Latent Factor Models

Geoff Gordon, John Stamper

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

Cognitive tutors and models

- Cognitive Model: A system that can solve problems in the various ways students can

$$3(2x - 5) = 9$$

If goal is solve a(bx+c) = d
Then rewrite as abx + ac = d

If goal is solve a(bx+c) = d
Then rewrite as bx+c = d/a

If goal is solve a(bx+c) = d
Then rewrite as abx + c = d

$$6x - 15 = 9 \qquad 2x - 5 = 3 \qquad 6x - 5 = 9$$

- Model Tracing: follow students through individual approaches to a problem ⇒ context-sensitive instruction

Cognitive tutors and models

- Cognitive Model: A system that can solve problems in the various ways students can

$3(2x - 5) = 9$

If goal is solve a(bx+c) = d
Then rewrite as abx + ac = d

If goal is solve a(bx+c) = d
Then rewrite as bx+c = d/a

If goal is solve a(bx+c) = d
Then rewrite as abx + c = d

skills

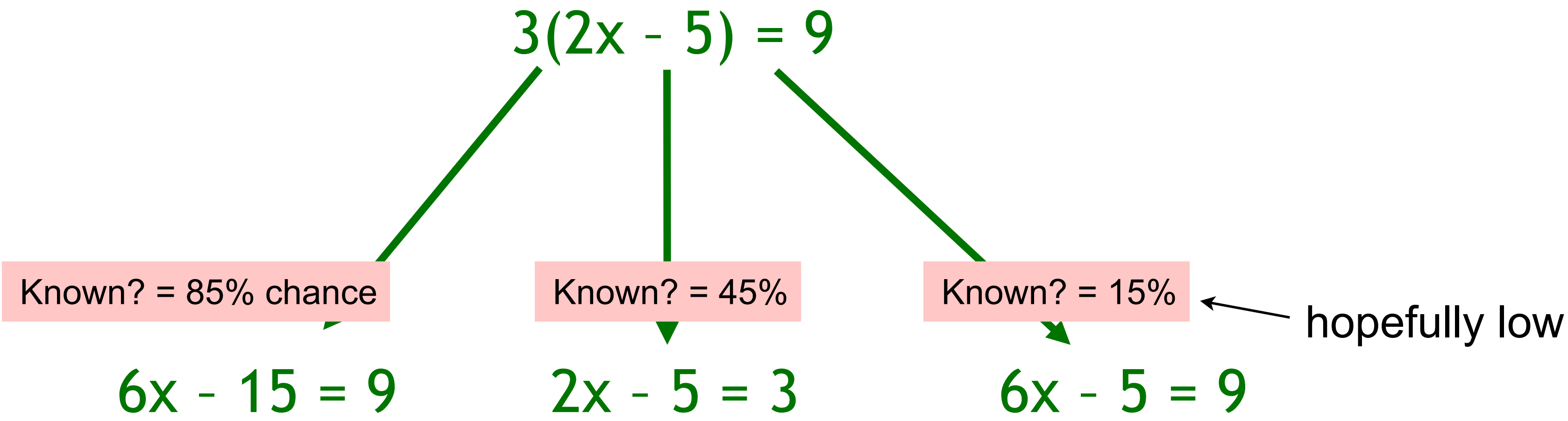
bug

$6x - 15 = 9$ $\quad 2x - 5 = 3$ $\quad 6x - 5 = 9$

- Model Tracing: follow students through individual approaches to a problem ⇒ context-sensitive instruction

Cognitive tutors and models

- Cognitive Model: A system that can solve problems in the various ways students can



- Model Tracing: follow students through individual approaches to a problem ⇒ context-sensitive instruction
- Knowledge Tracing: Assess student's knowledge growth ⇒ individualized activity selection and pacing

Cognitive tutors and models

- Cognitive Model: A system that can solve problems in the various ways students can

$$3(2x - 5) = 9$$

Hint message: "Distribute *a* across the parentheses."

Bug message: "You need to multiply *c* by *a* also."

Known? = 85% chance

Known? = 45%

Known? = 15%

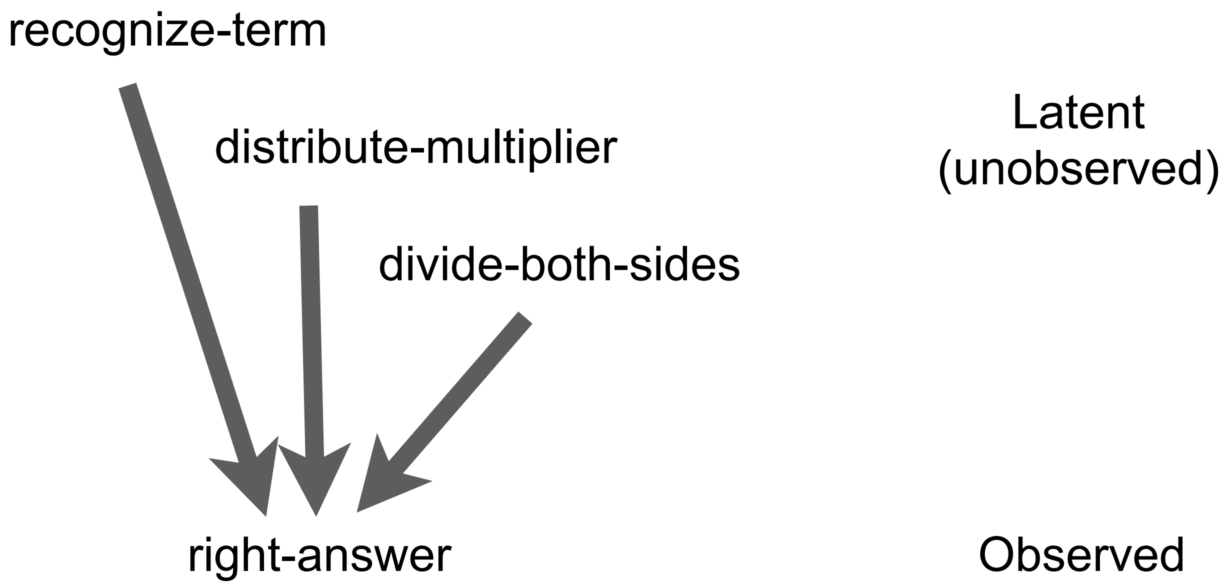
hopefully low

$$6x - 15 = 9 \qquad 2x - 5 = 3 \qquad 6x - 5 = 9$$

- Model Tracing: follow students through individual approaches to a problem ⇒ context-sensitive instruction
- Knowledge Tracing: Assess student's knowledge growth ⇒ individualized activity selection and pacing

Latent Factors

- Model tracks ***what skills*** student currently knows—***latent factors***



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!

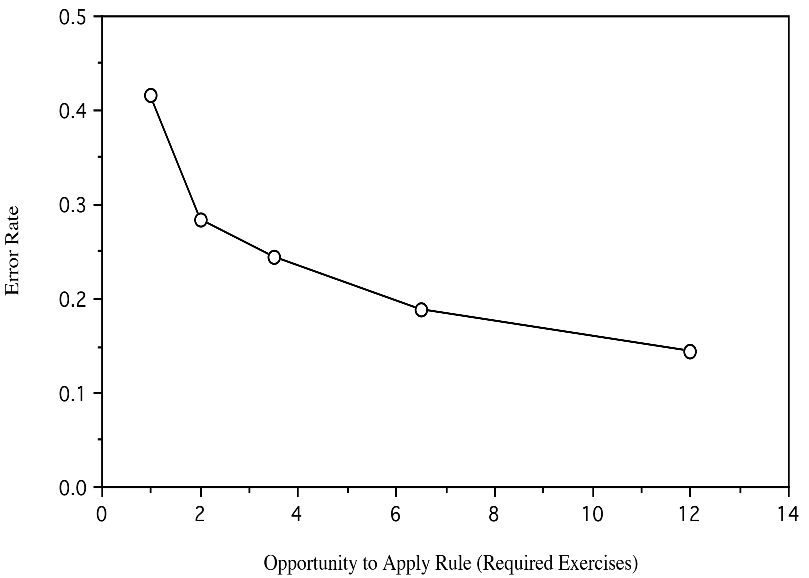
- *Cognitive model determines instruction*
  - Through instructional decisions like problem selection, hints, …
- A correct model is one that is consistent with student behavior, predicting *task difficulty* and *transfer between instruction and test*
- *Cognitive models ~~are~~ discovered not designed*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

