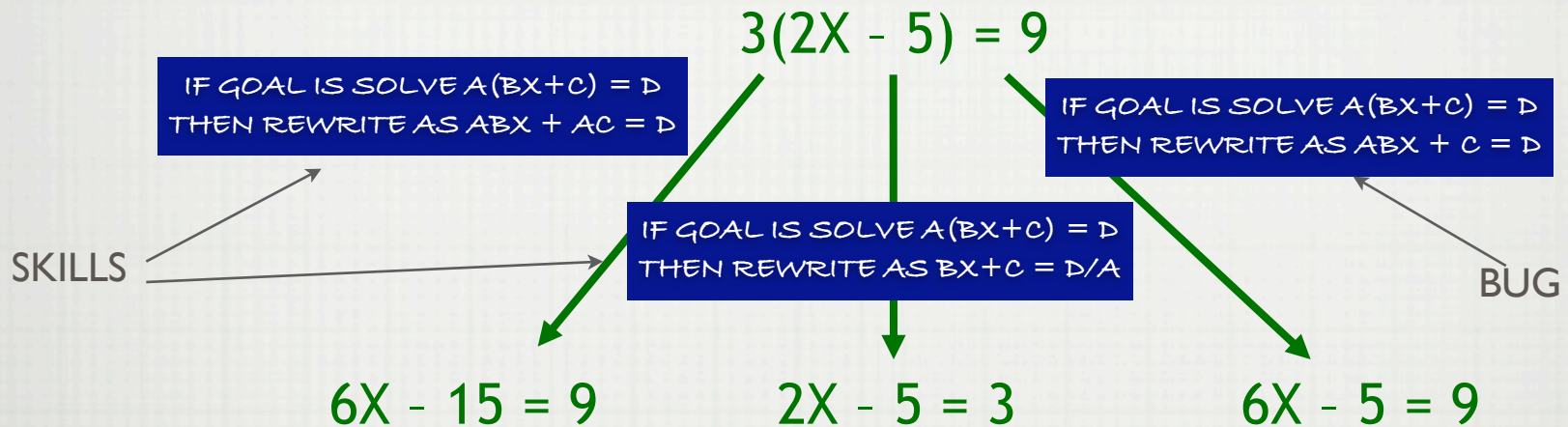


***Geoff Gordon***

Joint work w/ Ajit Singh, Hao Cen, John Stamper, Ken Koedinger

# COGNITIVE MODEL

Solves or simulates solving problems in the many ways students can



- Example benefit of cognitive model: **Model Tracing**
  - follow students through individual approaches to problem  $\Rightarrow$  context-sensitive instruction

# COGNITIVE MODEL

Solves or simulates solving problems in the many ways students can

---

## STUDENT

### **HAS:**

KC1: 80%

KC2: 72%

KC3: 11%

KC4: 34%

...

E.g., James L.

## STEP

### **REQUIRES:**

KC1: 2.32

KC2: 0

KC3: 1.07

KC4: 0

...

E.g., *find the area  
of region B*

## RESULT

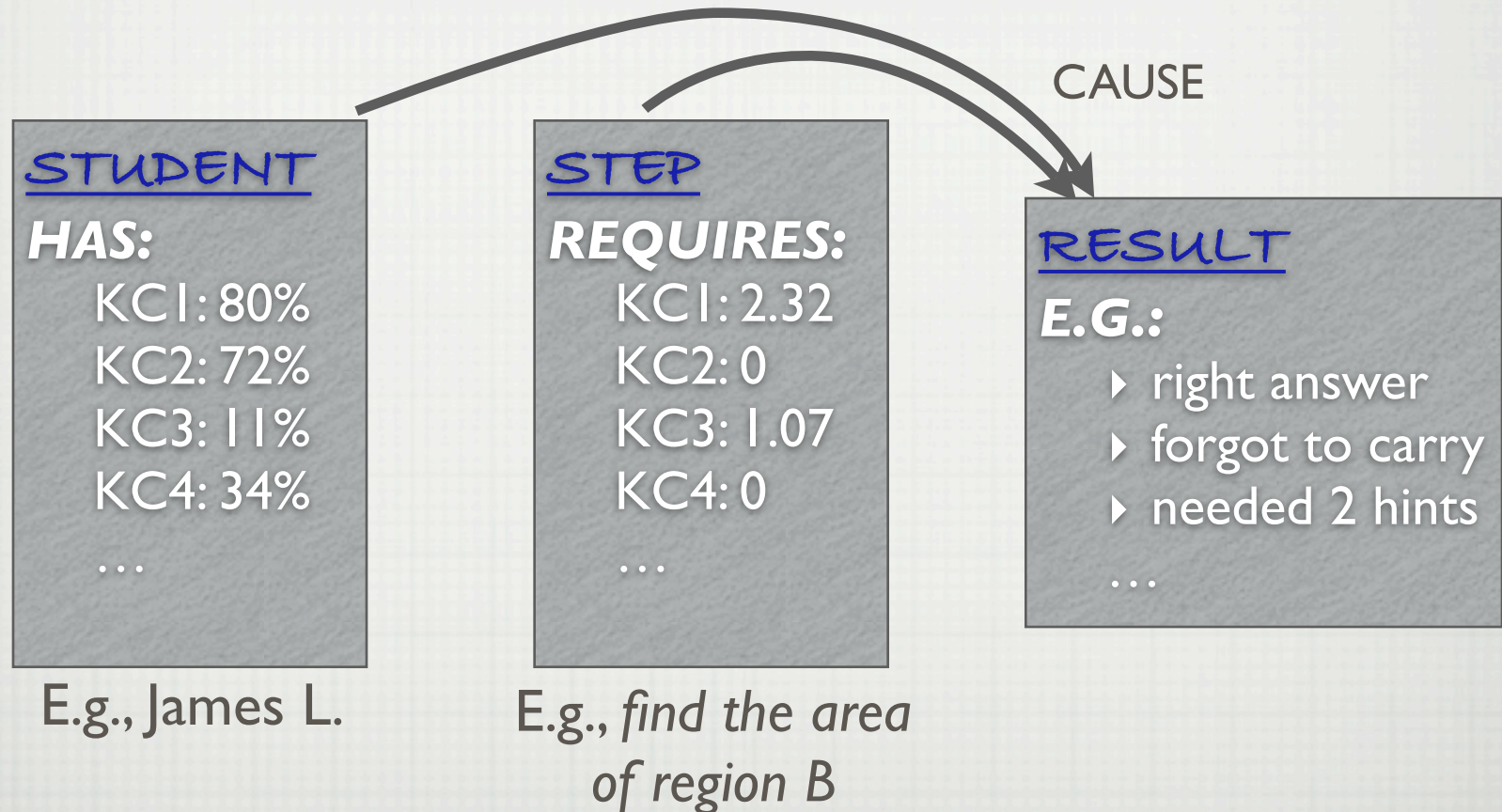
### **E.G.:**

- ▶ right answer
- ▶ forgot to carry
- ▶ needed 2 hints

...

