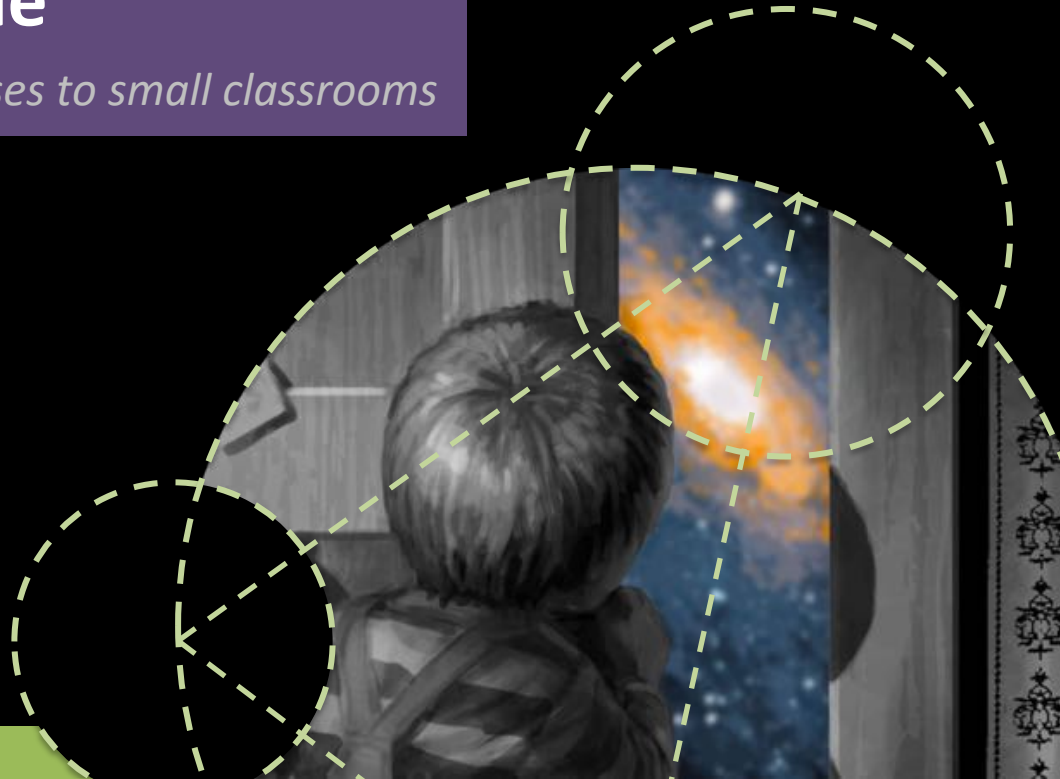
Stanford ENGINEERING
Computer Science

Understanding Students at Scale

From massive online courses to small classrooms

April 2019

Chris Piech



Chapter 0: The Grand Challenge

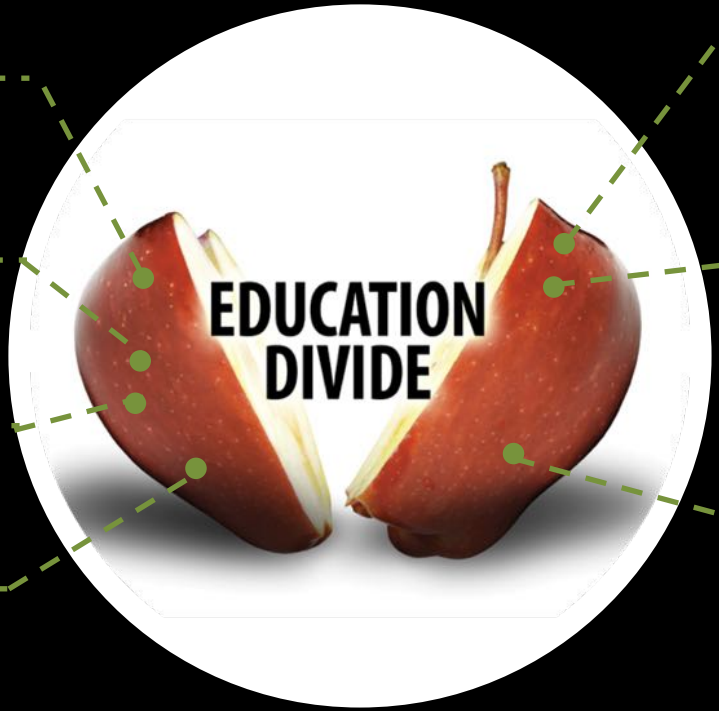
Quality Education Gap

90% of children enroll in primary education [1]

