Adaptive Information Systems by Dr. Peter Brusilovsky

Probablistic Student Modeling

Yun Huang 03/30/2015

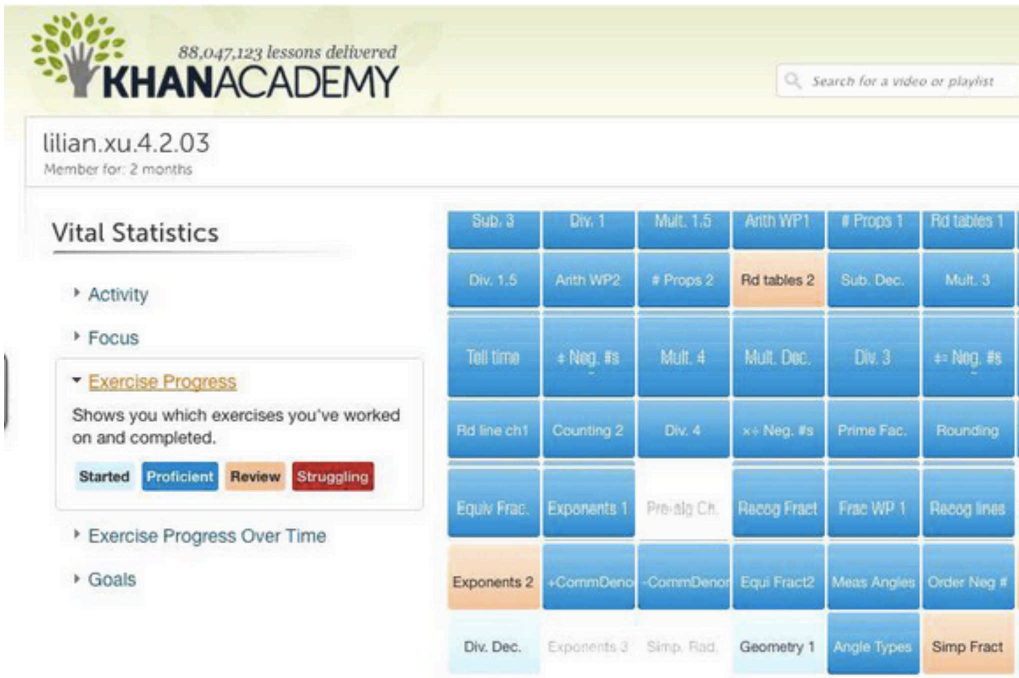
Outline

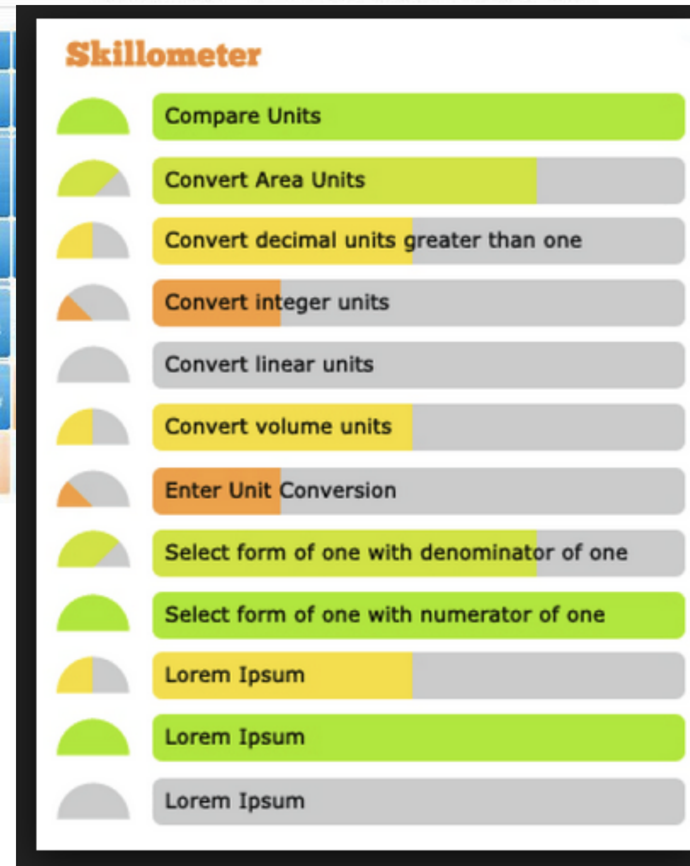
- Introduction
  - What is a student model?
  - Why does a student model matter?
  - Why "probablistic"?
- Student models
  - From IRT to Performance Factor Analysis
  - From Knowledge Tracing to FAST
- Skill models
  - Automatic skill model refinement
- Issues and directions
- Visualization: Open student modeling

Outline

- Introduction
  - What is a student model?
  - Why does a student model matter?
  - Why “probablistic”?
- Student models
  - From IRT to Performance Factor Analysis
  - From Knowledge Tracing to FAST
- Skill models
  - Automatic skill model refinement
- Issues and directions
- Visualization: Open student modeling

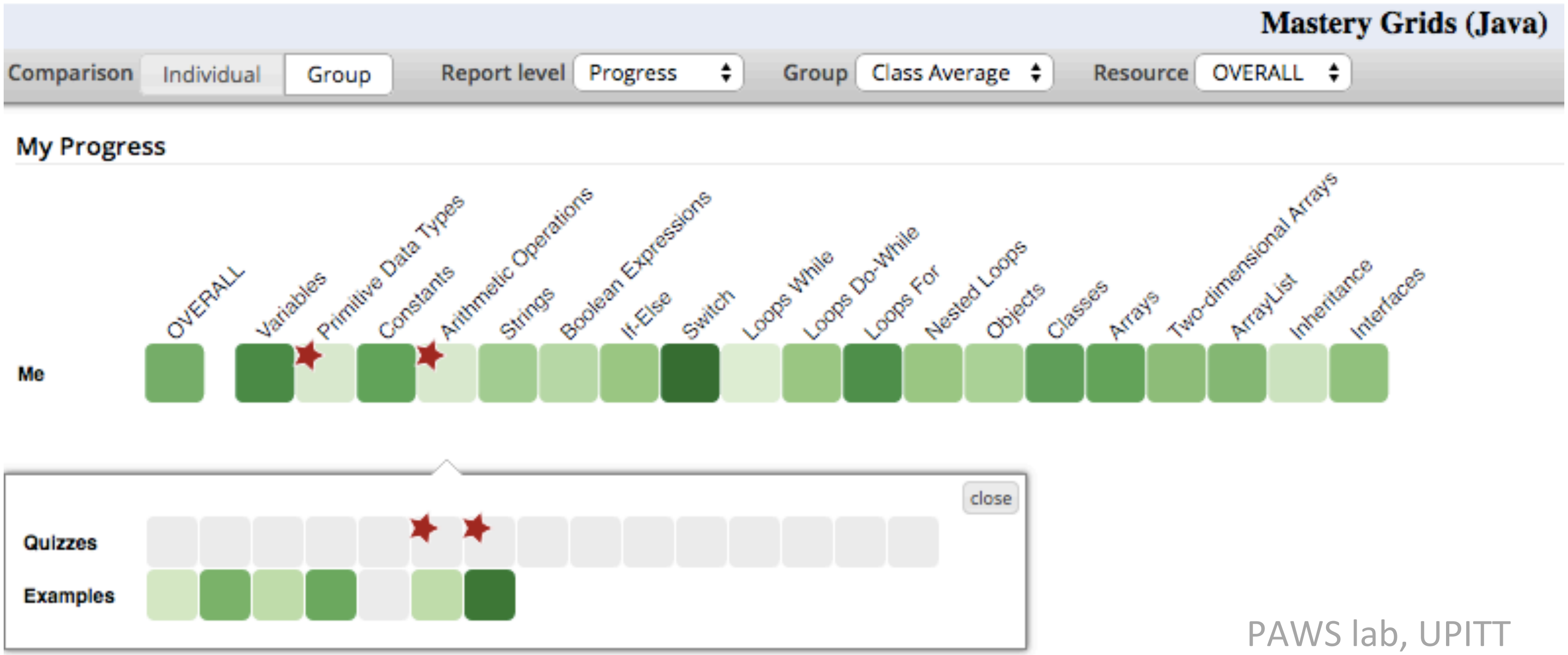
What is a student model?