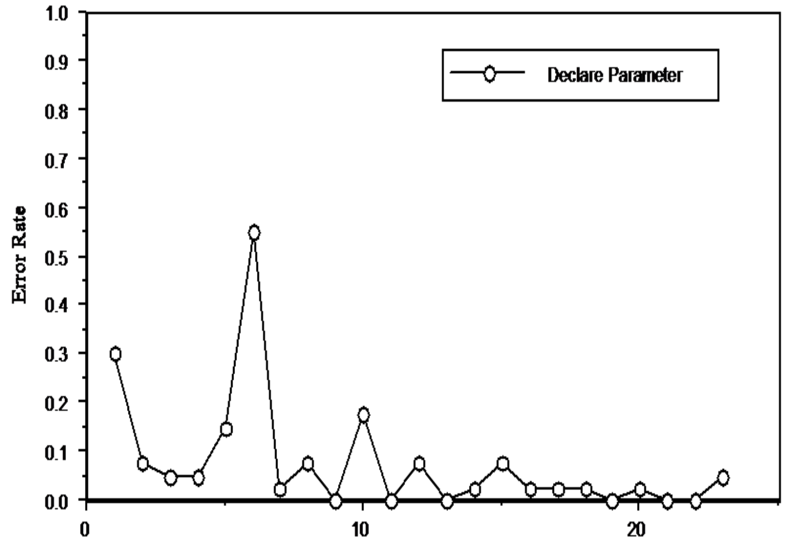
⇒ Huge data mining opportunity

It’s not easy

- Student models are a key bottleneck in cognitive tutor authoring and performance
  - rough estimate: 20-80 hrs to hand-code model for 1 hr of content
  - result may be too simple, not rigorously verified
- But, demonstrated improvements in learning from better models
  - E.g., Cen et al [2007]:12% less time to learn 6 geometry units (same retention) using tutor w/ more accurate model

Using learning curves to evaluate a cognitive model





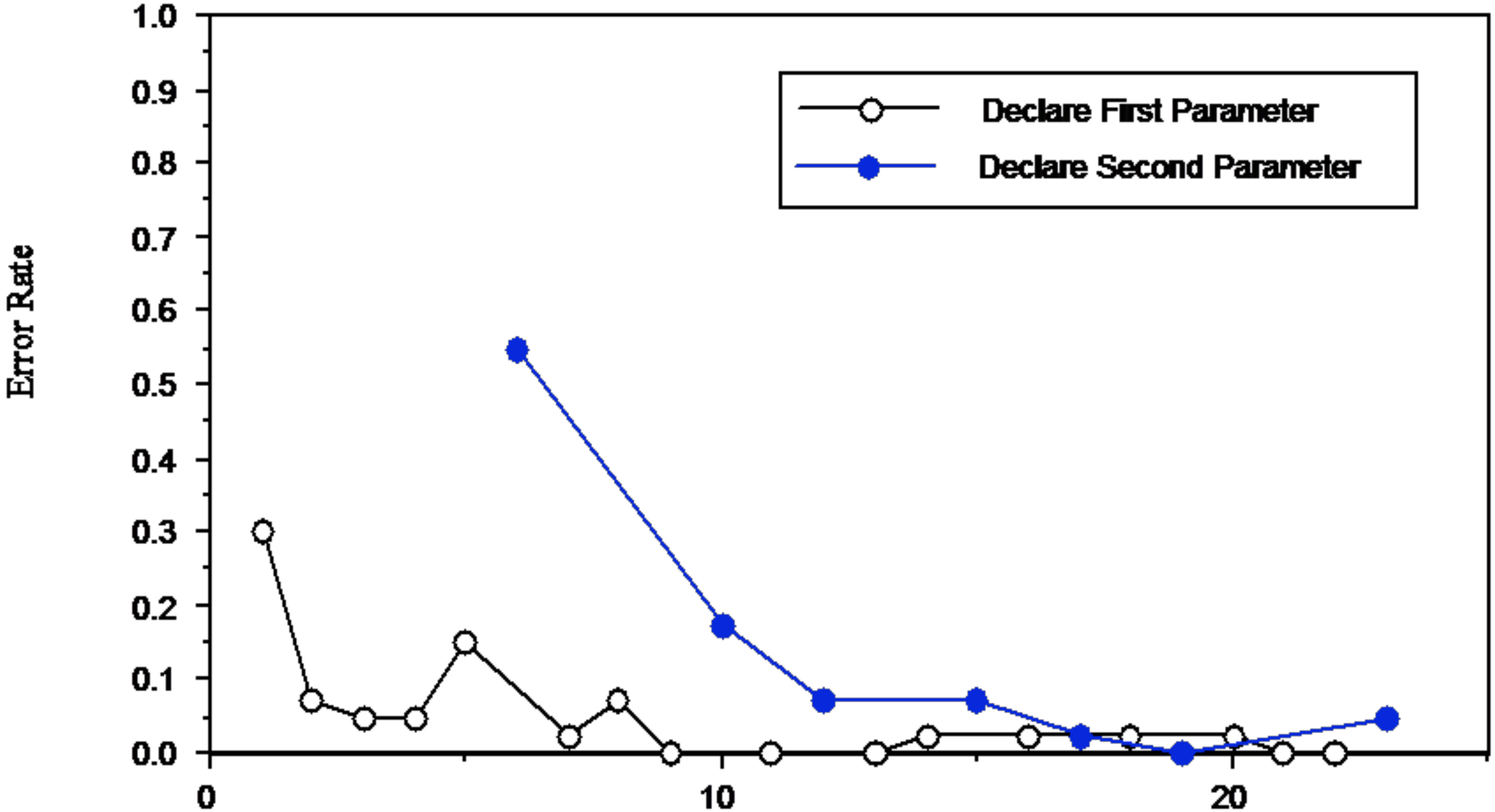
A "good" learning curve
Model appears to be working well

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
- 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

Statistical model of learning curves

- Additive Factor Model (Draney et al., 1995)
- Logistic regression model of P(correct answer | skill info)

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

Statistical model of learning curves

- Additive Factor Model (Draney et al., 1995)
- Logistic regression model of P(correct answer | skill info)

Correct?

Step i uses skill k

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

Opp count

Statistical model of learning curves

- Additive Factor Model (Draney et al., 1995)
- Logistic regression model of P(correct answer | skill info)