40% in secondary education [1]

20% in tertiary education [1]

Dramatic quality differences



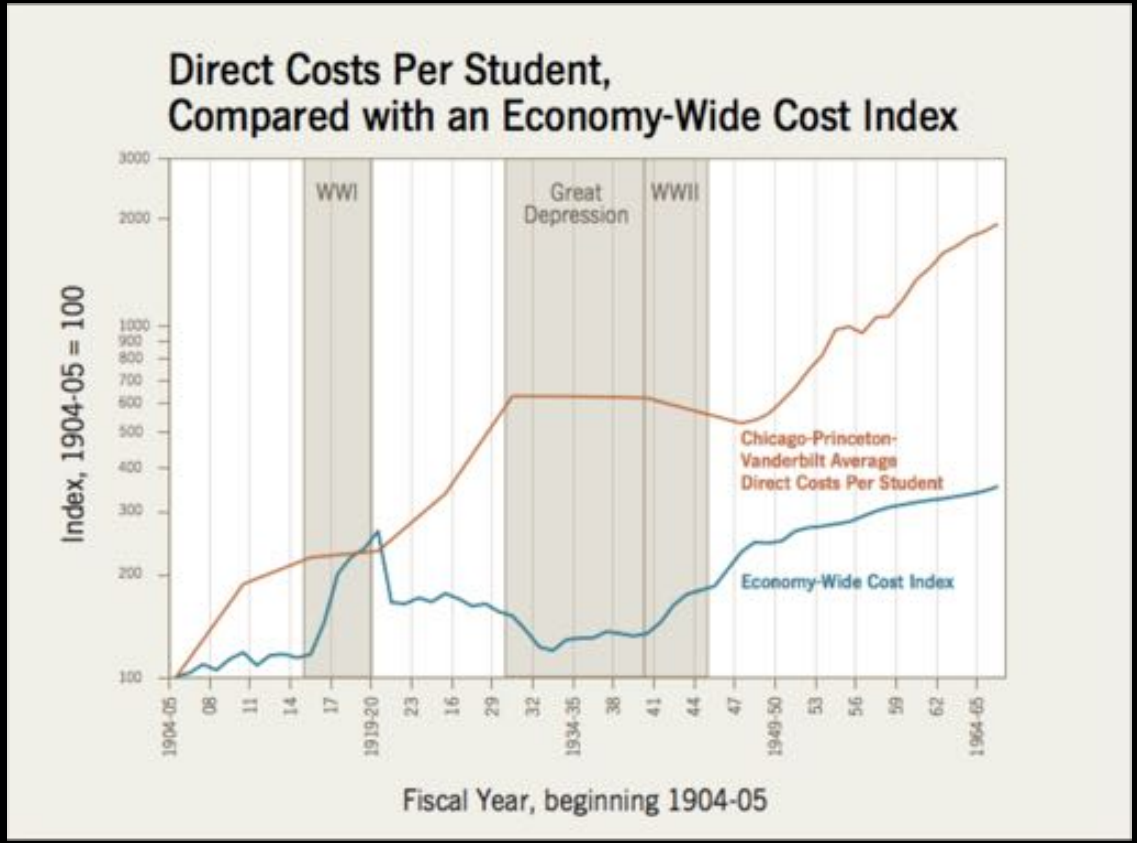
375 million workers need to be retrained by 2030 [3]

Half a million unfilled computer science jobs (60% of STEM jobs) [2]

For all learners we want *quality*.