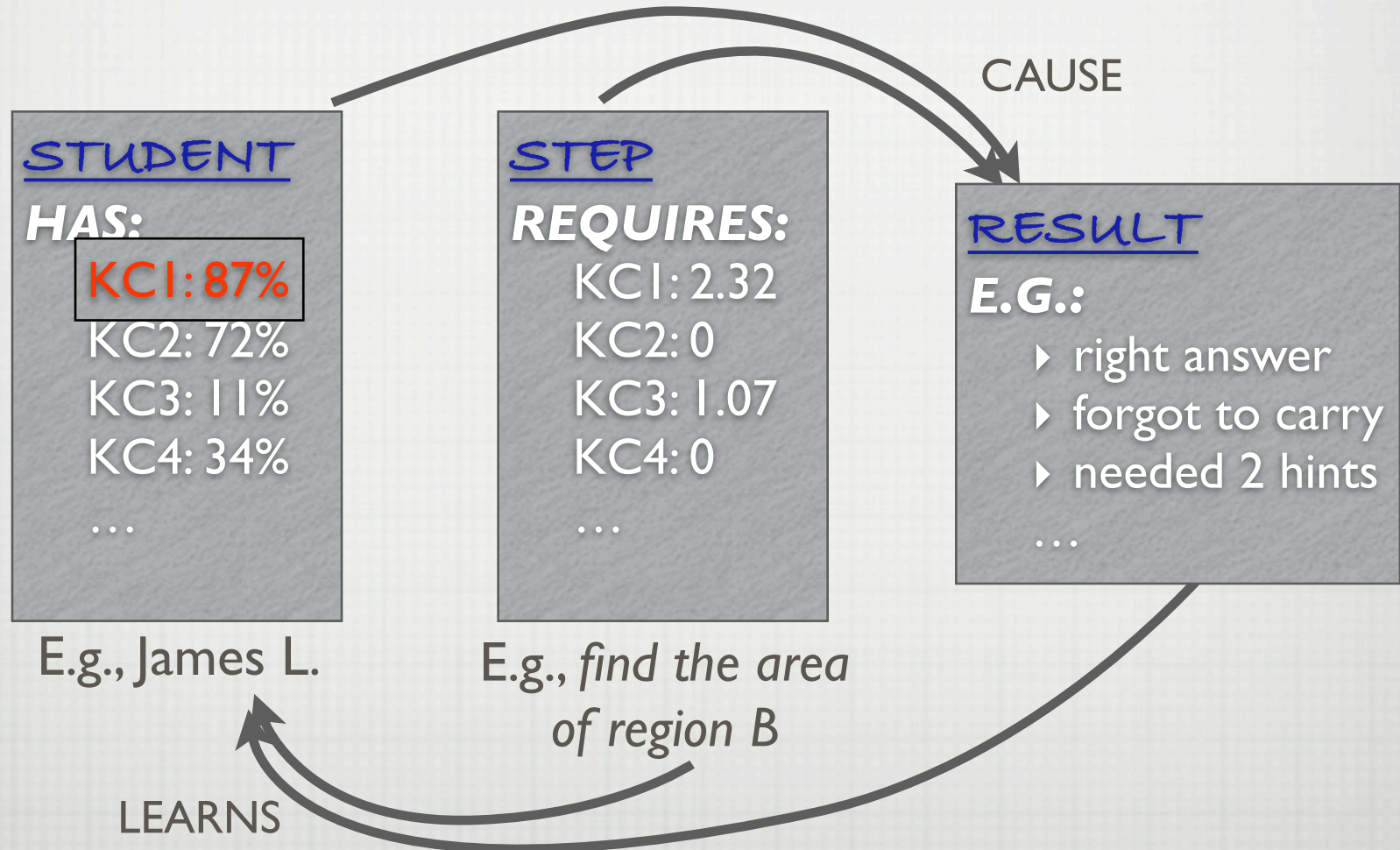
# COGNITIVE MODEL

Solves or simulates solving problems in the many ways students can



# COGNITIVE MODEL

Solves or simulates solving problems in the many ways students can



# SKILLS × ITEMS: THE Q MATRIX

---

Item   Skills:	Add	Sub	Mul	Div
2*8	0	0	1	0
2*8 - 3	0	1	1	0

$Q_{kj}$  = does step  
k need skill j?

Simplest representation of a cognitive model

Fancier: e.g., skills  $\leftrightarrow$  rules in a production system

# GETTING THE MODEL RIGHT!

---

- Cognitive model determines instruction
  - Through instructional decisions like problem selection, hints, ...
- A correct model is one that is consistent with student behavior, predicting task difficulty and transfer between instruction and test
- Cognitive models are discovered not designed

# GETTING THE MODEL RIGHT!

---

- Cognitive model determines instruction
  - Through instructional decisions like problem selection, hints, ...
- A correct model is one that is consistent with student behavior, predicting task difficulty and transfer between instruction and test
- Cognitive models ~~are~~ discovered not designed
  - should be*
  - ⇒ Huge data mining opportunity

# IT'S NOT EASY

---

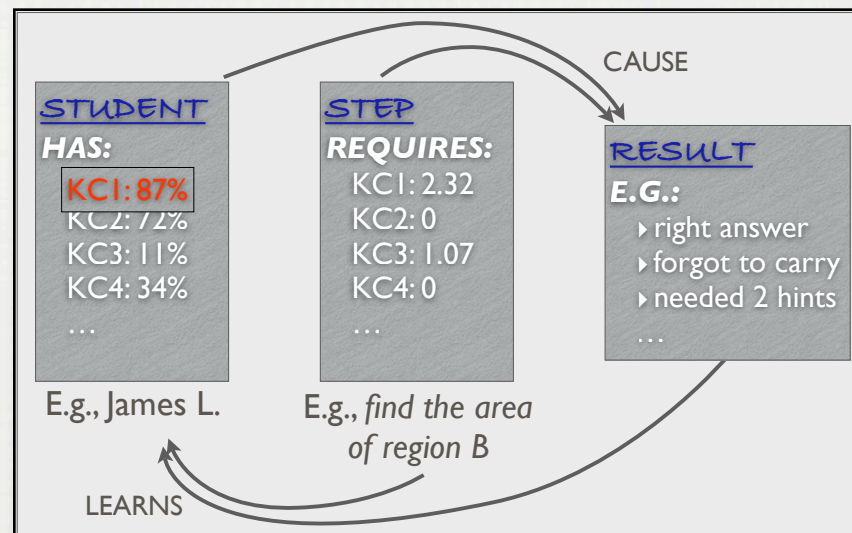
- **Student models** are a **key bottleneck** in cognitive tutor authoring and performance
  - ~80 hrs (and up!) to hand-develop model for 1 hr content
  - result may be too simple, not rigorously verified
- **Machine learning**, computational modeling, and data mining can help us **discover** detailed, accurate models of how students learn
- **New** models; data-driven **revision** of existing models

# BUT IT'S WORTH IT

---

- We have demonstrated **improvements in learning** from these more accurate models
  - E.G., Salden et al [2009]: adapting #examples vs. #problems yielded better transfer, same time spent
  - E.g., Cen et al [2007]: 12% less time to learn 6 geometry units (same retention) using tutor w/ better model
  
- Results can **transfer** beyond PSLC

# STATISTICAL METHODS FOR LEARNING COGNITIVE MODELS



- Data → learn a better model:
  - improved parameters ⇒ more accurate sequencing
  - refine list of skills (e.g., split a KC) ⇒ better coverage
  - discover completely new skills ⇒ aid problem design

# RAW DATA

Student ID	Step ID	Correct?	Skills
1	1	0	DECLARE_PARAM
1	2	1	WHILE_LOOP
1	3	1	DECLARE_PARAM
2	1	0	DECLARE_PARAM
2	4	1	PREFIX_OP
...	...	...	...

# RAW DATA

Student ID	Step ID	Correct?	Skills
1	1	0	DECLARE_PARAM
1	2	1	WHILE_LOOP
1	3	1	DECLARE_PARAM
2	1	0	DECLARE_PARAM
2	4	1	PREFIX_OP
...	...	...	...

AUTOMATICALLY  
RECORDED

FROM Q MATRIX

# RAW DATA

COMPUTED

Student ID	Step ID	Correct?	Skills	Opp
1	1	0	DECLARE_PARAM	1
1	2	1	WHILE_LOOP	1
1	3	1	DECLARE_PARAM	2
2	1	0	DECLARE_PARAM	1
2	4	1	PREFIX_OP	1
...	...	...	...	...

AUTOMATICALLY RECORDED

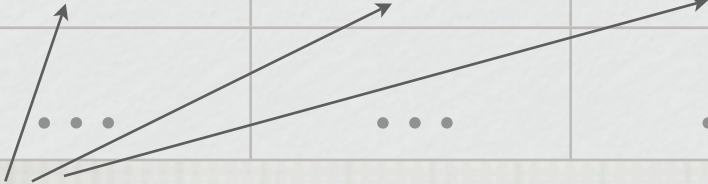
FROM Q MATRIX

# RAW DATA

COMPUTED

Student ID	Step ID	Correct?	Skills	Opp
1	1	0	DECLARE_PARAM	1
1				1
1				2
2	1	0	DECLARE_PARAM	1
2	4	1	PREFIX_OP	1
...	...	...	...	