Basically, a model that tells a student's **progress or knowledge** over a set of **skills**.

Why does a student model matter?



PAWS lab, UPITT

- Identify which skill a student already knows, and move on.
- Identify which skill a student has problem with, and…
    - Recommend proper questions or examples
    - Improve contents

Why “probablistic”?

- Deals with uncertainty: nothing is for sure
  - Probabilistic methods are mainly used for proving the existence of certain (mathematical) objects without explicitly constructing them.
- Assumes distributions for uncertain parameters
  - Probabilistic methods treat the uncertain parameters as random variables, and estimates the uncertain parameters through assumed probability density function.

Outline

- Introduction
  – What is a student model?
  – Why does a student model matter?
  – Why “probablistic”?
- Student models
  – From IRT to Performance Factor Analysis
  – From Knowledge Tracing to FAST
- Skill models
  – Automatic skill model refinement
- Issues and directions
- Visualization: Open student modeling

Item Response Theory Model

(Rasch et al. '60)

- Scenario: Assuming you are having a test with a bunch of questions (items), what do you think are the factors that decide whether you get an answer correct or not?
- Idea:

The probability of getting a question correct depends on student ability and item difficulty

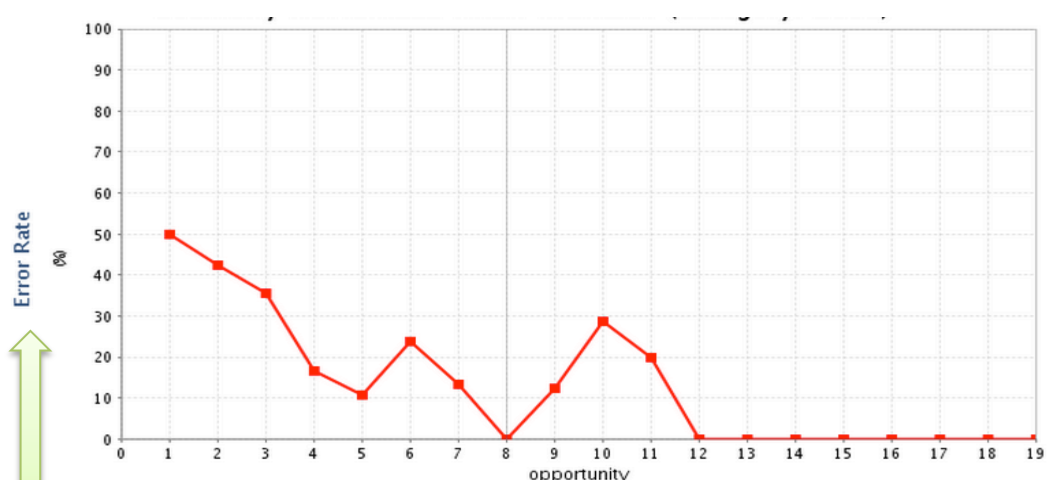
$$\underbrace{P = \frac{1}{1+e^{-(\theta-\beta)}}}_{\text{Logisic regression model}}$$

This is just the simpliest one : Rasch Model

What if you are learning not testing?

power law and learning curve (Newell et al. ‘81)

- IRT estimates one fixed student ability under test environment. What if you are practicing/learning? Can we know your “dynamic ability”?
- Someone found the power law of learning:
  - Practice helps learning!
  - The chance of making an error decreases by a power function.
- This can be visually presented by learning curves.



$$y = aX^{-b}$$

initial difficulty of a skill

learning rate

Y = #incorrect / #total

X = #opportunities a student practices a skill

What if you are learning multiple skills?

Additive Factor Model (Cen et al. '06)

- Classic IRT models model individual skills.
- Additive Factor Model models multiple skills
  - Additive: assuming skills "additively" affect performance
  - Factor: skill
- AFM requires Q-matrix as input:
  - Q-matrix: specifying what skills are required for a question

Table 14 A Q-matrix

Item | Skill	Add	Sub	Mul	Div
2*8	0	0	1	0
2*8 – 3	0	1	1	0
2*8 - 30	0	1	1	0
2*8 +30	1	0	1	0

Additive Factor Model

IRT: $$\ln\left(\frac{p_{ij}}{1-p_{ij}}\right) = \theta_i + \beta_j \Longleftarrow P = \frac{1}{1+e^{-(\theta-\beta)}}$$

AFM: $$\ln\left(\frac{p_{ij}}{1-p_{ij}}\right) = \theta_i + \sum_k q_{jk}(\beta_k + \gamma_k T_{ik})$$

Prob. of getting question i correct of student j

student i ability

Binary indicator for question j skill k (by Q-matrix)

skill k initial difficulty

skill k learng rate

#practices student i had on skill k

Do you learn the same amount from failures vs. successes?

Performance Factor Analysis (Pavlik et al. '09)

AFM:
$$\ln\left(\frac{p_{ij}}{1-p_{ij}}\right) = \cancel{\alpha_i} + \sum_k q_{jk}(\beta_k + \gamma_k T_{ik})$$

prior **successes** of student i in skill k

Prior **failures** of student i in skill k

PFA:
$$\ln\left(\frac{p_{ij}}{1-p_{ij}}\right) = \sum_k q_{jk}(\beta_k + \lambda_k S_{ik} + \rho_k F_{ik})$$

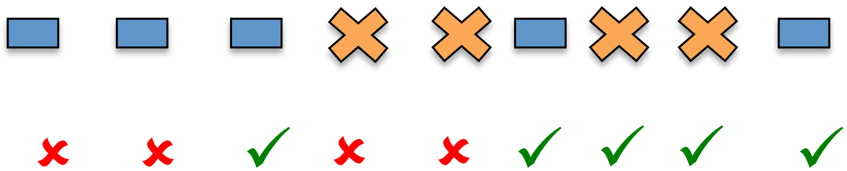
Question about the name of the model:
Where is the "performance"?
Where is the "factor"?

learning rate from correct practices

learning rate from incorrect practices

How to model latent knowledge level?