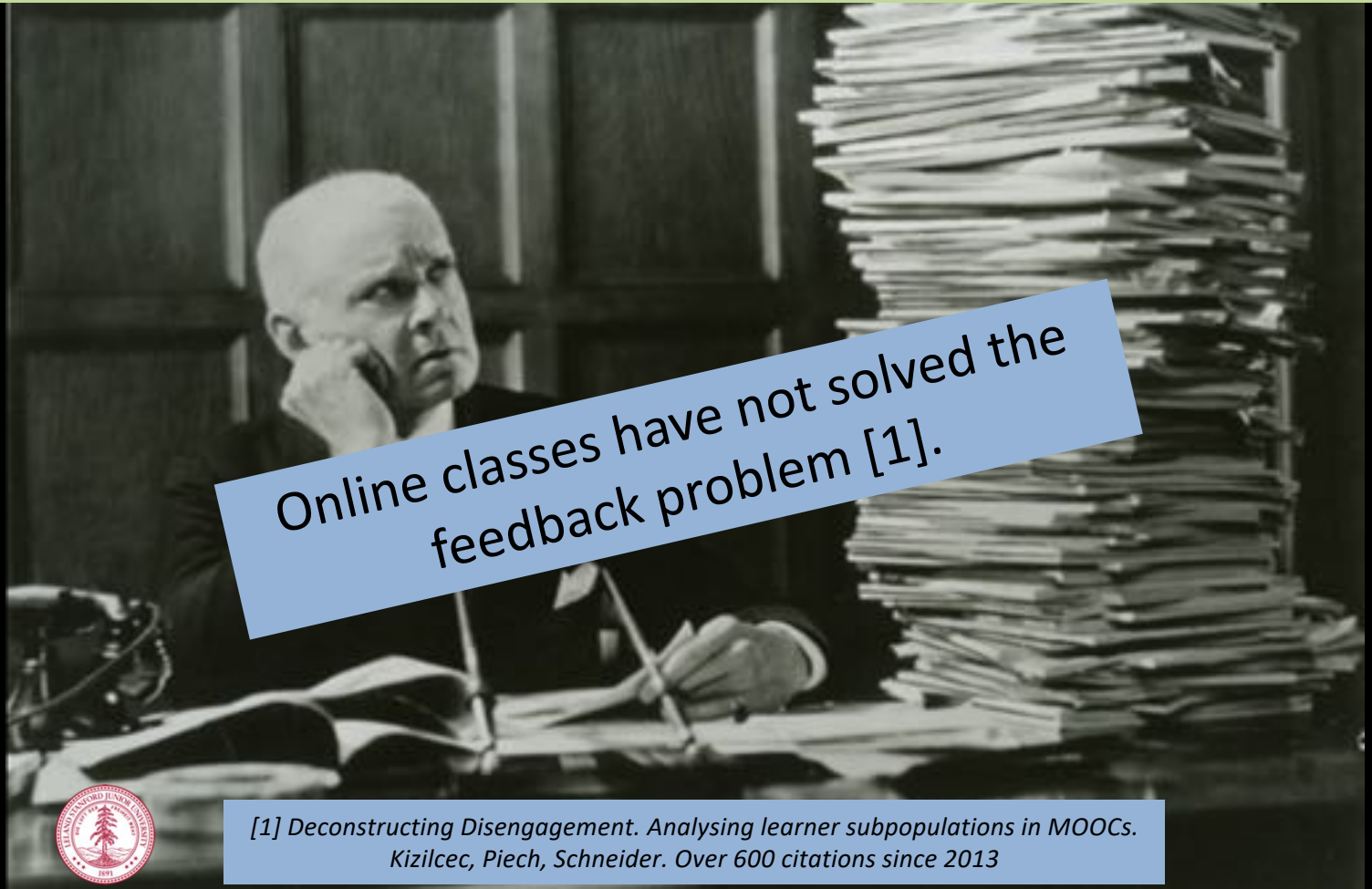
[1] *World development indicators 2015*. World Bank Publications, 2015.
[2] USA Bureau of Labor Statistics Employment Projections , 2016.
[3] Jobs lost, jobs gained. McKinsy Global Institute, 2017.

Cost Disease of Education



William G Bowen. *The cost disease in higher education: Is technology the answer*. 2012.