E.g., KDD Cup data: 1,000s of students, 1,000,000s of steps



FROM Q MATRIX

AUTOMATICALLY RECORDED

# STATISTICAL METHODS FOR LEARNING COGNITIVE MODELS

---

- Model  $P(\text{correct} \mid \text{features of student and step})$

$$P(Y_i = 1 \mid X_{i1}, X_{i2}, \dots)$$

$$i = 1, 2, \dots, N$$

$$Y_i \in \{0, 1\}$$

$$X_{ij} \in \mathbb{R} \quad \leftarrow \text{may be arbitrary real, but often binary}$$

# INDICATOR FEATURES

$i$	Student ID	Step ID	Correct?	Skills
1	1	1	0	DECLARE_PARAM
2	1	2	1	WHILE_LOOP
3	1	3	1	DECLARE_PARAM
4	2	1	0	DECLARE_PARAM
5	2	4	1	PREFIX_OP

$i$	$Y_i$	$X_{i1}$	$X_{i2}$	$X_{i3}$	$X_{i4}$	$X_{i5}$	$X_{i6}$	$X_{i7}$	$X_{i8}$	$X_{i9}$
1	0	1	0	1	0	0	0	1	0	0
2	1	1	0	0	1	0	0	0	1	0
3	1	1	0	0	0	1	0	1	0	0
4	0	0	1	1	0	0	0	1	0	0
5	1	0	1	0	0	0	1	0	0	1

Student ID
Step ID
Skill

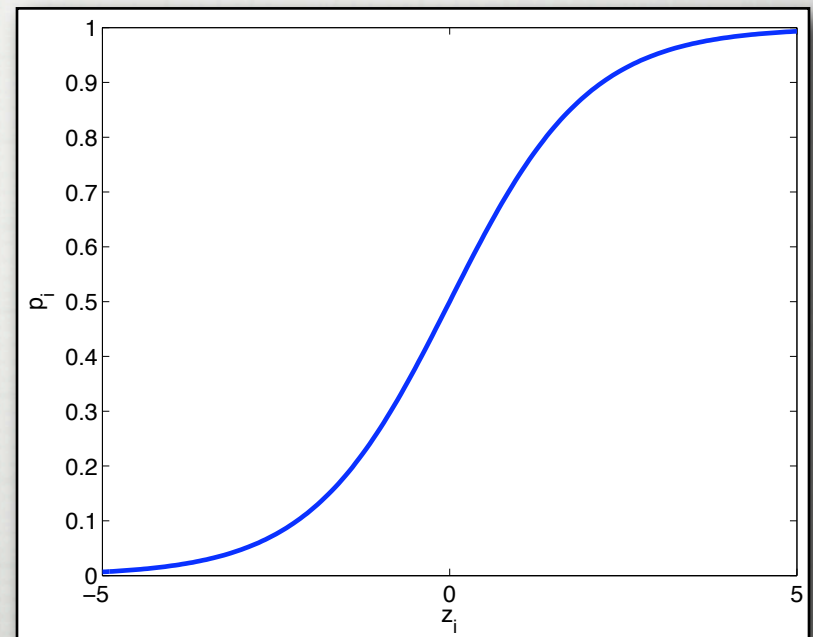
# STATISTICS OF COGNITIVE MODELS: LOGISTIC REGRESSION

- Write  $p_i = P(Y_i = 1 \mid \text{features of student and step for example } i)$

$$\log \frac{p_i}{1 - p_i} = \alpha_0 + \sum_{j=1}^d \alpha_j X_{ij} \equiv z_i$$

$$\alpha_j \in \mathbb{R}$$

Logistic  
regression  
model



# STATISTICS OF COGNITIVE MODELS: ADDITIVE FACTOR MODEL

---

- Additive Factor Model (Draney et al., 1995)
  - Logistic regression for  $P(\text{correct answer} \mid \text{student \& skill info})$

$$\log \frac{p_{ij}}{1 - p_{ij}} = \theta_i + \sum_k \beta_k Q_{kj} + \sum_k Q_{kj} (\gamma_k T_{ik})$$

i: student    j: step    k: skill

# STATISTICS OF COGNITIVE MODELS: ADDITIVE FACTOR MODEL

- Additive Factor Model (Draney et al., 1995)
  - Logistic regression for  $P(\text{correct answer} \mid \text{student \& skill info})$

**CORRECT?**

**STEP J USES SKILL K**

**OPP COUNT**

$$\log \frac{p_{ij}}{1 - p_{ij}} = \theta_i + \sum_k \beta_k Q_{kj} + \sum_k Q_{kj} (\gamma_k T_{ik})$$

i: student    j: step    k: skill

# STATISTICS OF COGNITIVE MODELS: ADDITIVE FACTOR MODEL

- Additive Factor Model (Draney et al., 1995)
  - Logistic regression for  $P(\text{correct answer} \mid \text{student \& skill info})$

**CORRECT?** points to  $p_{ij}$

**PARAMETERS** points to  $\theta_i$ ,  $\beta_k$ , and  $\gamma_k$

**STEP J USES SKILL K** points to  $Q_{kj}$  and  $T_{ik}$

**OPP COUNT** points to  $T_{ik}$

$$\log \frac{p_{ij}}{1 - p_{ij}} = \theta_i + \sum_k \beta_k Q_{kj} + \sum_k Q_{kj} (\gamma_k T_{ik})$$

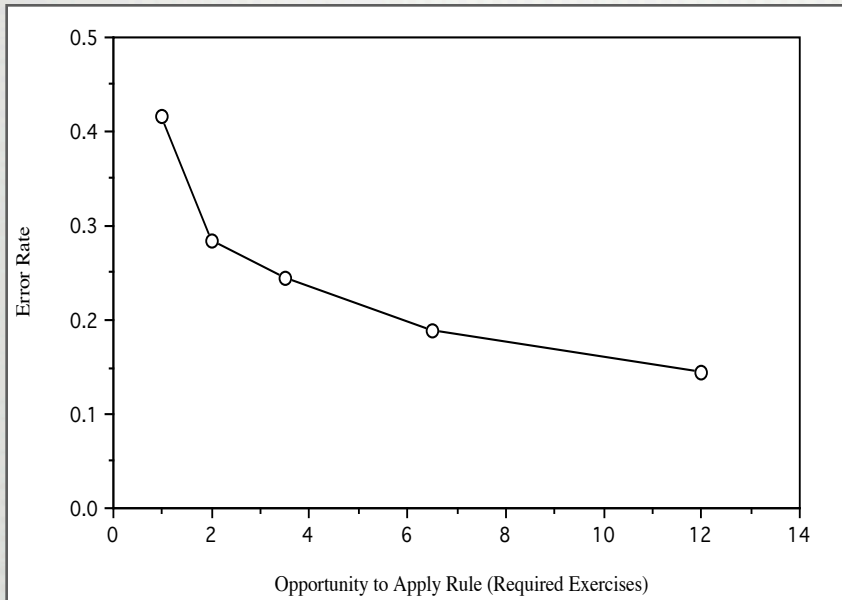
$\theta$  = student mean  
 $-\beta$  = skill initial difficulty  
 $\gamma$  = skill learning rate  
i: student    j: step    k: skill

# VISIBLE MANIFESTATION: THE LEARNING CURVE

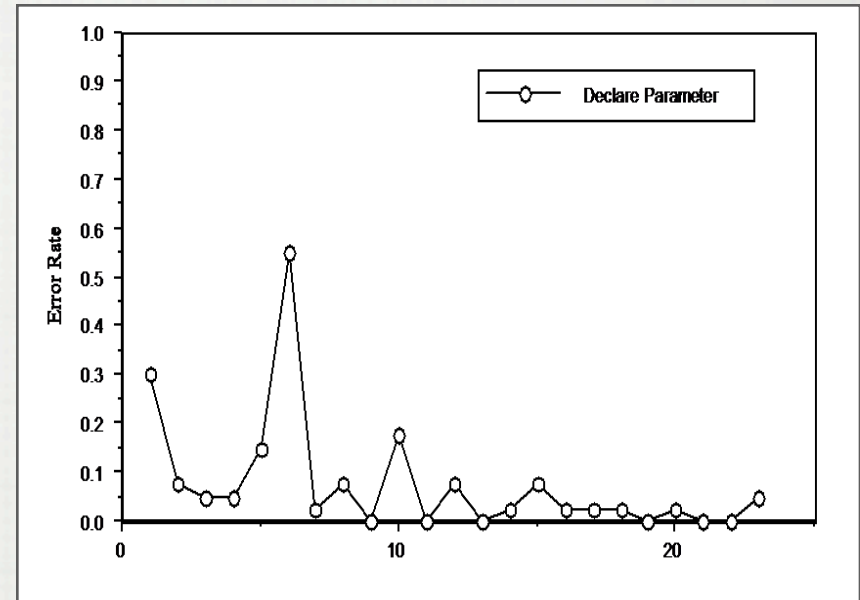


- Good models  $\Rightarrow$  smooth, decreasing curves
- Good models  $\Rightarrow$  accurate predictions