Knowledge Tracing (Corbett et al. ‘95)

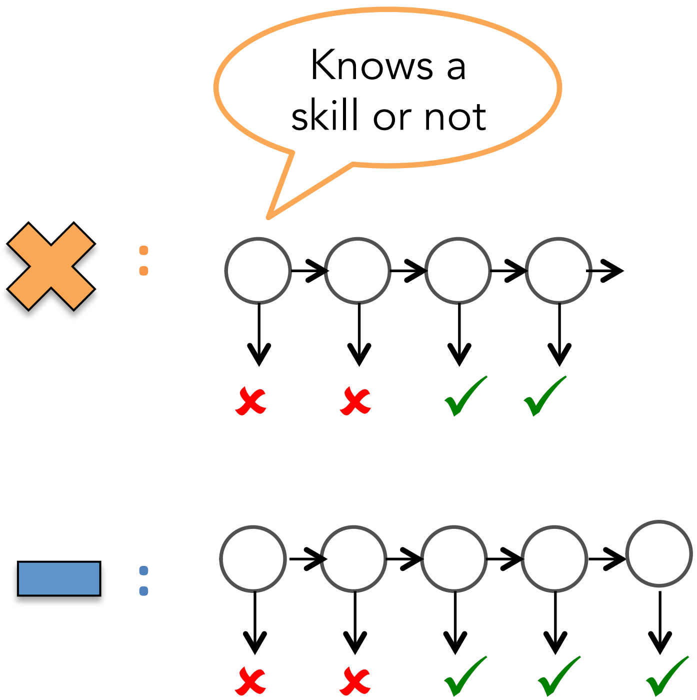


- Previous models can’t tell directly the dynamic **knowledge** level
- Is knowledge level observable?
- Can we model it as a latent variable?

:

:

Knowledge Tracing



- HMM models:
  - Binary latent variables (K) indicate the student knowledge
  - Binary observed variables (Y) indicate the student performance
- Four parameters:
  1. Initial Knowledge
  2. Learn
  3. Guess
  4. Slip

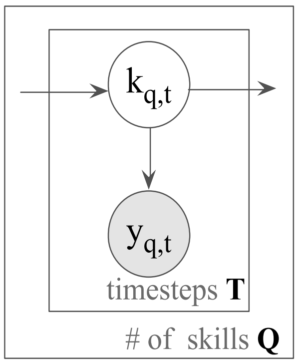
Transition

Emission

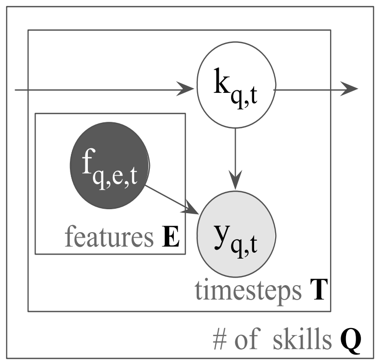
Init	$P(K_0)$=learned	**Learn**	$P(K_{t+1}$=learned $\mid K_t$=unlearned)
Guess	$P(Y_t$=correct $\mid K_t$=unlearned)	**Slip**	$P(Y_t$=incorrect $\mid K_t$=learned)

Knowledge Tracing Family

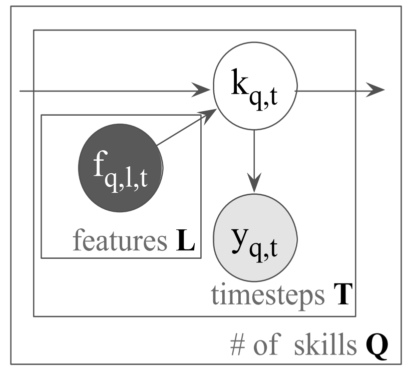
Original formulation [Corbett et al '95]



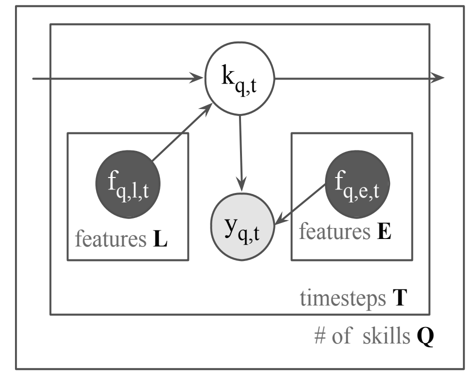
Emission (guess/slip) Features



Transition (init/learn) Features



Both



Feature	Emission	Transition	Both
Student ability		[Pardos et al '10]	[Lee et al '12] [Yudelson et al '13]
Item difficulty	[Gowda et al '11] [Pardos et al '11]		[Schultz et al '13]
Subskills		[Xu et al '12]	
Help		[Sao Pedro et al '13]	[Beck et al '08]

Feature-Aware Student knowledge Tracing
(Gonzales, Huang and Brusilovsky '14)

- Unsupervised learning (HMM) with features
- Incorporated features into Knowledge Tracing
  - *Features can be student ability, item difficulty, whether a student ask for help or not, etc.*

	features	slip/ guess	recency/ ordering	learning
FAST *	✓	✓	✓	✓
PFA Pavlik et al '09	✓	✗	✗	✓
Knowledge Tracing Corbett & Anderson '95	✗	✓	✓	✓
Rasch Model Rasch '60	✓	✗	✗	✗

* Code: http://ml-smores.github.io/fast/
Paper: http://educationaldatamining.org/EDM2014/uploads/procs2014/long%20papers/84_EDM-2014-Full.pdf

Outline

- Introduction
  - What is a student model?
  - Why does a student model matter?
  - Why “probablistic”?
- Student models
  - From IRT to Performance Factor Analysis
  - From Knowledge Tracing to FAST
- Skill models
  - Automatic skill model refinement
- Issues and directions
- Visualization: Open student modeling

Skill Model and how we construct it

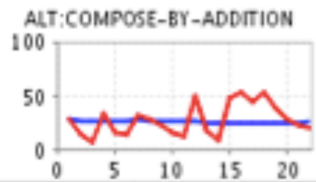
- Student model needs a good skill model
  - Question: “3*(1-4)=?”. A student answers incorrectly because he/she doesn’t know negative numbers.
    - If we label this question with only skill “multiplication”, then we may wrongly infer the student has problem with “multiplication”! We need to also label skill “negative number”!
- Most of the time, we rely on expert engineering…
- Can we use data-driven probablistic methods?
  - Yes, here is a succesful method (Koedinger et al. ‘12): **Automated Student Model Improvement**

    data repository + crowd sourcing + statistical models

Let’s look at the “data repository + crowd sourcing” part

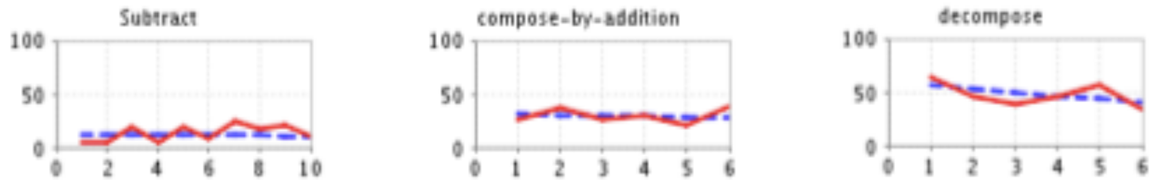
PSLC Datashop: https://pslcdatashop.web.cmu.edu/

KC Values For Original Model



KC Name	Intercept (logit)	Intercept (probability)	Slope
ALT:COMPOSE-BY-ADDITION	1.04	0.74	0

KC Values For DecompArithDiam Model



KC Name	Intercept (logit)	Intercept (probability)	Slope
Subtract	2.05	0.89	0
compose-by-addition	0.84	0.7	0
decompose	-0.56	0.36	0.148

Figure 3. A knowledge component (KC) with a non-smooth learning curve (see top half of the figure) is replaced in an improved student model with three new KCs with smoother curves (see bottom half of the figure).