Feedback is Labor Intensive

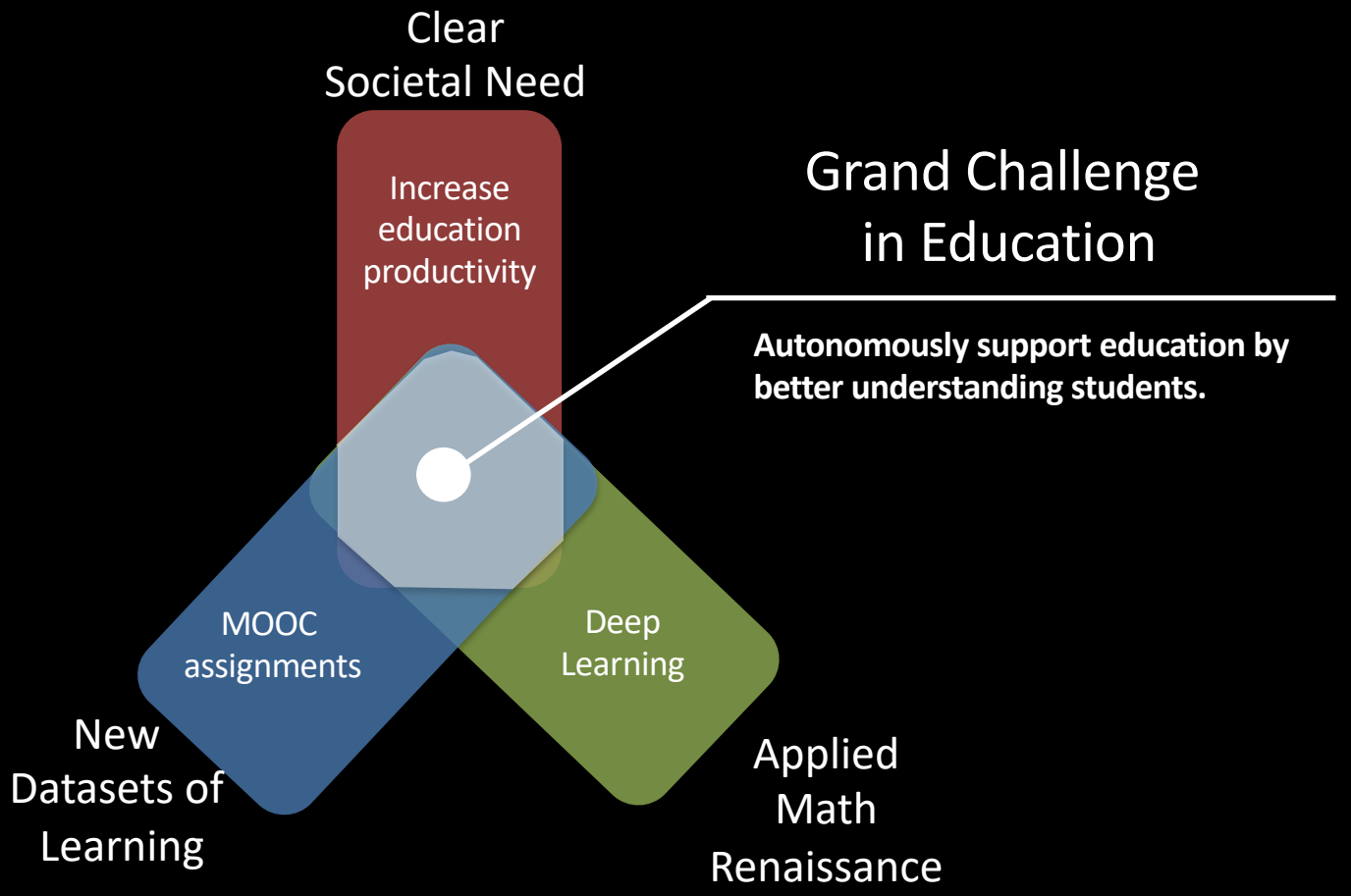


[1] Deconstructing Disengagement. Analysing learner subpopulations in MOOCs. Kizilcec, Piech, Schneider. Over 600 citations since 2013