Correct? Parameters Step i uses skill k

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

Opp count

θ = student mean

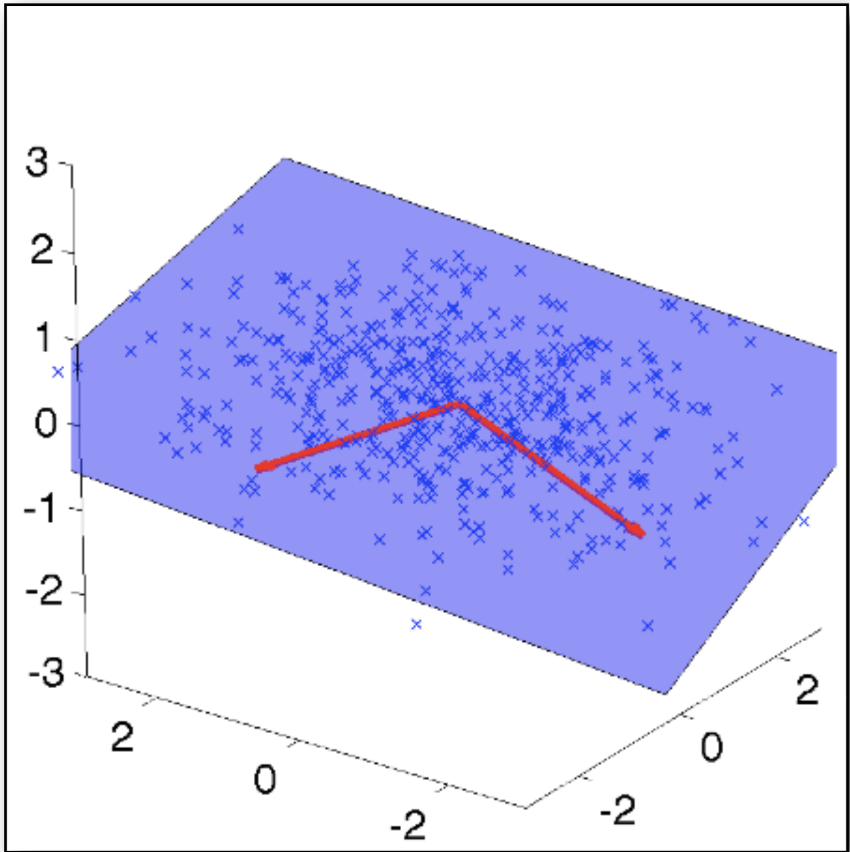
–β = skill difficulty

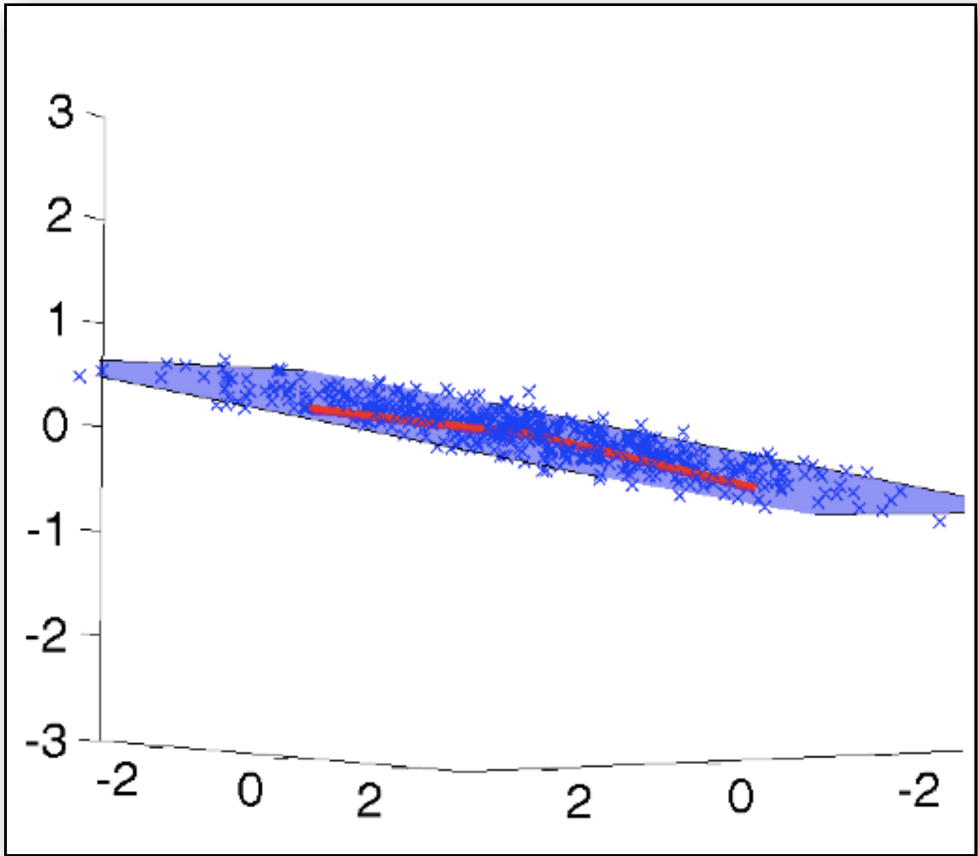
γ = skill learning rate

AFM problems

- Requires a lot of up-front time from expert to define skills
- Can potentially ***discover*** automatically that skills are wrong, but can't ***fix*** automatically

A possible answer: PCA





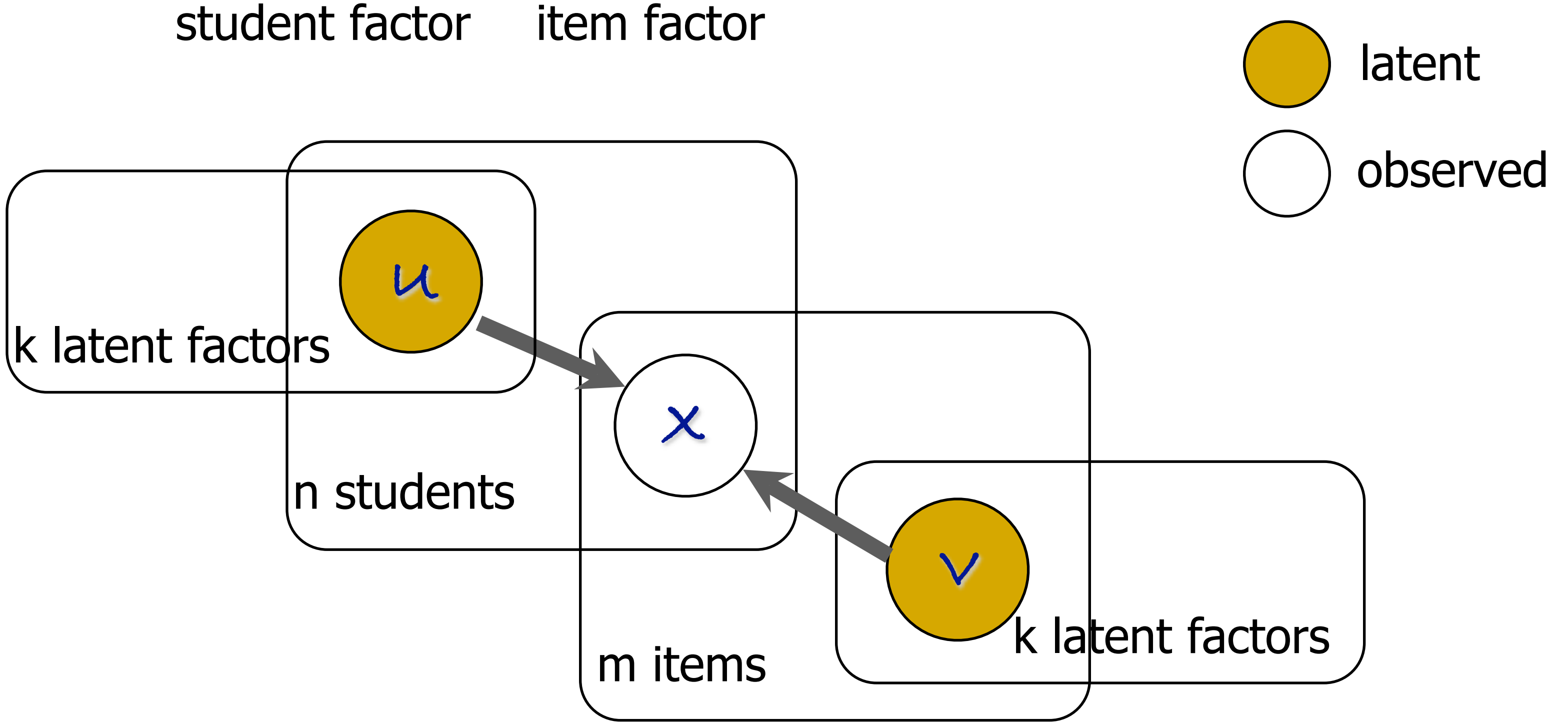
PCA: estimates latent factors under linear-Gaussian assumption