# USING LEARNING CURVES TO EVALUATE A COGNITIVE MODEL



“Good” learning curve  
Model appears to be  
predicting well

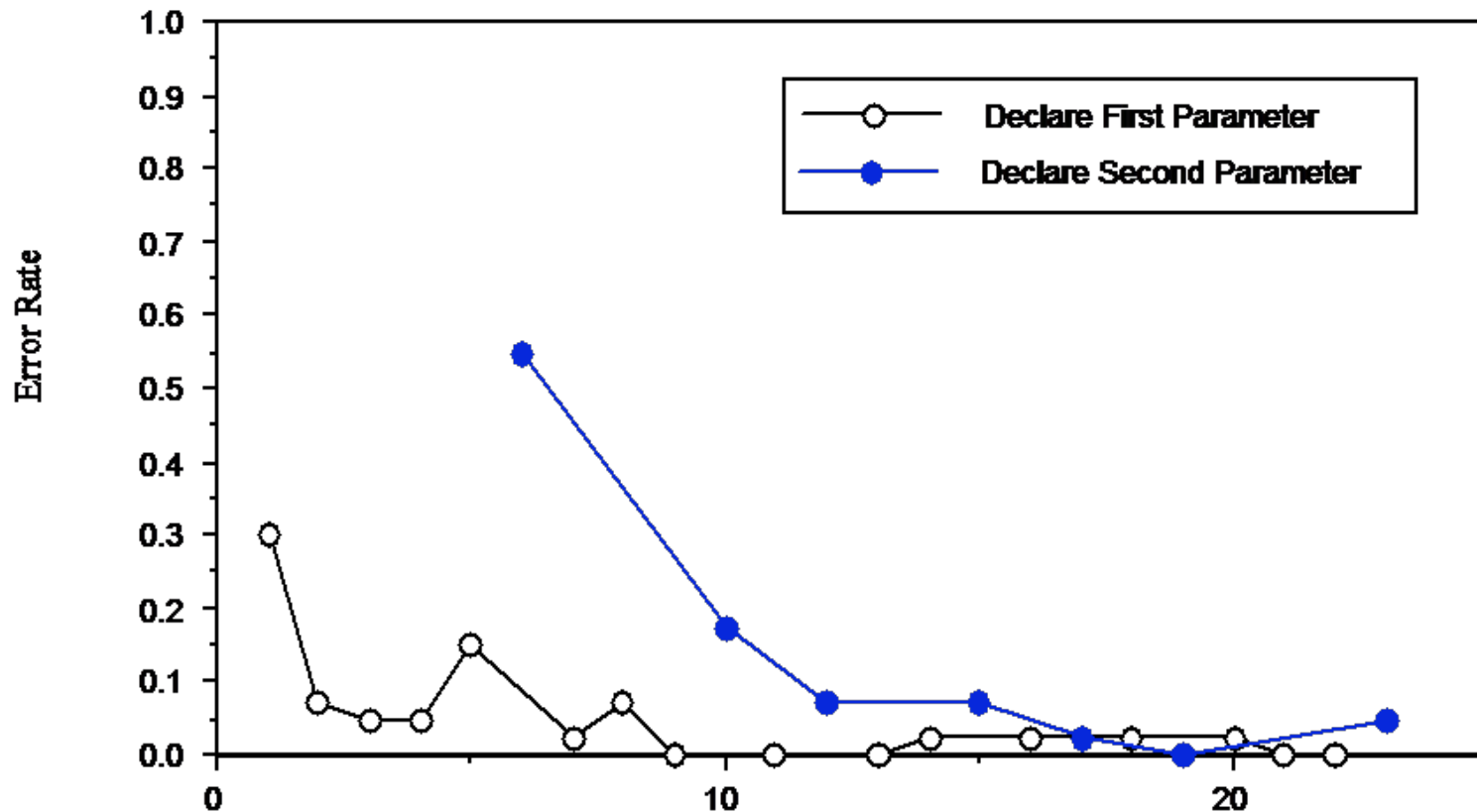


Problematic learning curve  
Model fails to predict  
performance

Corbett, Anderson, O'Brien (1995)

# Modify cognitive model

- Blips occur when a new, unmodeled latent skill appears
- *Split* skill into two new skills



***With new model, tutor can treat these skills separately***

# AUTOMATED DETECTION OF “BLIPS” IN LEARNING CURVES

---

- We identified a latent factor by manually examining learning curves
- Problem:
  - Requires lots of up-front time from expert to define skills
  - Can potentially **discover** automatically that skills are wrong, but can't **fix** automatically
- Can we automate the process of finding latent factors?
  - increase repeatability, reduce bias, reduce human expert time
  - will still need human judgement to connect the identified latents to properties of the problems

# AFM RESULTS

<b>KC Models</b>	<b>KCs</b>	<b>BIC</b>	<b>RMSE</b>
<b>DecompArithDiam</b>	13	5613	0.401
<b>Textbook</b>	10	5678	0.405
<b>Original</b>	15	5762	0.409
<b>Geometry</b>	1	6039	0.427
<b>Unique_step</b>	132	7182	0.403

- “Original”: proposed by domain experts
- “Unique step,” “Geometry”: maximal or minimal splitting
- “Unique step” yields IRT model
- “Textbook”: discovered by an automated model-search technique
- “DecompArithDiam”: we discovered by manual search **(best)**
- made possible by visualization and analysis tools in DataShop

# AFM RESULTS

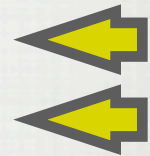
<b>KC Models</b>	<b>KCs</b>	<b>BIC</b>	<b>RMSE</b>
<b>DecompArithDiam</b>	13	5613	0.401
<b>Textbook</b>	10	5678	0.405
<b>Original</b>	15	5762	0.409
<b>Geometry</b>	1	6039	0.427
<b>Unique_step</b>	132	7182	0.403



- “Original”: proposed by domain experts
- “Unique step,” “Geometry”: maximal or minimal splitting
- “Unique step” yields IRT model
- “Textbook”: discovered by an automated model-search technique
- “DecompArithDiam”: we discovered by manual search **(best)**
- made possible by visualization and analysis tools in DataShop

# AFM RESULTS

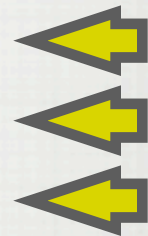
<b>KC Models</b>	<b>KCs</b>	<b>BIC</b>	<b>RMSE</b>
<b>DecompArithDiam</b>	13	5613	0.401
<b>Textbook</b>	10	5678	0.405
<b>Original</b>	15	5762	0.409
<b>Geometry</b>	1	6039	0.427
<b>Unique_step</b>	132	7182	0.403



- “Original”: proposed by domain experts
- “Unique step,” “Geometry”: maximal or minimal splitting
- “Unique step” yields IRT model
- “Textbook”: discovered by an automated model-search technique
- “DecompArithDiam”: we discovered by manual search **(best)**
- made possible by visualization and analysis tools in DataShop

# AFM RESULTS

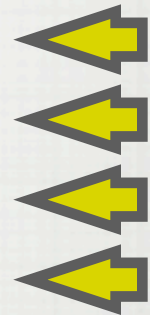
<b>KC Models</b>	<b>KCs</b>	<b>BIC</b>	<b>RMSE</b>
<b>DecompArithDiam</b>	13	5613	0.401
<b>Textbook</b>	10	5678	0.405
<b>Original</b>	15	5762	0.409
<b>Geometry</b>	1	6039	0.427
<b>Unique_step</b>	132	7182	0.403



- “Original”: proposed by domain experts
- “Unique step,” “Geometry”: maximal or minimal splitting
- “Unique step” yields IRT model
- “Textbook”: discovered by an automated model-search technique
- “DecompArithDiam”: we discovered by manual search **(best)**
- made possible by visualization and analysis tools in DataShop

# AFM RESULTS

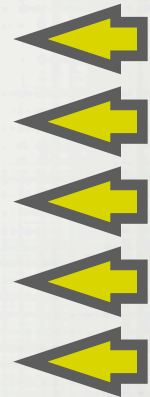
<b>KC Models</b>	<b>KCs</b>	<b>BIC</b>	<b>RMSE</b>
<b>DecompArithDiam</b>	13	5613	0.401
<b>Textbook</b>	10	5678	0.405
<b>Original</b>	15	5762	0.409
<b>Geometry</b>	1	6039	0.427
<b>Unique_step</b>	132	7182	0.403



- “Original”: proposed by domain experts
- “Unique step,” “Geometry”: maximal or minimal splitting
- “Unique step” yields IRT model
- “Textbook”: discovered by an automated model-search technique
- “DecompArithDiam”: we discovered by manual search **(best)**
- made possible by visualization and analysis tools in DataShop