Red learng curve is fluctuating, non-smooth.

human inspection

Change to three skills, each of which has a smoother learning curve

Use new skill models as the starting model for further refinement

Let's look at the "statistical models" part:

Learning Factor Analysis (Cen et al. '06)

Table 14 A Q-matrix

Item | Skill	Add	Sub	Mul	Div
2*8	0	0	1	0
2*8 – 3	0	1	1	0
2*8 - 30	0	1	1	0
2*8 +30	1	0	1	0

Table 15 A P-matrix

Item | Skill	Dealing with negative numbers	Two digit arithmetic	…
2*8	0	0	
2*8 - 3	0	1	
2*8 - 30	1	1	
2*8 +30	0	1	

Split

Table 18 Splitting "Sub" in Q by "neg" in P

Item | Skill	Add	Sub^Neg	Mul	Div	Sub &neg
2*8	0	0	1	0	0
2*8 – 3	0	1	1	0	0
2*8 - 30	0	0	1	0	1
2*8 +30	1	0	1	0	0

Add

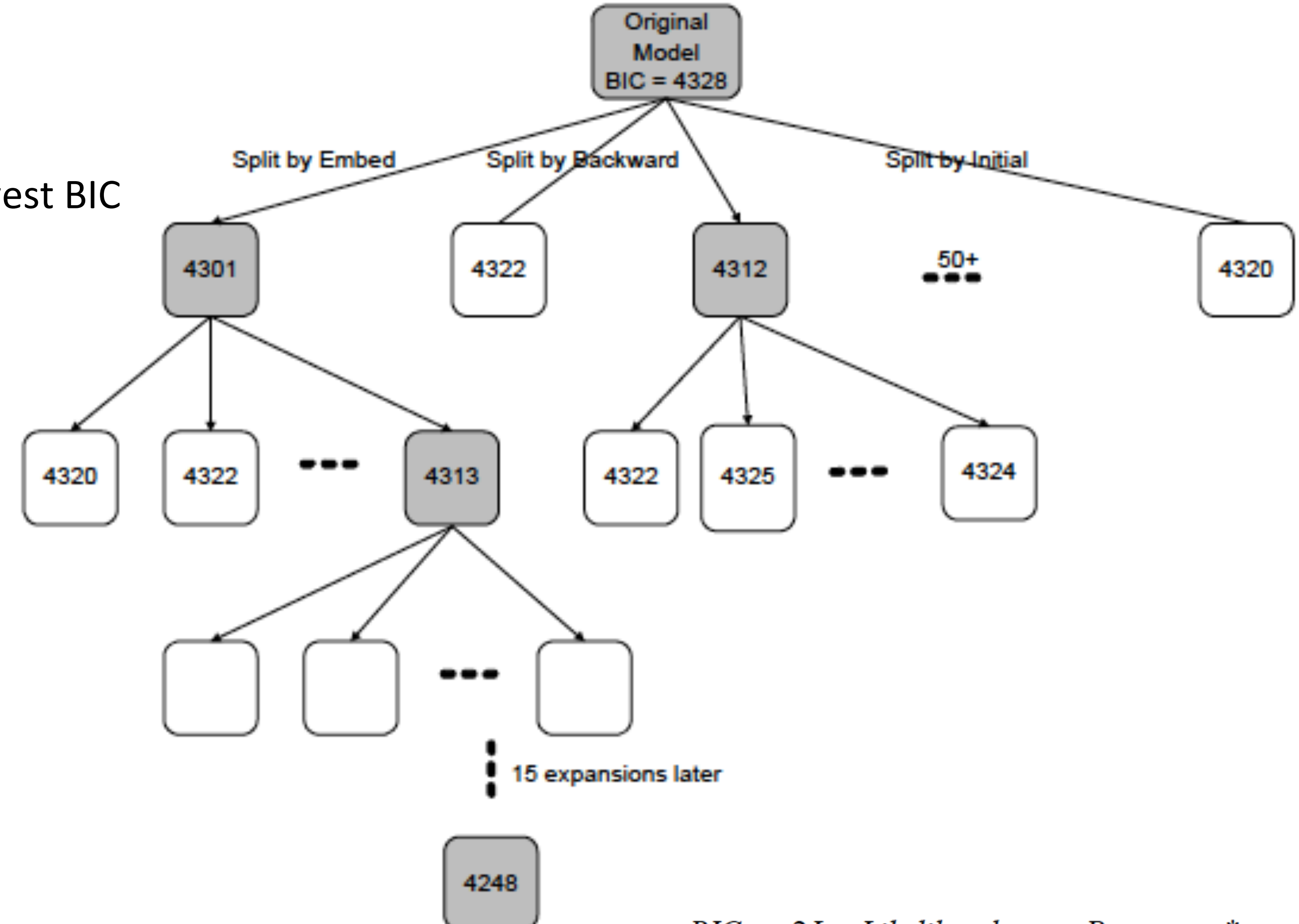
Item | Skill	Add	Sub	Mul	Div	neg
2*8	0	0	1	0	0
2*8 – 3	0	1	1	0	0
2*8 - 30	0	1	1	0	1
2*8 +30	1	0	1	0	0

Merge

Item | Skill	Add-Sub	Mul	Div
2*8 + 3	1	1	0
2*8 – 3	1	1	0
2*8 - 30	1	1	0
2*8 +30	1	1	0

Learning Factor Analysis Best-first Search

Lowest BIC



$$BIC = -2LogLikelihood + numParemeter * numObservation$$

Figure 7 A best-first search through the cognitive model space

Outline

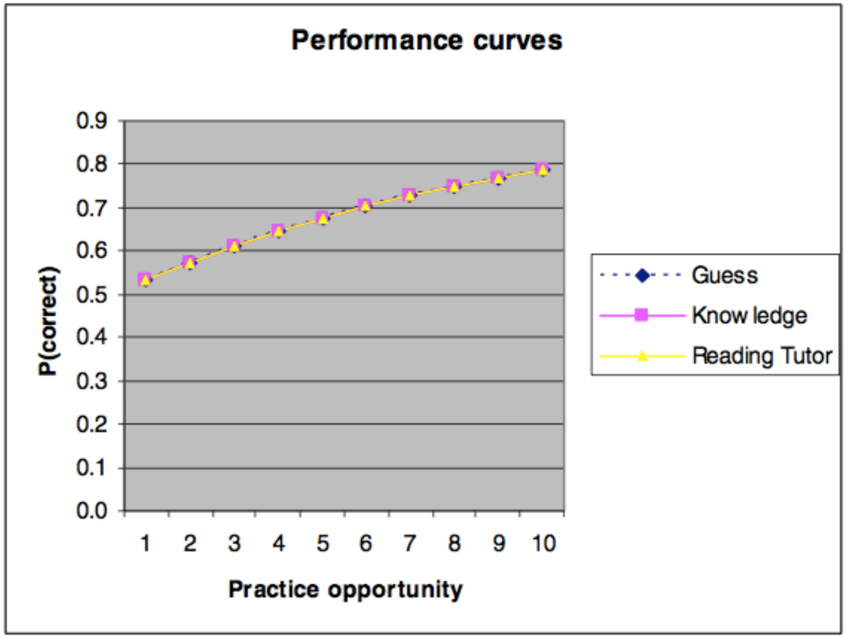
- Introduction
  - What is a student model?
  - Why does a student model matter?
  - Why “probablistic”?
- Student models
  - From IRT to Performance Factor Analysis
  - From Knowledge Tracing to FAST
- Skill models
  - Automatic skill model refinement
- Issues and directions
- Visualization: Open student modeling