PCA: the model

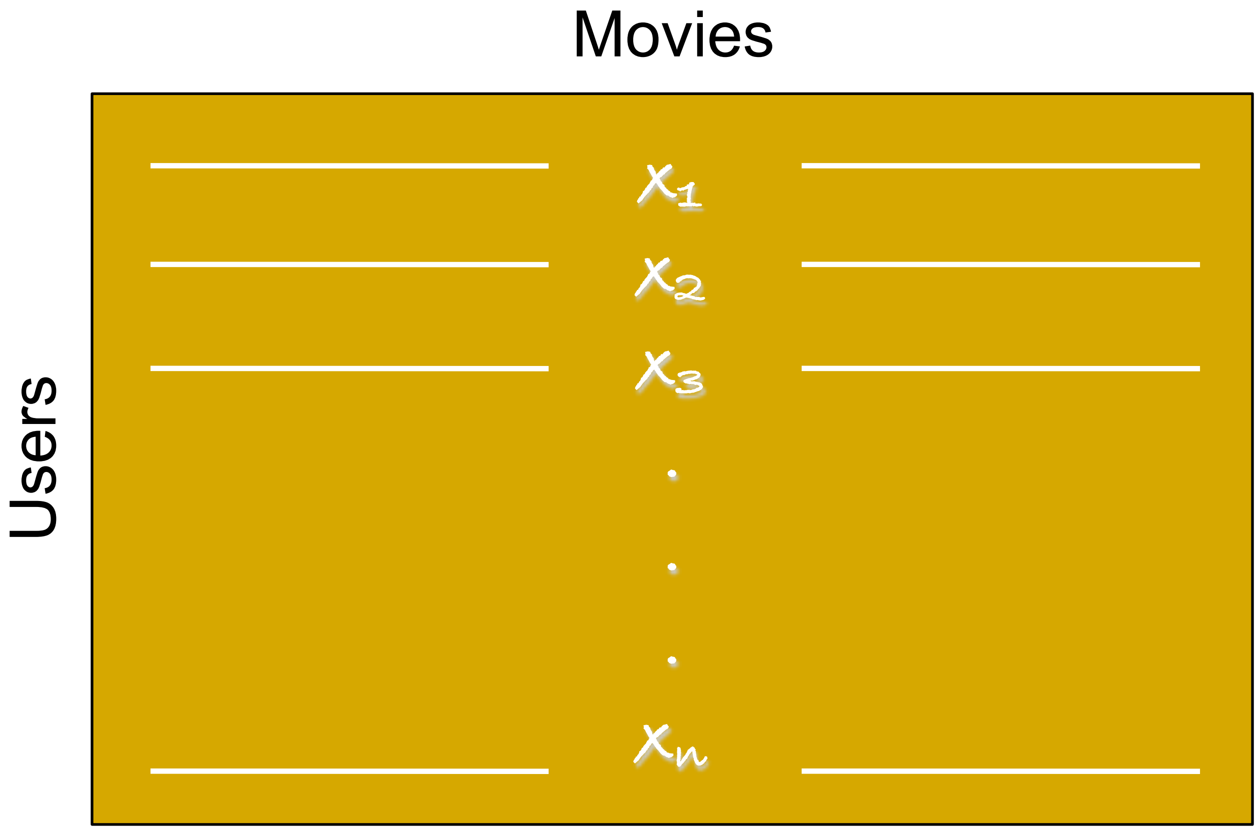
$$\mathbb{E}(X_{ij} \mid U, V) = \sum_k U_{ik} V_{jk}$$

student factor item factor

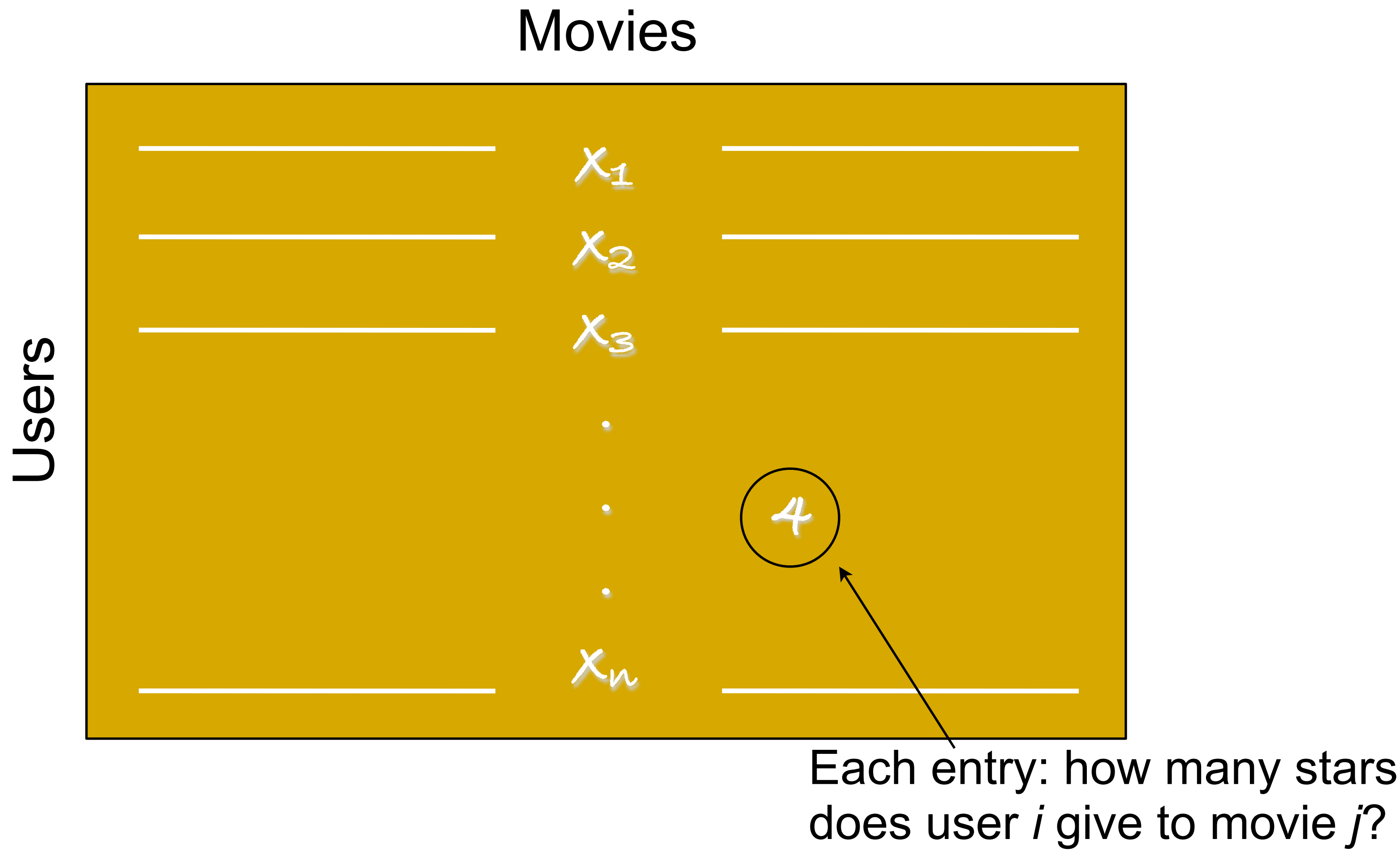
U: Gaussian (0 mean, fixed var)
V: Gaussian (0 mean, fixed var)
X: Gaussian (fixed var, mean at left)