# AFM RESULTS

KC Models	KCs	BIC	RMSE
<b>DecompArithDiam</b>	13	5613	0.401
<b>Textbook</b>	10	5678	0.405
<b>Original</b>	15	5762	0.409
<b>Geometry</b>	1	6039	0.427
<b>Unique_step</b>	132	7182	0.403



- “Original”: proposed by domain experts
- “Unique step,” “Geometry”: maximal or minimal splitting
- “Unique step” yields IRT model
- “Textbook”: discovered by an automated model-search technique
- “DecompArithDiam”: we discovered by manual search **(best)**
- made possible by visualization and analysis tools in DataShop

# CAN WE GENERALIZE AFM?

$$\text{AFM: } \log \frac{p_{ij}}{1 - p_{ij}} = \theta_i + \sum_k \beta_k Q_{kj} + \sum_k Q_{kj} (\gamma_k T_{ik})$$

$$\log \left( \frac{p_{ij}}{1 - p_{ij}} \right) = \sum_k U_{ik} V_{jk}$$

$i, j, k$  = student, item, skill

$p$  = probability correct

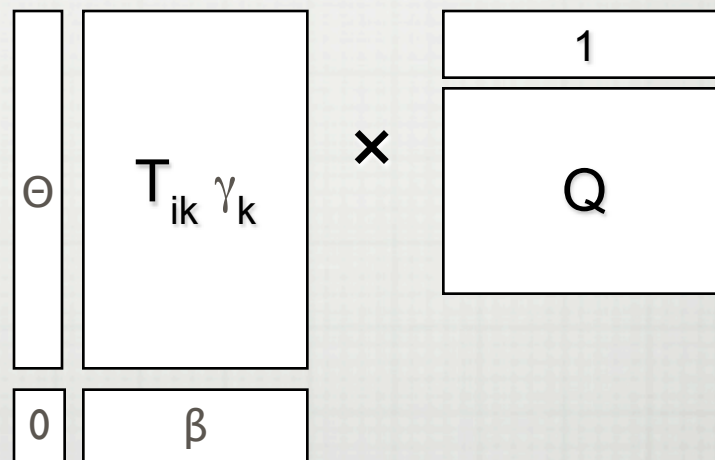
$\theta$  = student overall performance

$\beta$  = skill easiness / difficulty

$Q$  = item  $\times$  skill matrix

$\gamma$  = skill practice slope

$T$  = number of practice opportunities



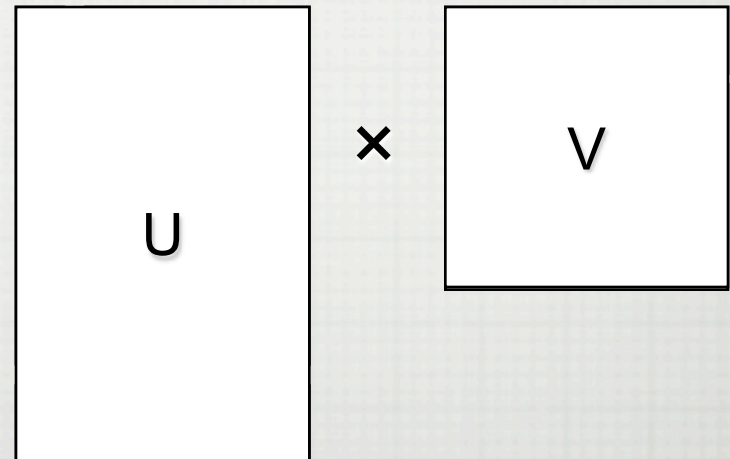
# CAN WE GENERALIZE AFM?

$$\text{AFM: } \log \frac{p_{ij}}{1 - p_{ij}} = \theta_i + \sum_k \beta_k Q_{kj} + \sum_k Q_{kj} (\gamma_k T_{ik})$$

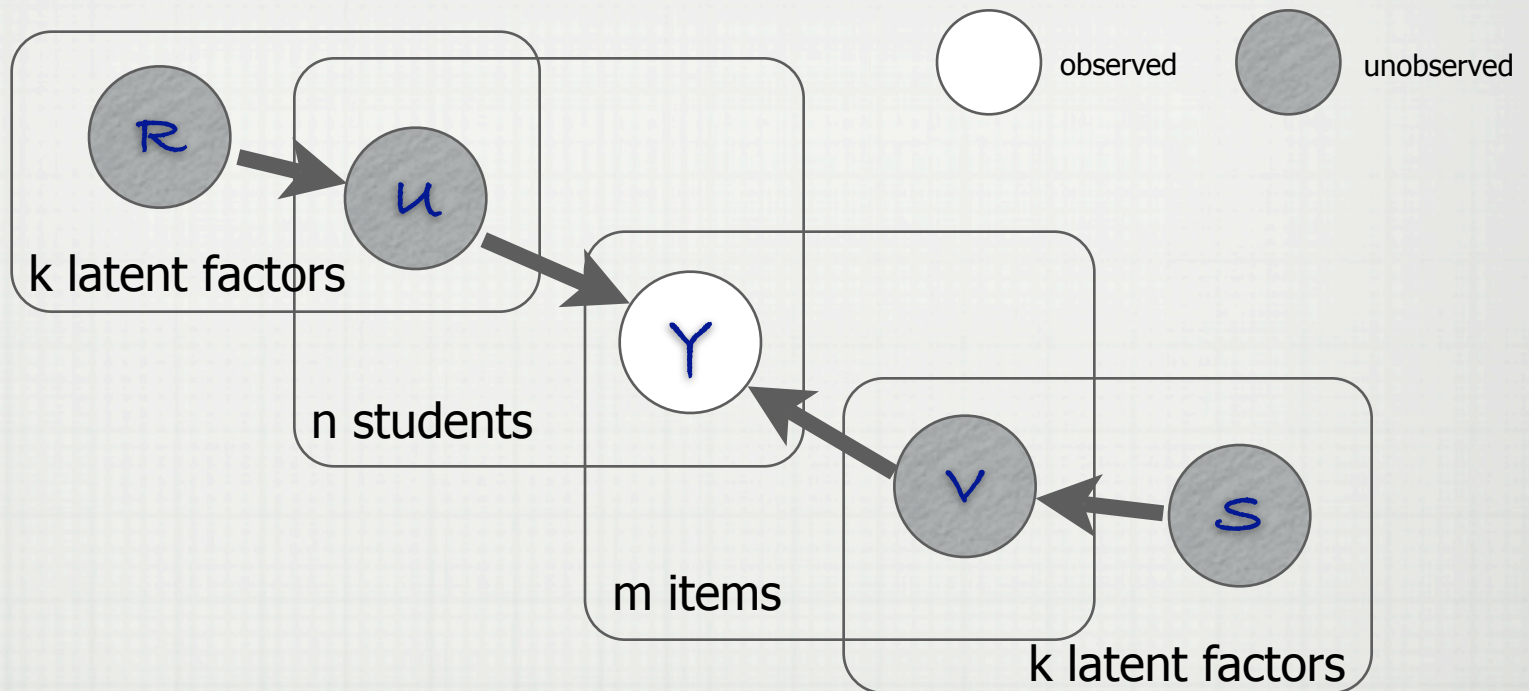
$$\log \left( \frac{p_{ij}}{1 - p_{ij}} \right) = \sum_k U_{ik} V_{jk}$$

*Relax constraints on U,V*

$i, j, k$  = student, item, skill  
 $p$  = probability correct  
 $\theta$  = student overall performance  
 $\beta$  = skill easiness/difficulty  
 $Q$  = item  $\times$  skill matrix  
 $\gamma$  = skill easiness slope  
 $T$  = number of practice opportunities