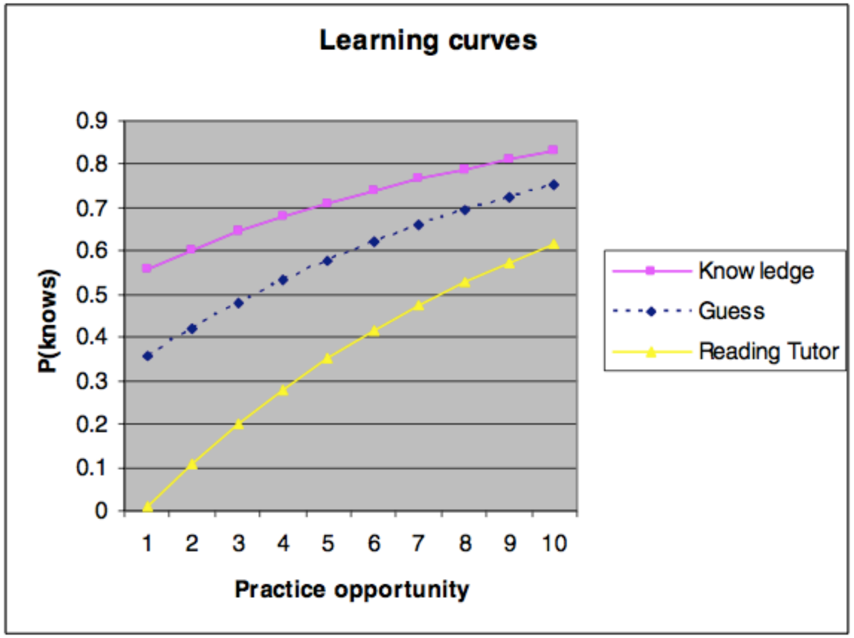
Issues and directions

- Quality of fitted parameters?
  - Predictive Performance : how accurate does the student model predict?
  - Plausiblity : how plausible are the parameters?
  - Consistency: If we train the model several times, does the model give the same (similar) parameter estimation?
- Dynamic multiple skill modeling and Dynamic cognitive diagnostics ?
- Automatic skill model discovery?
- Modeling knowledge from text?

Plausibility and Consistency

- Plausibility: A highly predictive model can give ridiculous parameters:
  - A student with low knowledge level is more likely to answer correctly than a student with high knolwedge level
- Consistency: Equivalently predictive models can have different knowledge estimations! Following three KT models fit the performance equally well (left), but show different esitmation of knowledge level (right).

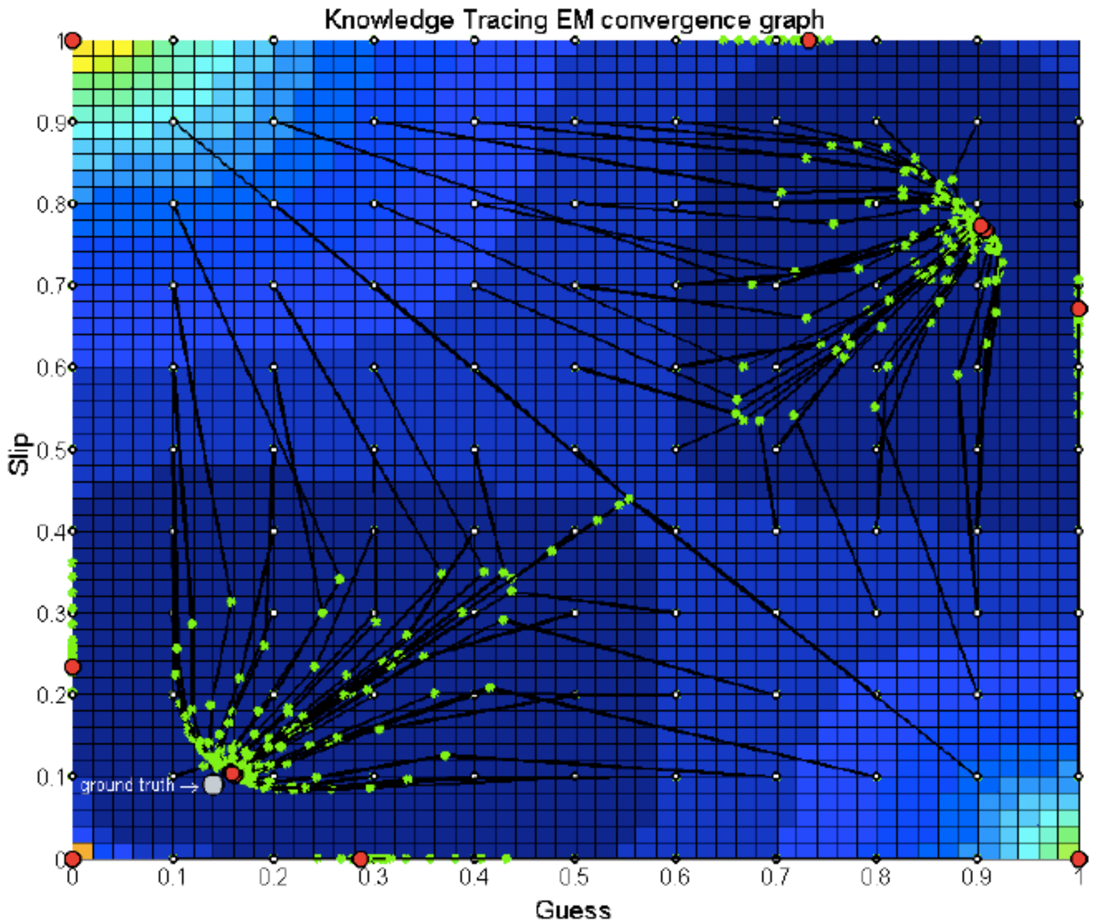




(Beck&Zhang 08')

Where do the problems come from?

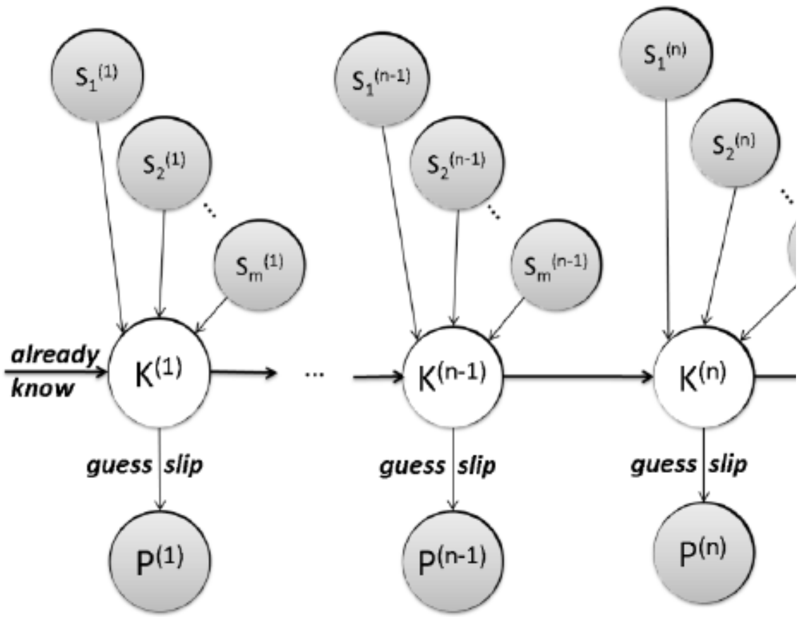
- Training Knowledge Tracing based models using Expectation Maximization can suffer from local optimum and yield multiple global optima *(with limited precision).*
  - Depending on different intial values, we can get different parameter estimations.