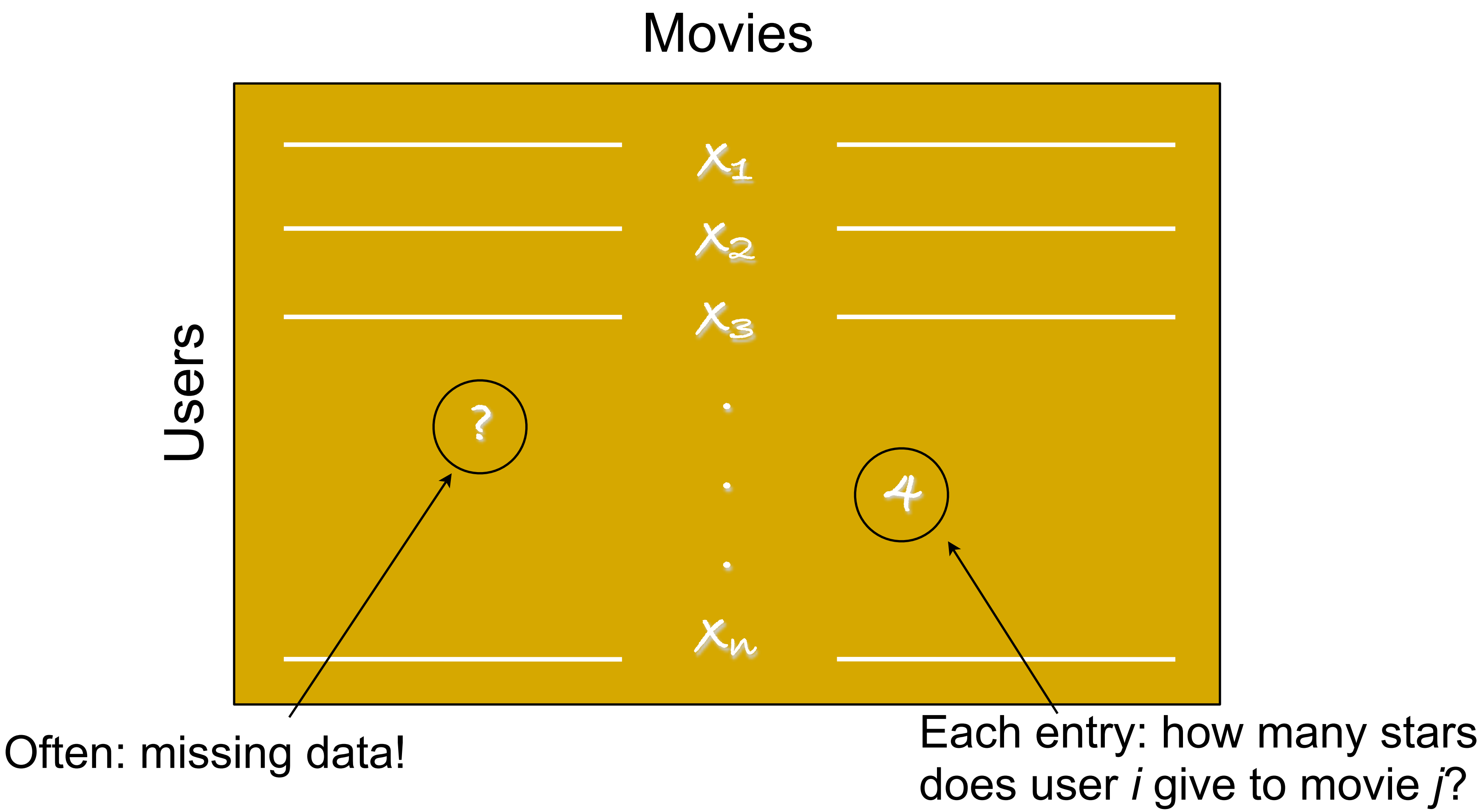
PCA is a widely used and successful model



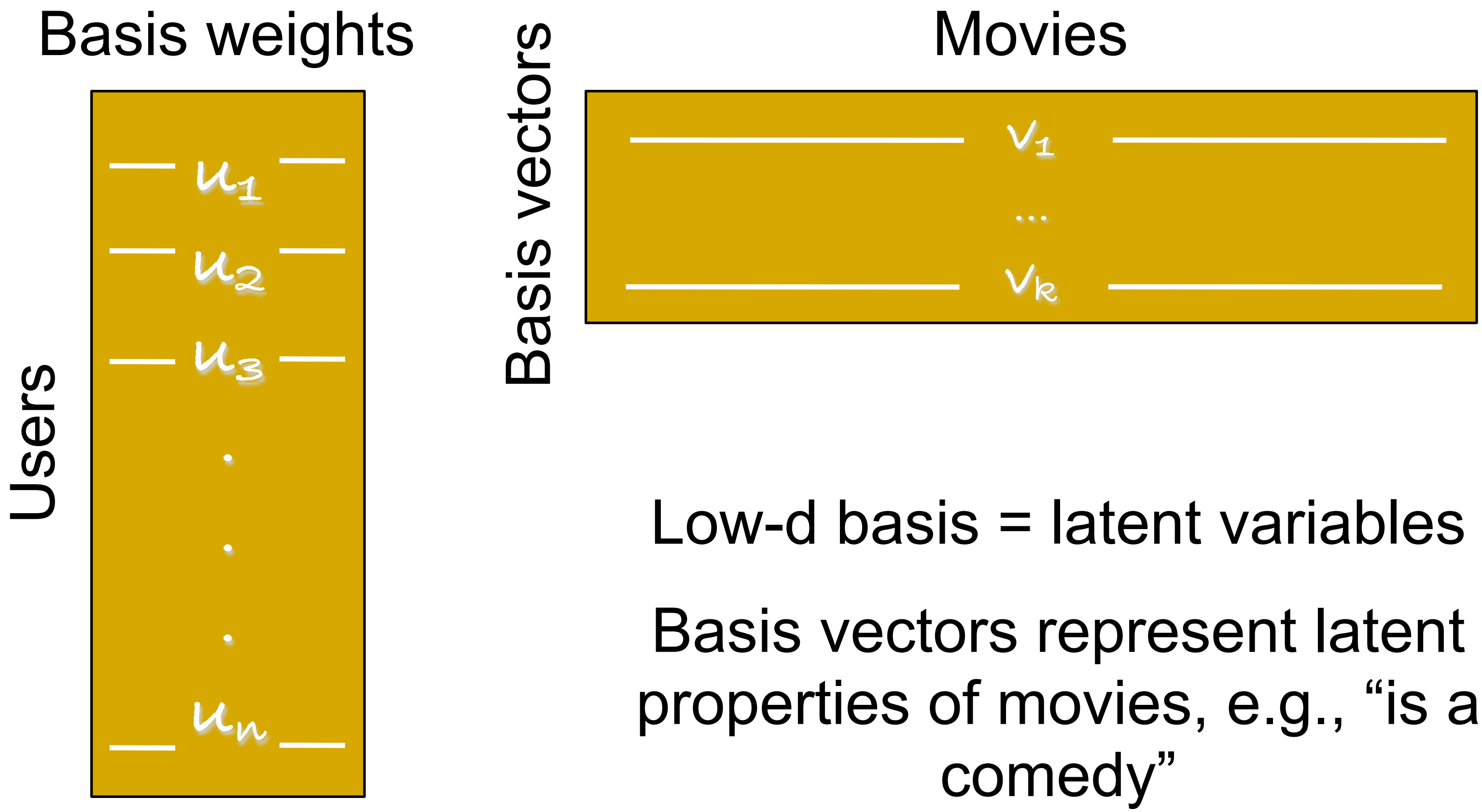
PCA is a widely used and successful model