# HIERARCHICAL LOGISTIC PRINCIPAL COMPONENTS ANALYSIS



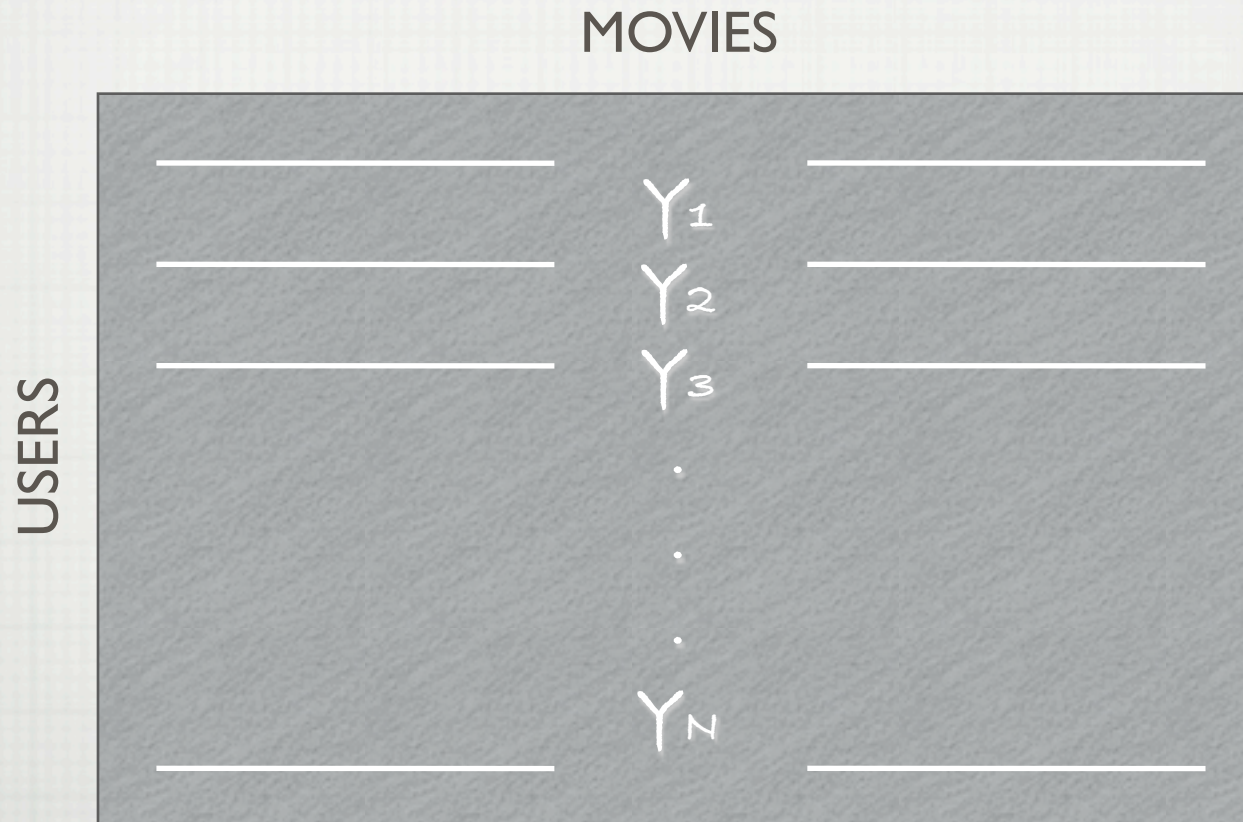
- U: student latent factors
- V: item latent factors
- Y: observed performance
- R: shared prior for student latents
- S: shared prior for item latents

$$\log \left( \frac{p_{ij}}{1 - p_{ij}} \right) = \sum_k U_{ik} V_{jk}$$

↑ student factor
 ↑ item factor

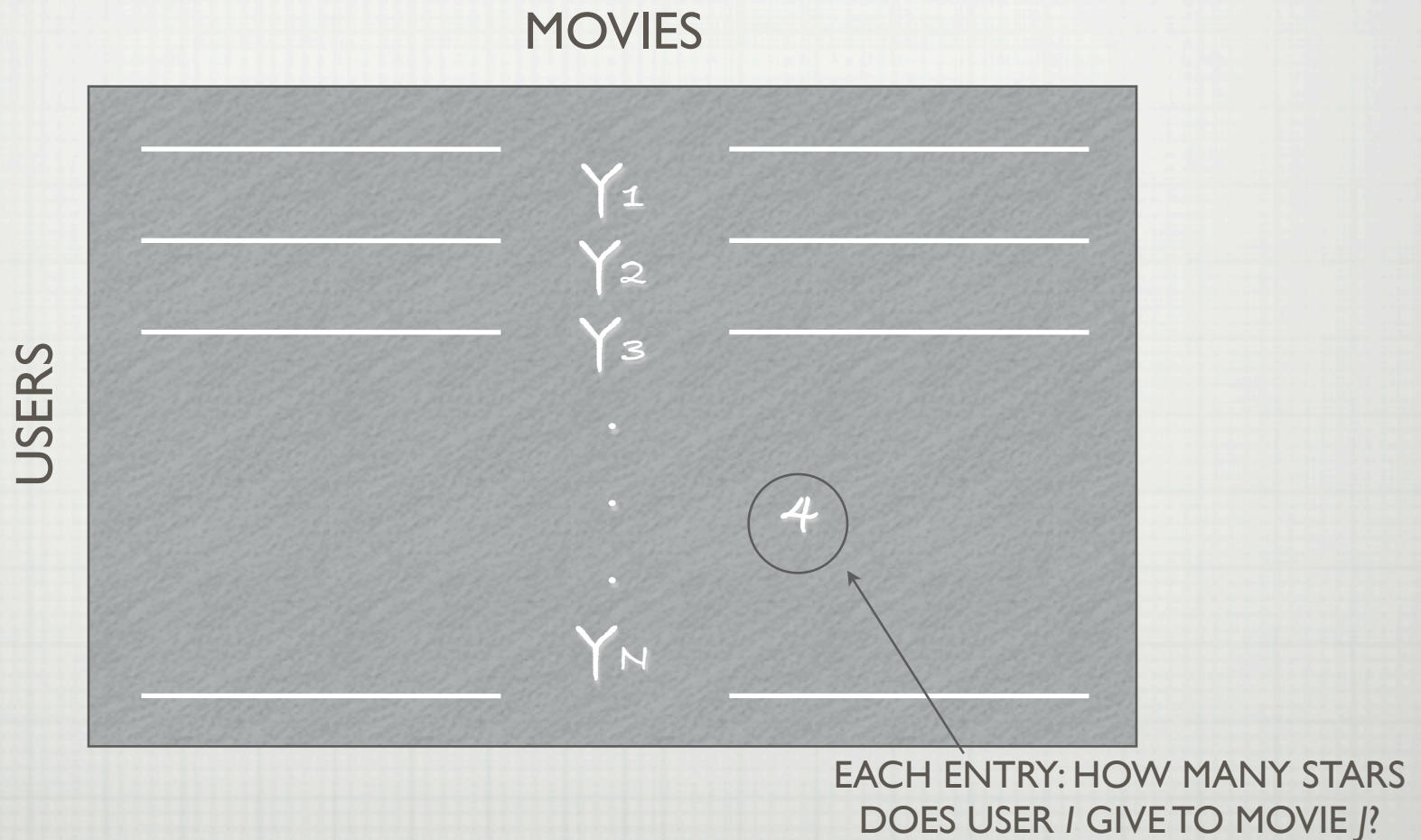
# PCA IS A WIDELY USED AND SUCCESSFUL MODEL