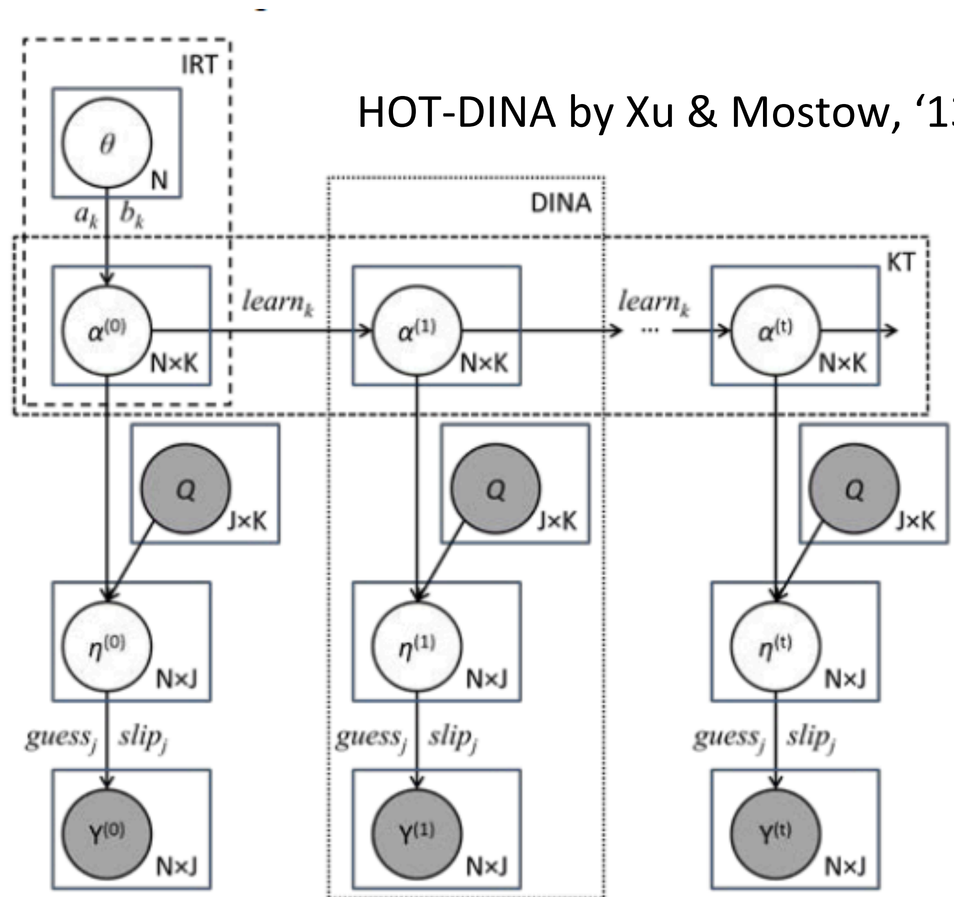
- Also, **student data** has noise, and the **skill model** has noise too, which may result in violations of the assumptions of student models.

Zach&Heffernan '10

Dynamic multiple skill modeling and Dynamic cognitive diagnostics



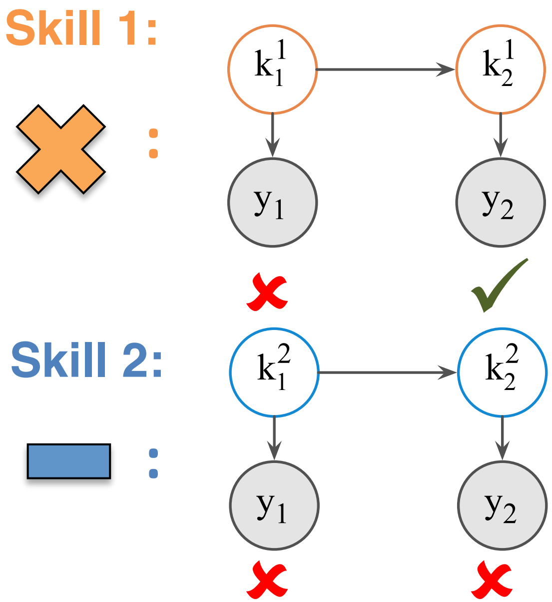
LR-DBN by Xu & Mostow, '12



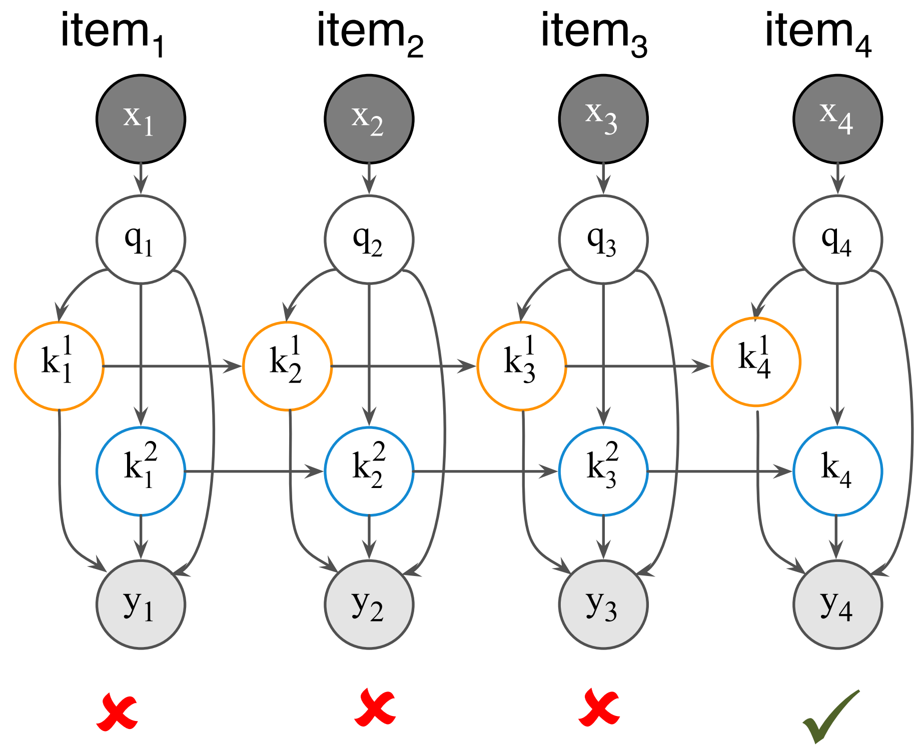
HOT-DINA by Xu & Mostow, '13

Automatic skill model discovery

Knowledge Tracing



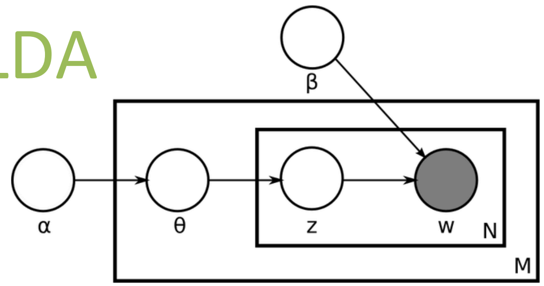
Topical HMM (with 2 skills)