PCA is a widely used and successful model



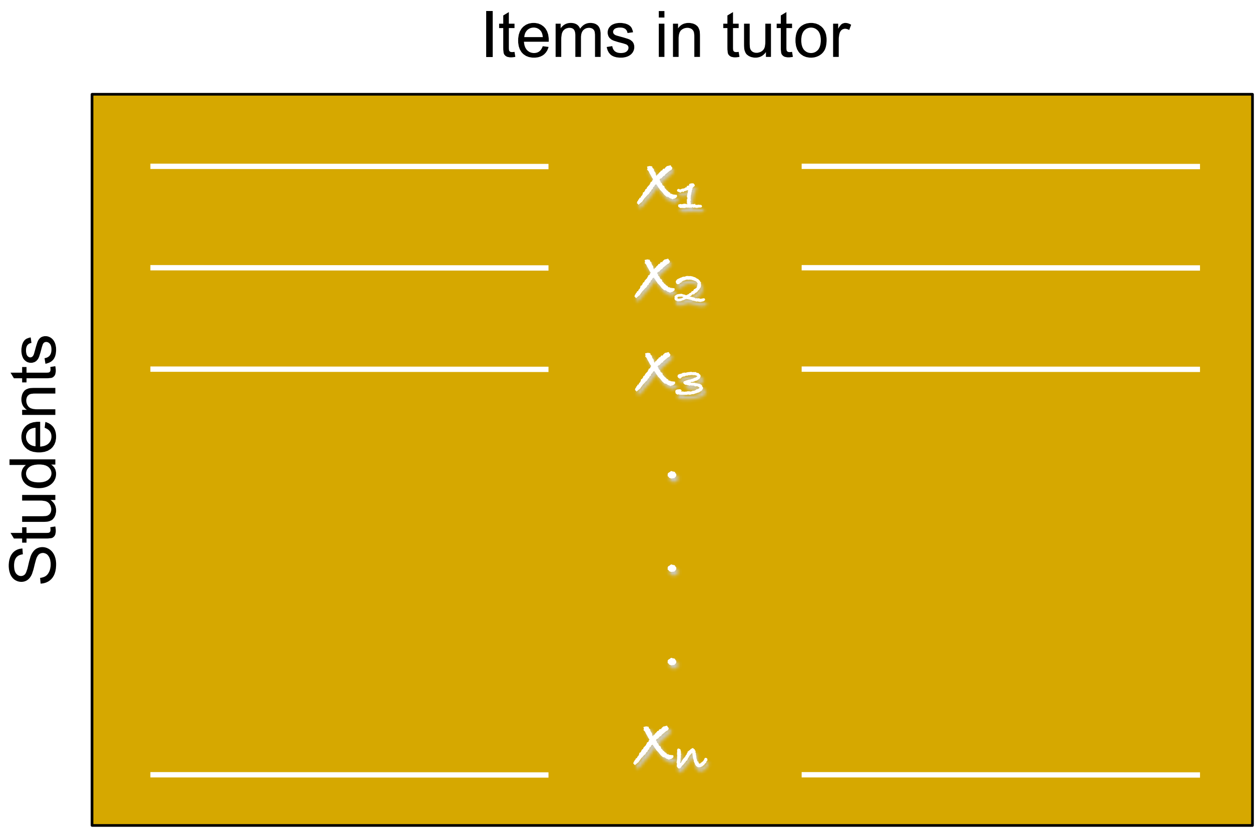
Result of factoring



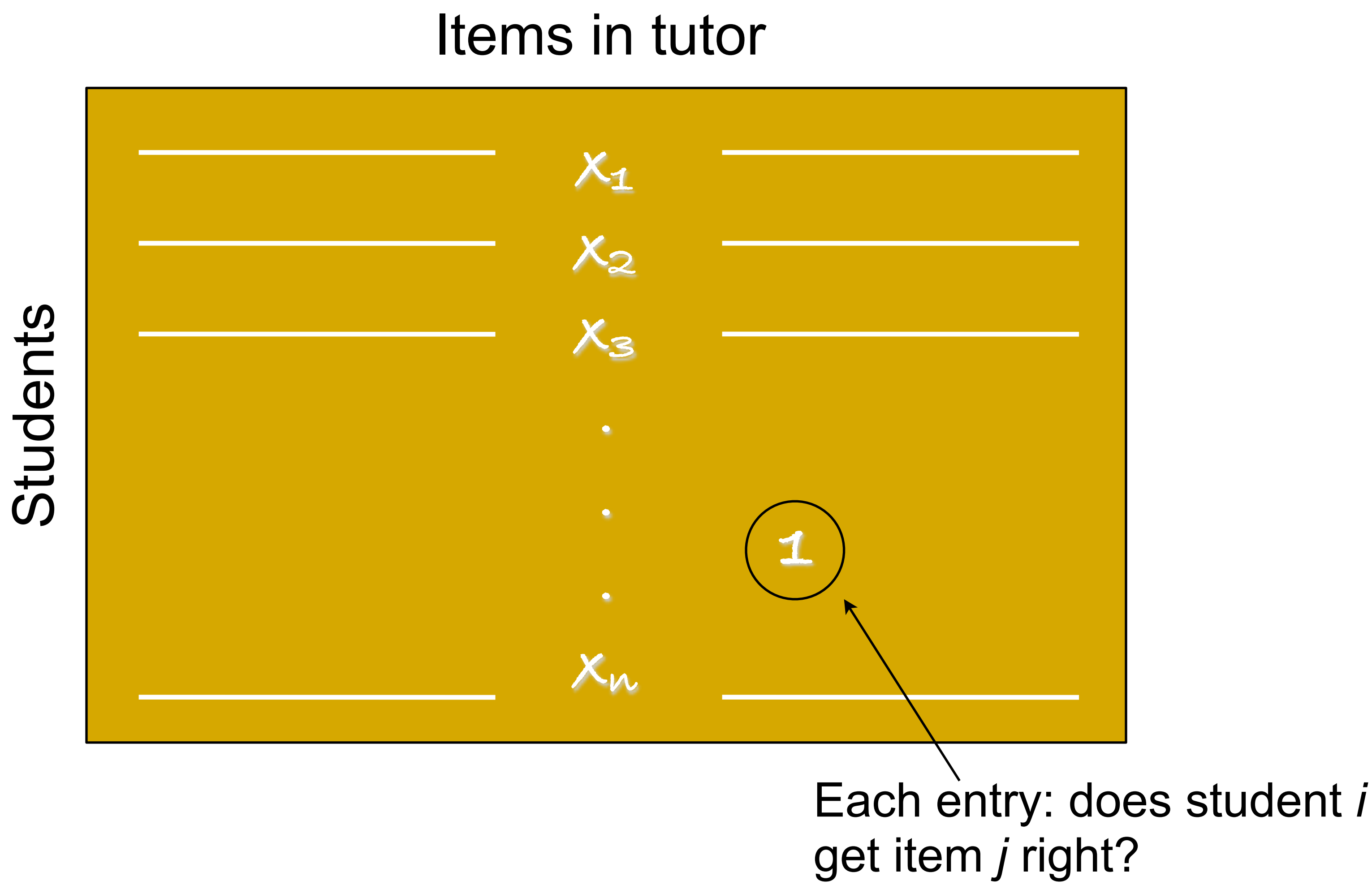
Low-d basis = latent variables

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

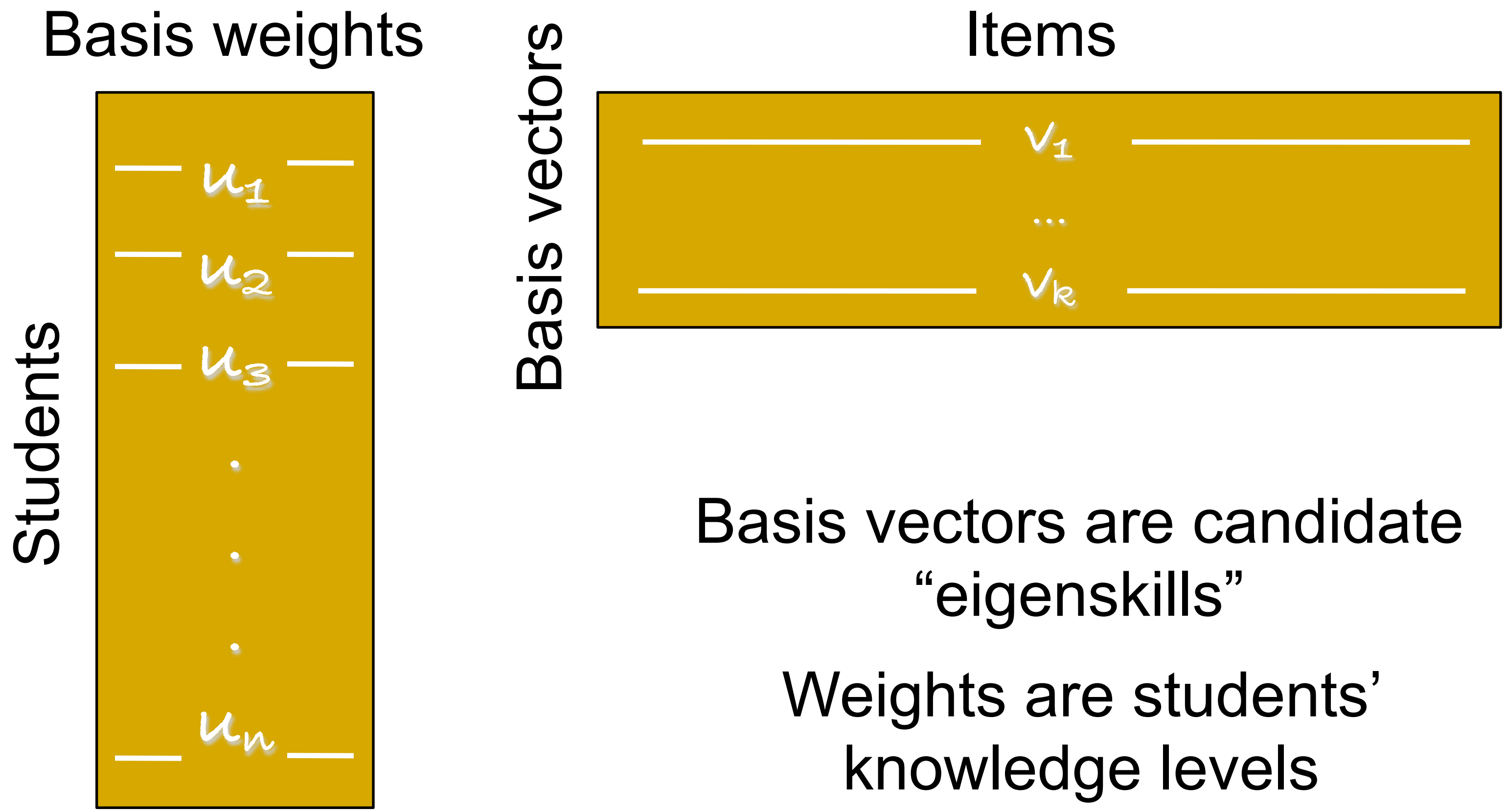
In our case: student-item data



In our case: student-item data