---



# PCA IS A WIDELY USED AND SUCCESSFUL MODEL

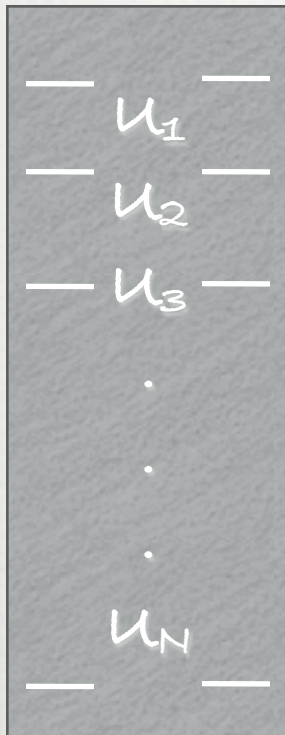
---



# RESULT OF FACTORING

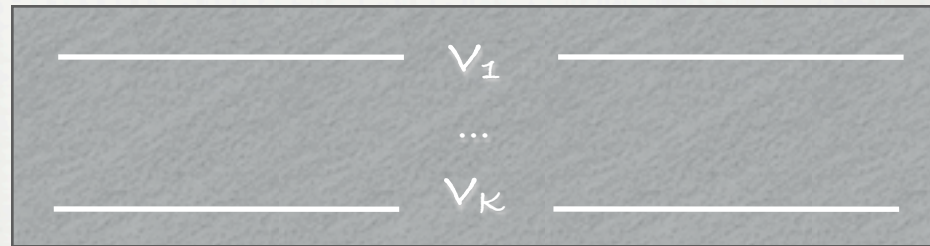
BASIS WEIGHTS

USERS



BASIS VECTORS

MOVIES



Low-d basis = latent variables

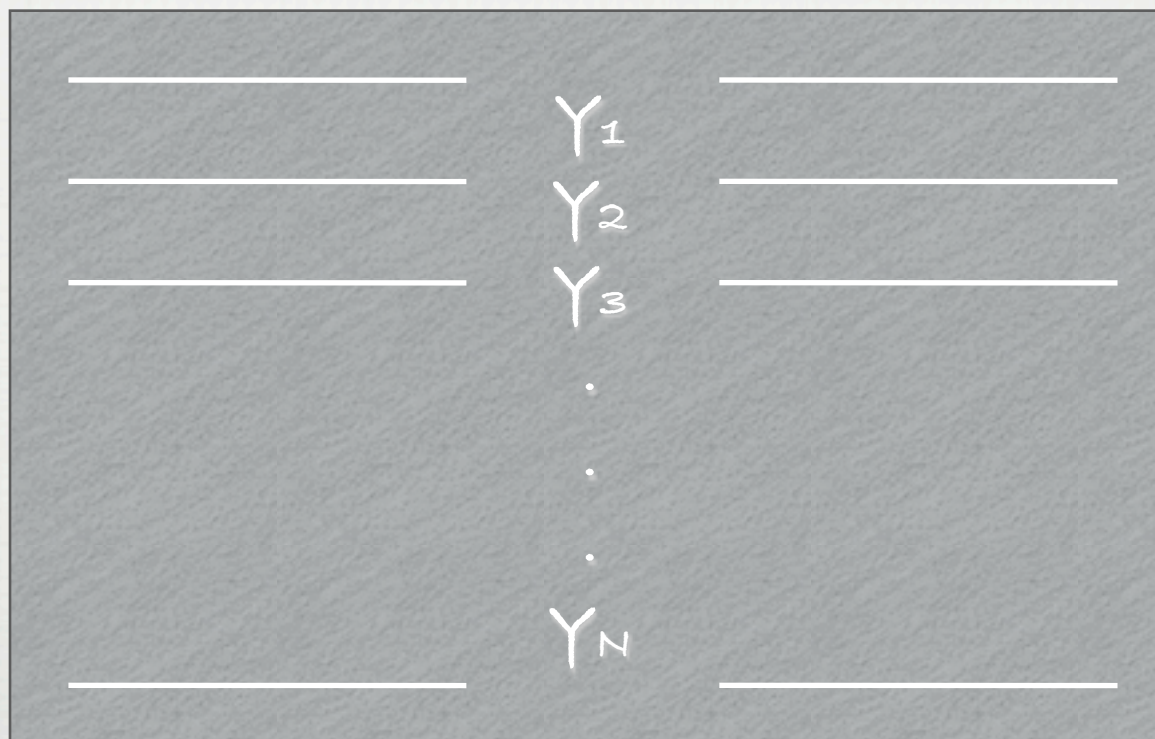
Basis vectors represent latent properties of movies, e.g., “is a comedy”

# IN OUR CASE: STUDENT-ITEM DATA

---

ITEMS IN TUTOR

STUDENTS

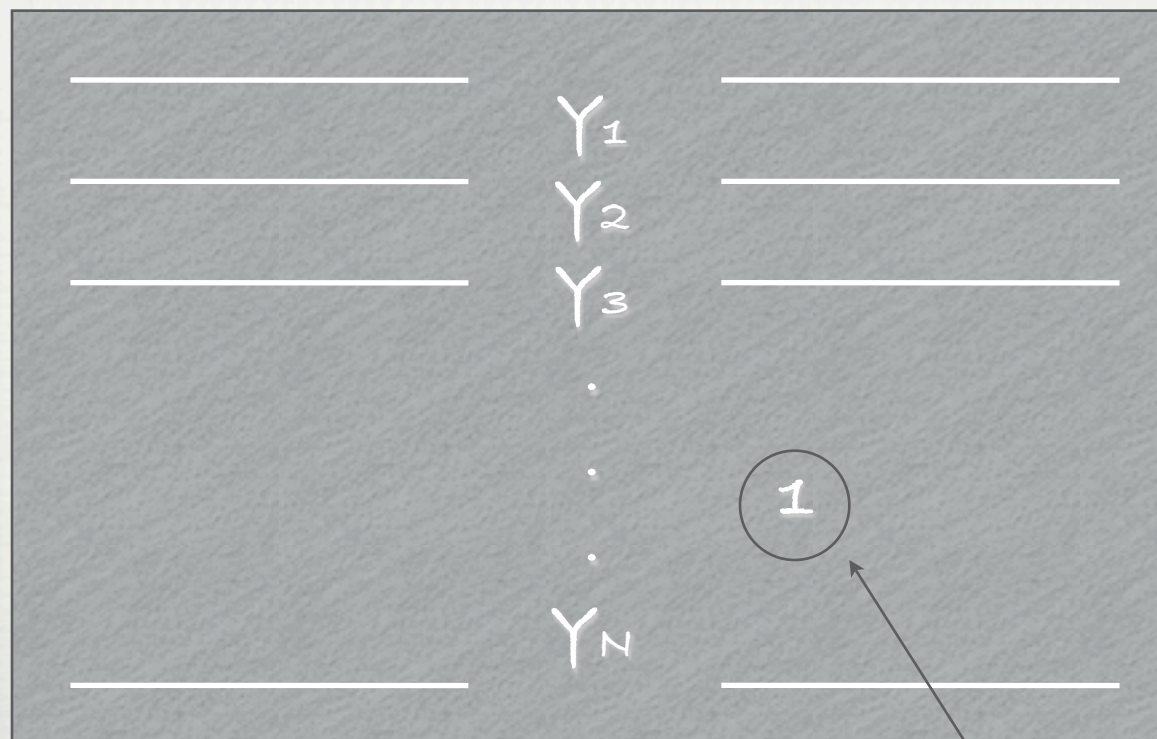


# IN OUR CASE: STUDENT-ITEM DATA

---

## ITEMS IN TUTOR

STUDENTS



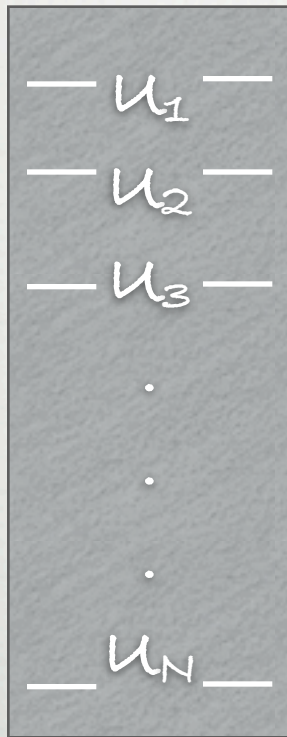
EACH ENTRY: DOES STUDENT  $i$  GET  
ITEM  $j$  RIGHT?

# RESULT OF FACTORING

---

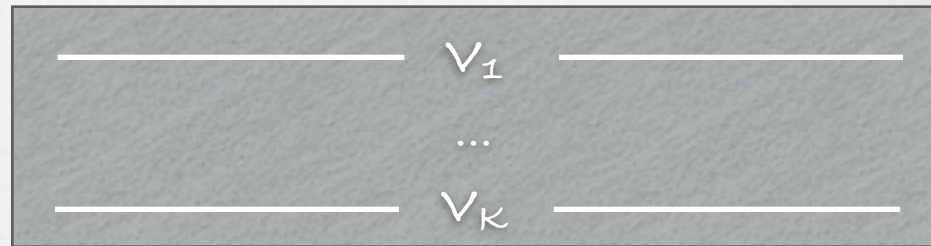
BASIS WEIGHTS

STUDENTS



BASIS VECTORS

ITEMS



Basis vectors are candidate  
“eigenskills”

Weights are students’  
knowledge levels

# PCA VS LOGISTIC PCA

- Ordinary PCA: linear, Gaussian
- Logistic PCA: can handle conjunctive, disjunctive skills