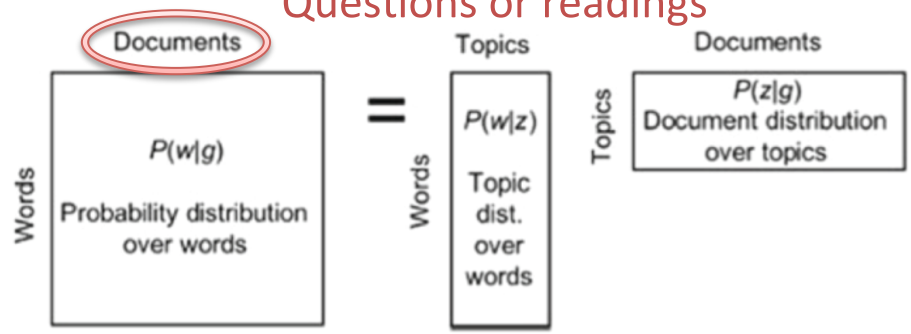
Gonzales & Mostow '12~'14

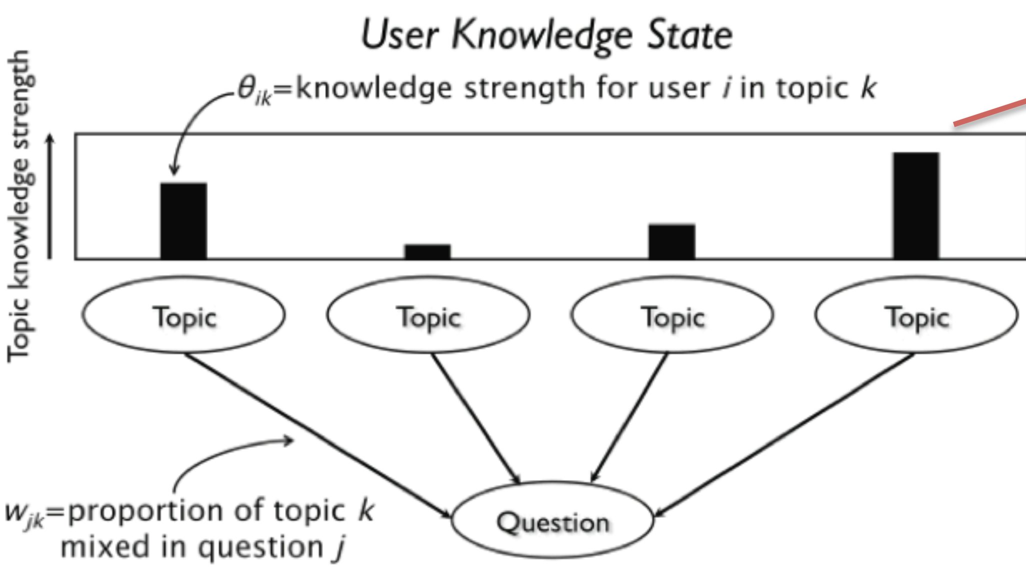
Modeling knowledge from text (Pirolli & Kairam, ’13)

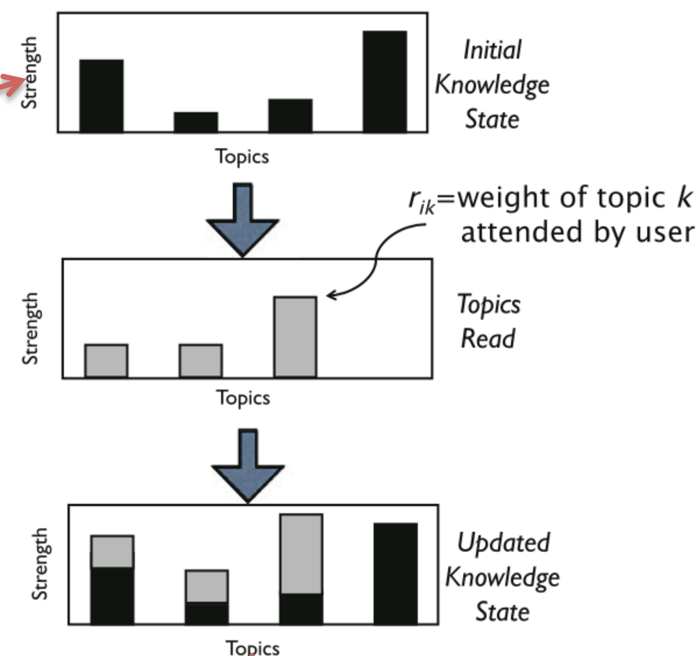
LDA



Questions or readings







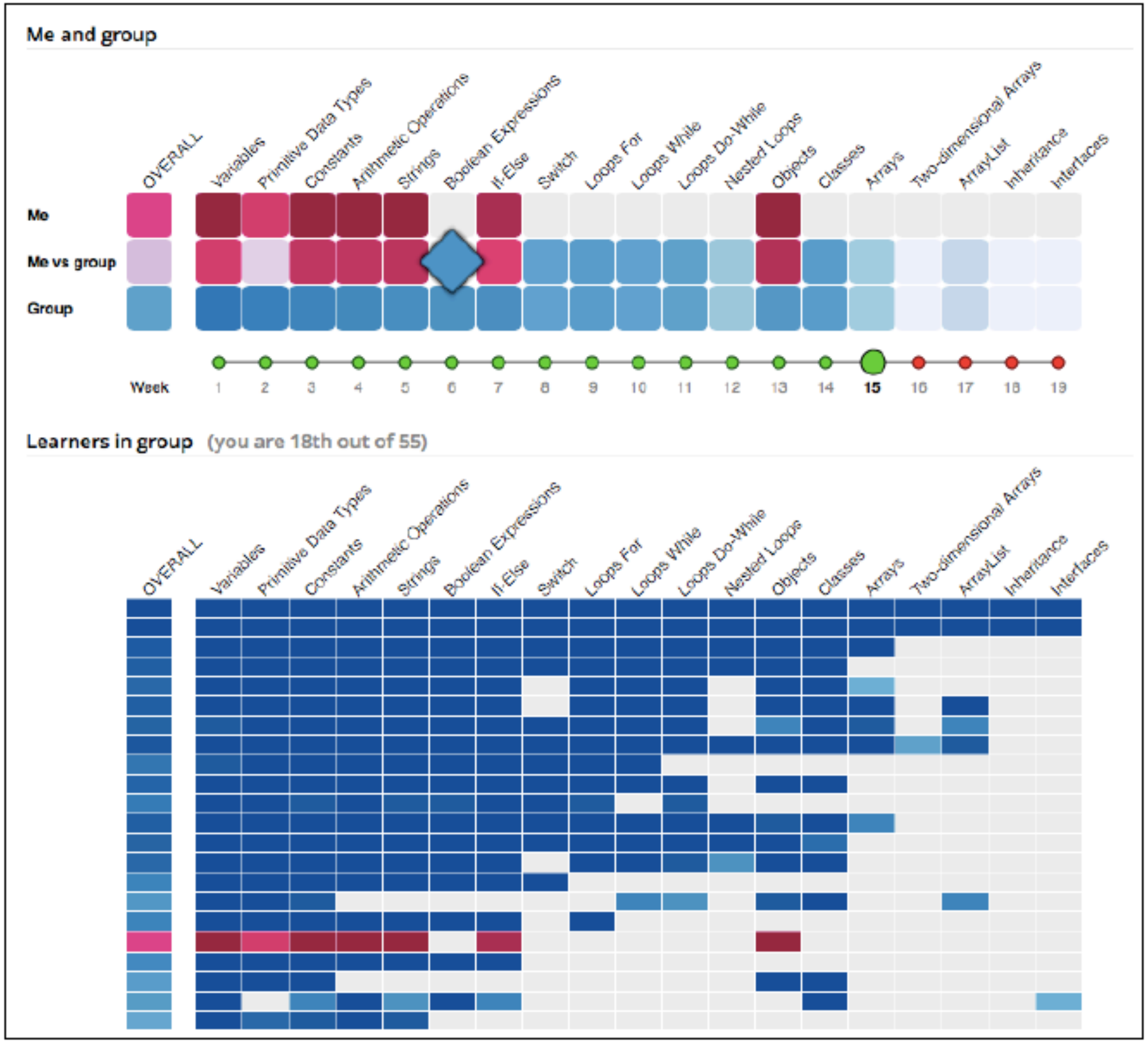
A user knowledge state is represented as a profile over topics. A question tests a mix of topic knowledge