Interpretation of factors

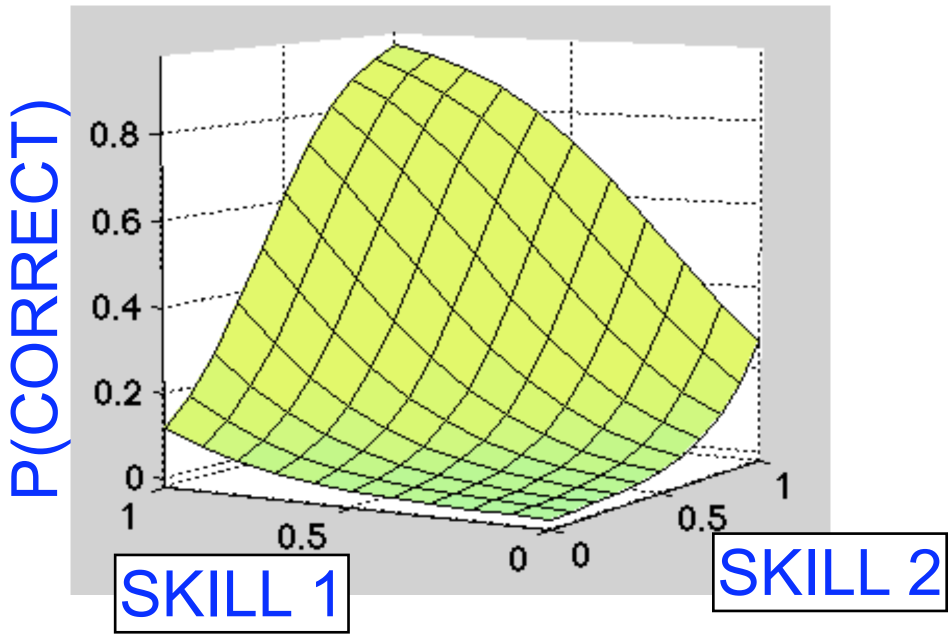


Basis vectors are candidate "eigenskills"

Weights are students' knowledge levels

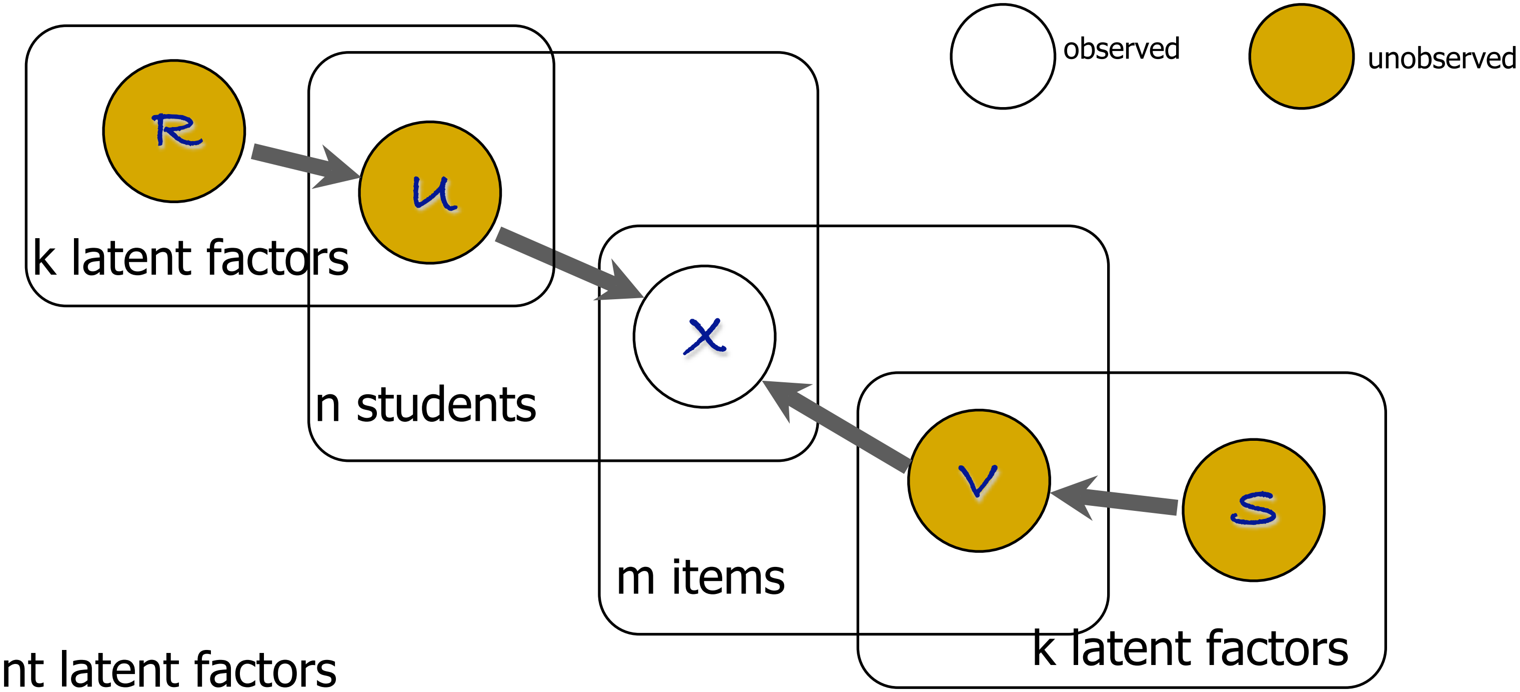
PCA: the good, the bad, and the ugly

- Good: popular, successful, unsupervised
- Bad: linear, Gaussian
- Ugly: maximum likelihood causes overfitting, even w/ lots of data