NONLINEARITY:  
CONJUNCTIVE SKILLS

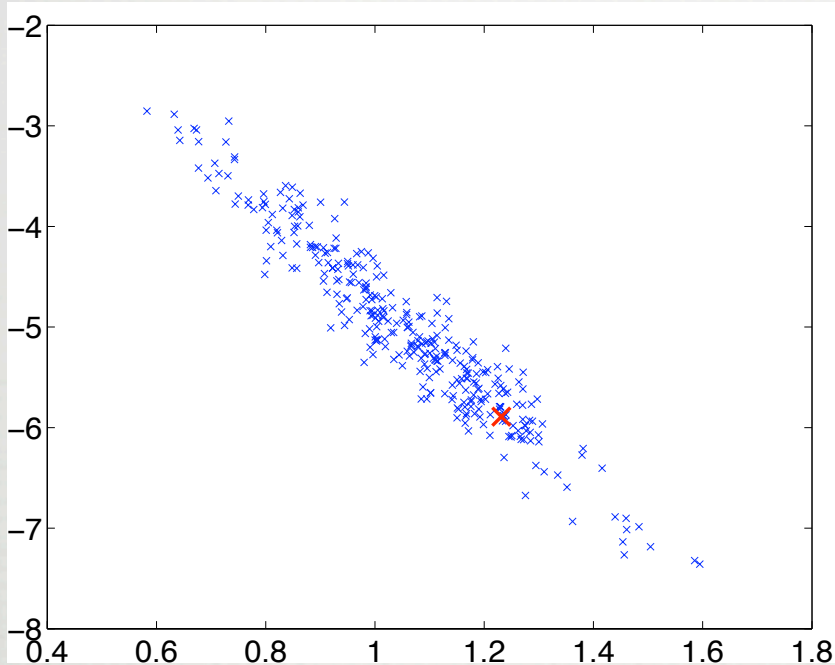


# BAYESIAN INFERENCE

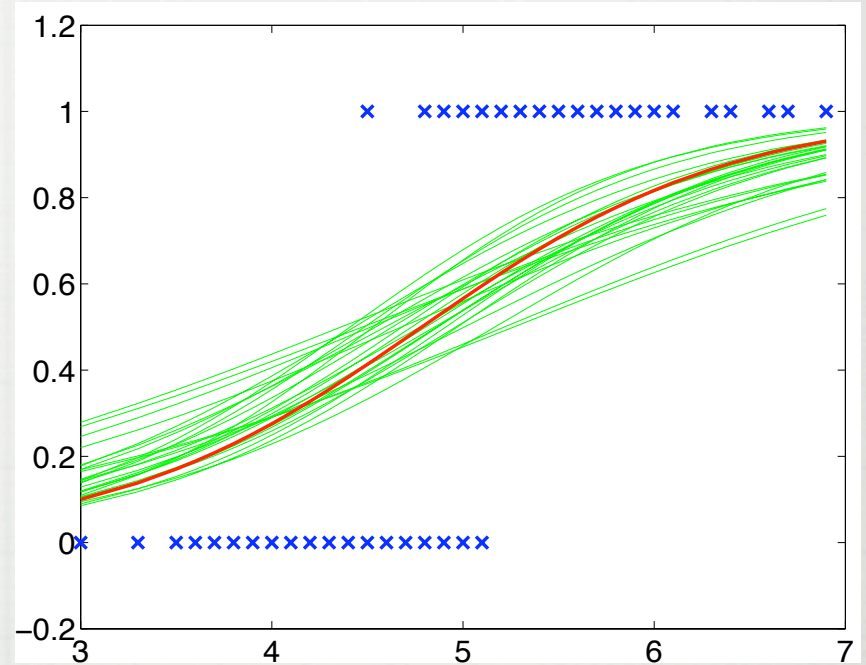
---

- Ordinary PCA yields maximum-likelihood estimate
- Good, right?
  - sadly, the usual reasons to want the MLE don't apply here
  - e.g., consistency: variance and bias of estimates of U and V **do not** approach 0 (unless #items/student and #students/item  $\rightarrow \infty$ )
- Result: MLE is typically far too confident of itself

# TOO CERTAIN: EXAMPLE



Learned coefficients  
(e.g., a column of  $V$ )



Predictions

# RESULT: “FOLD-IN PROBLEM”

---

- Nonsensical results when trying to apply learned model to a new student or item
- Similar to **overfitting** problem in supervised learning: confident-but-wrong parameters do not generalize to new examples
- Unlike overfitting, fold-in problem doesn't necessarily go away with more data

# EXPERIMENTAL COMPARISON GEOMETRY AREA 1996-1997 DATA

	item 1	item 2	item 3	item 4	item 5
student 1		1		0	1
student 2	0		1	1	
student 3	0	0			1
student 4		1			0
student 5	0			0	
student 6		1	1		1
student 7	1	0	1	0	

## Legend

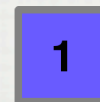


student not  
presented  
with item



student  
answered  
the item

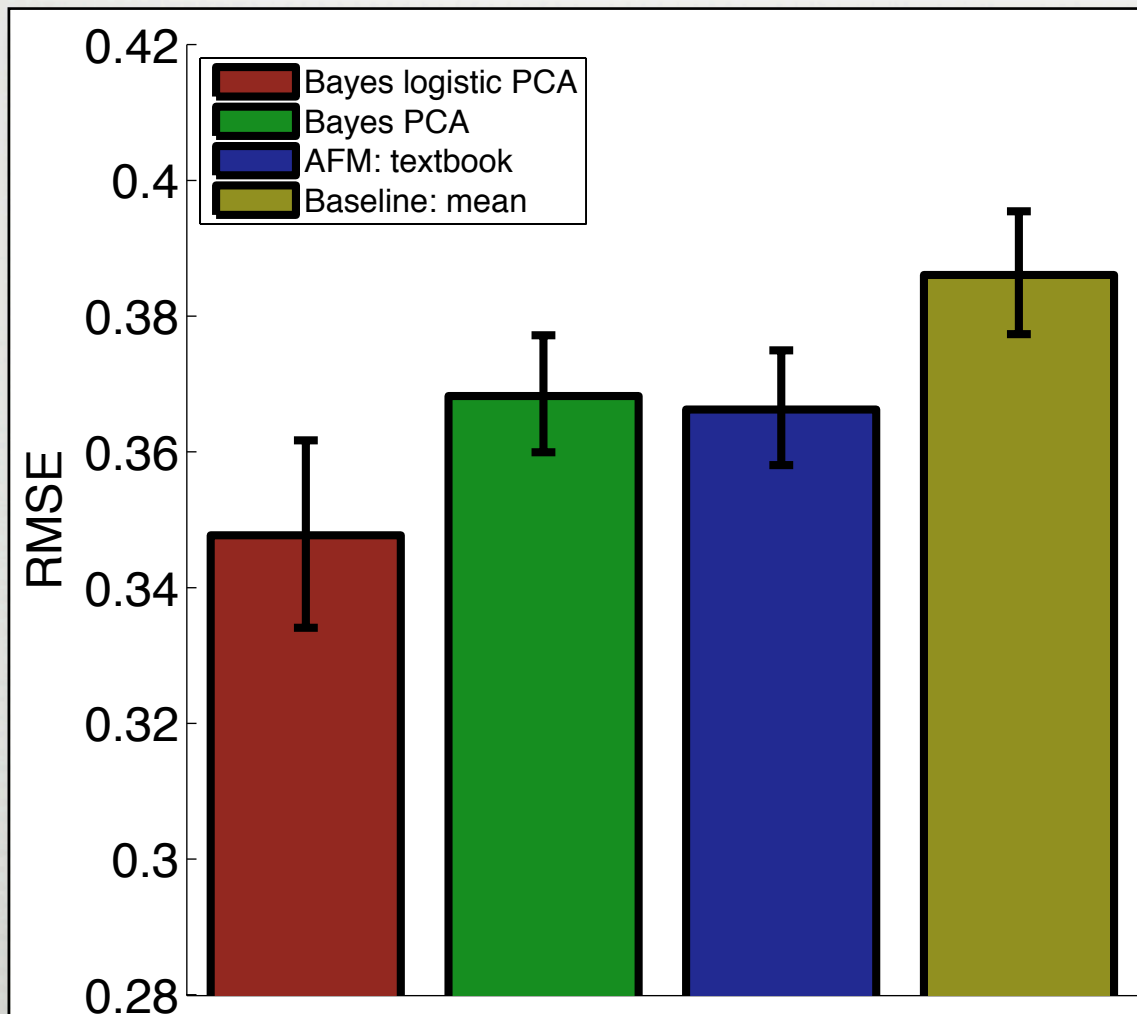
0 = WRONG  
1 = RIGHT



student  
answered  
the item, but  
we hide the  
answer

- Geometry tutor: 139 items presented to 59 students
- On average, each student tested on 60 items

# RESULTS: HOLD-OUT ERROR



Non-Bayesian PCA/  
LPCA performs at about  
chance level in similar  
problems

Embedding dimension  
is  $k = 15$ , except PCA  
+AFM where  $k = 1$

Credit for  
logistic PCA:  
Ajit Singh

# STILL MISSING

---

- A way to include **time** in PCA
- A way to encourage **interpretable** latent models
- A way to take advantage of **partial prior knowledge** of model