Predict pretest, posttest (etc.)

Outline

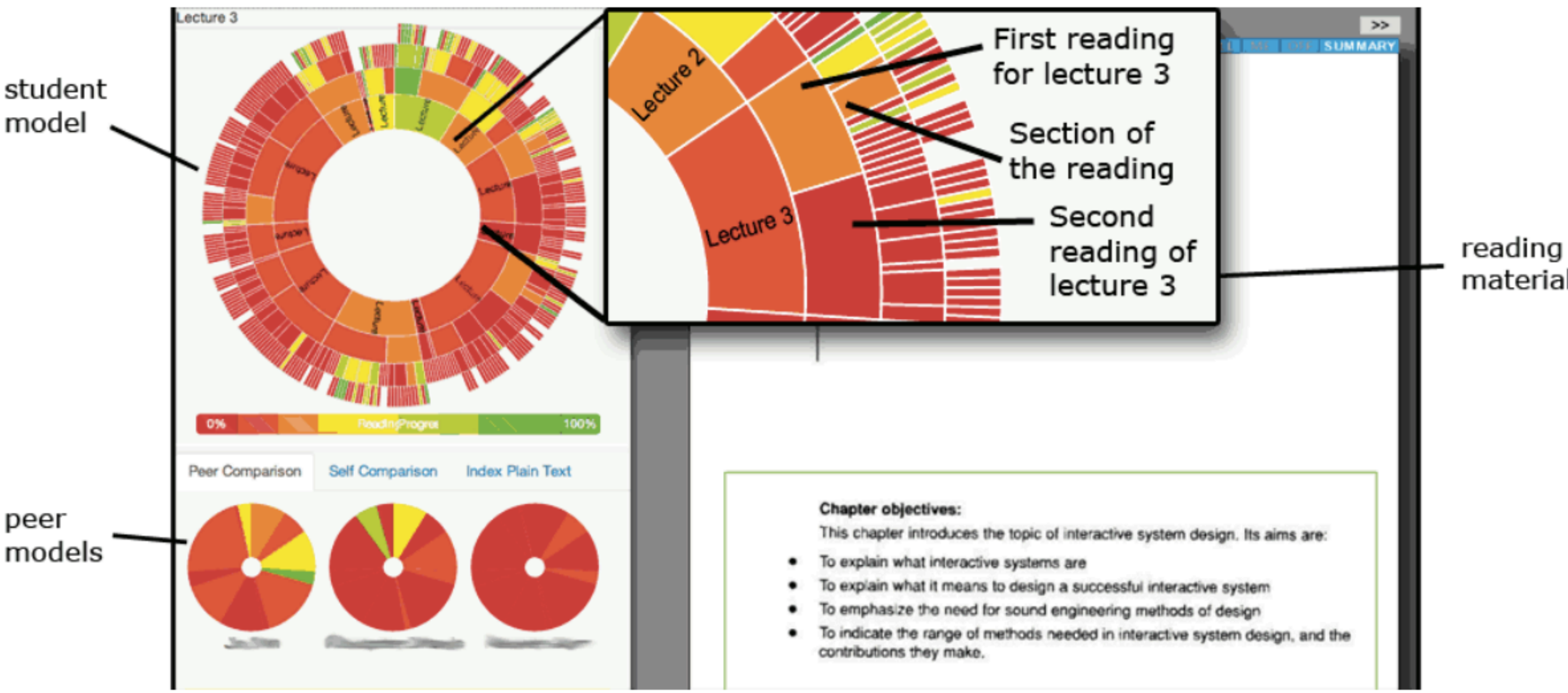
- Introduction
  - What is a student model?
  - Why does a student model matter?
  - Why “probablistic”?
- Student models
  - From IRT to Performance Factor Analysis
  - From Knowledge Tracing to FAST
- Skill models
  - Automatic skill model refinement
- Issues and directions
- Visualization: Open student modeling

Mastery Grid (Brusilovsky et al. ’13~‘14)



Open student model
&
social comparision

Reading Circle (Guerra, Parra & Brusilovsky, '13)



Welcome to join the
student modeling community!

Thank you!

BACKUP

Induce the topic model

- question document's distribution in the topics (w_jk)
- words read by user i's distribution in the topics (r_ik)
- They design following formula to deduct the "learning effect" from reading for performance in a question

(p_ij is relavance of user i's reading to test item j)

$$\rho_{ij} = \begin{cases} 0, \text{ if item } j \text{ is on the pretest} \\ \sum_{k=1}^{T} r_{ik} w_{jk}, \text{ if item } j \text{ is on the posttest} \end{cases}$$

Measurement Framework

- Predict pretest, posttest, and learning gain (post-pre) scores.
- Overall logistic regression formula:

$$\Pr\left(Y_{ij}=1|\tilde{\theta}_i,\tilde{w}_j,\rho_{ij}\right)=\frac{1}{1+e^{-f\left(\tilde{\theta}_i,\tilde{w}_j,\rho_{ij}\right)}}$$

$\tilde{\theta}_i$ User(i) ability distribution over topics(k):

- one for latent knowledge/ability to each topic
- one for learning ability from reading

$\tilde{w}_j$ Topic(k) relevance distribution over item(j):

ρ_{ij} Relevance of reading from topics to item(j) by user(i)

Compare model variations

- Compare different individual differences assumptions
  - Model1: users share same reading learning ability

$$f\left(\tilde{\theta}_i, \tilde{w}_j, \rho_{ij}\right) = \theta_0 \rho_{ij} + \theta_{i1} w_{j1} + \cdots + \theta_{iT} w_{jT} \quad (9)$$

  - Model2: users have different reading learning ability

$$f\left(\tilde{\theta}_i, \tilde{w}_j, \rho_{ij}\right) = \theta_{i0} \rho_{ij} + \theta_{i1} w_{j1} + \cdots + \theta_{iT} w_{jT} \quad (10)$$

- Also compare with other logistic regression models without topic modeling information

(just has user abilities parameters overall)