Nonlinearity: conjunctive skills

Hierarchical Bayesian exponential-family PCA



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

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

logistic PCA

student factor

item factor

Comparison to AFM

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

p = probability correct

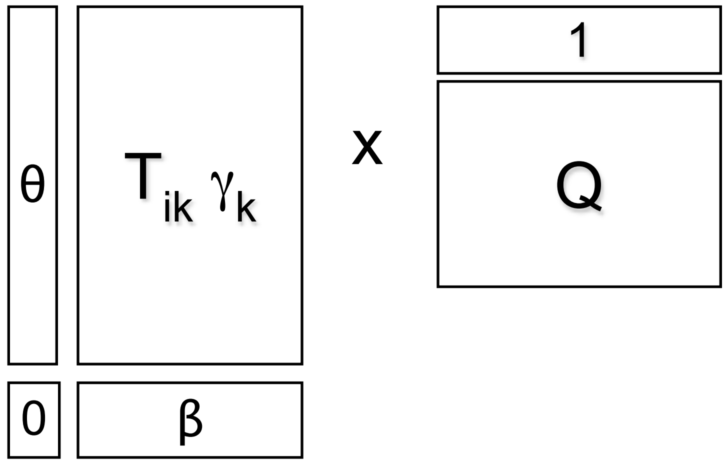
θ = student overall performance

β = skill easiness / difficulty

Q = item × skill matrix

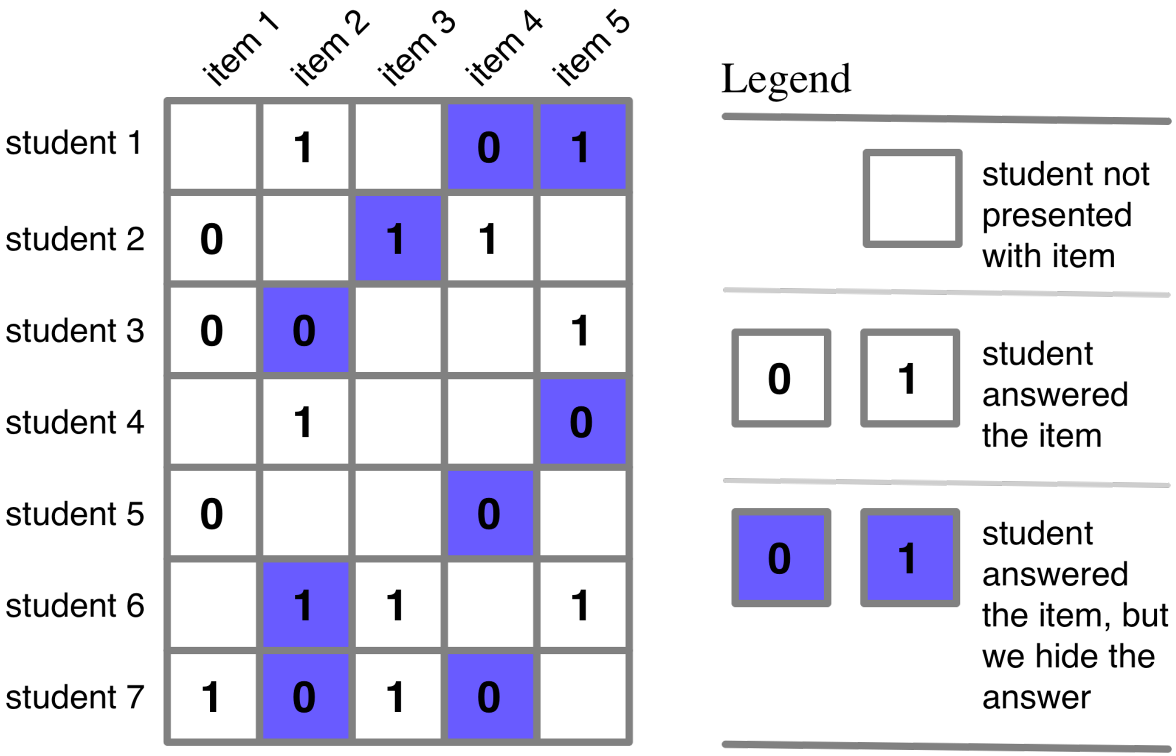
γ = skill practice slope

T = number of practice opportunities



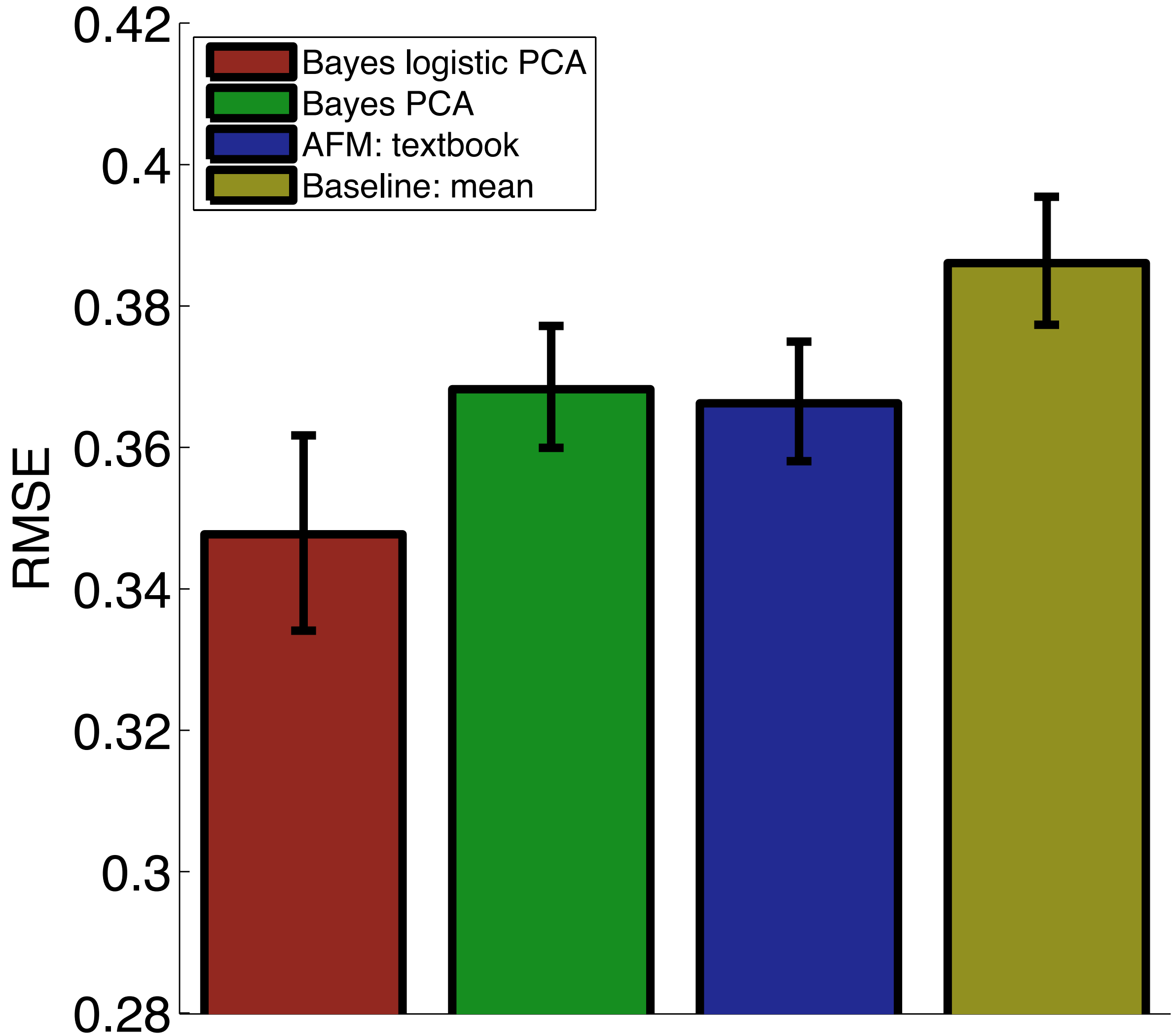
Bayesian logistic PCA: $$\log\left(\frac{p_{ij}}{1-p_{ij}}\right) = \sum_k U_{ik} V_{jk}$$

Geometry Area 1996-1997 data



- 139 items presented to 59 students
- Learn model on 2/3 of responses, test on 1/3

Results: hold-out error



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