Progress in other domains

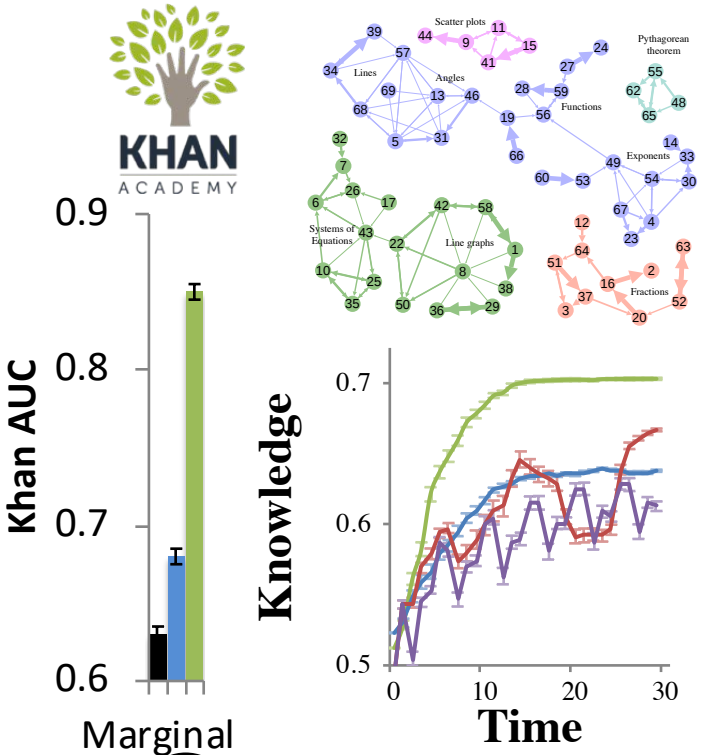
The Last Remaining Board Game

Self Driving Cars

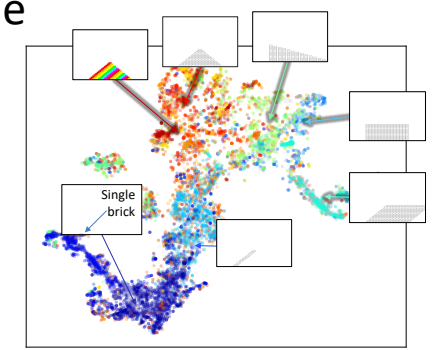

Clear
Societal Need
Increase education productivity
Grand Challenge in Education
Autonomously support education by better understanding students.
MOOC assignments
Deep Learning
New Datasets of Learning
Applied Math Renaissance

In Today's Talk

1 Temporal understanding of students



2 Zero-Shot Deep understanding of code

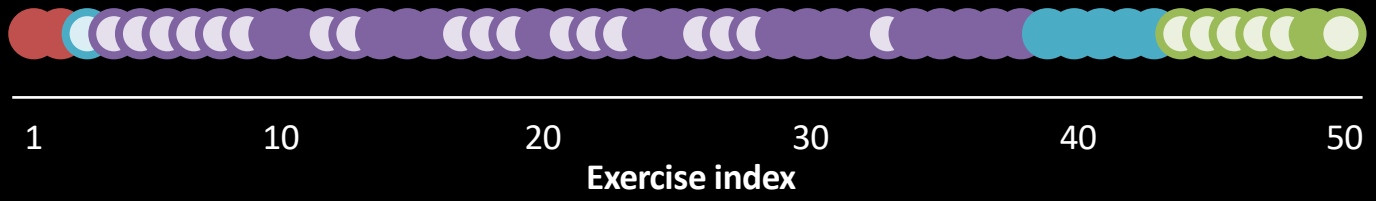


3 Real human impact

4 Taste of next steps for AI in Education

Chapter 1: Deep Knowledge Tracing
Feedback in a simple context

Story of Riley



Exercise Type:

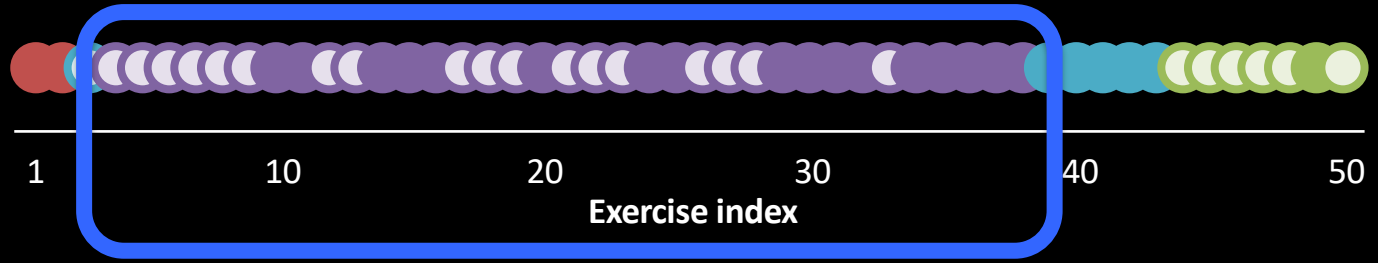
- Solving for x-intercept
- Solving for y-intercept
- Graphing linear equations
- Square roots
- Slope of a line

Answer:

- Correct
- Incorrect

KHAN ACADEMY

Story of Riley



Exercise Type:

- Solving for x-intercept
- Solving for y-intercept
- Graphing linear equations
- Square roots
- Slope of a line

Answer:

- Correct
- Incorrect

KHAN ACADEMY

Story of Riley

Exercise Type:

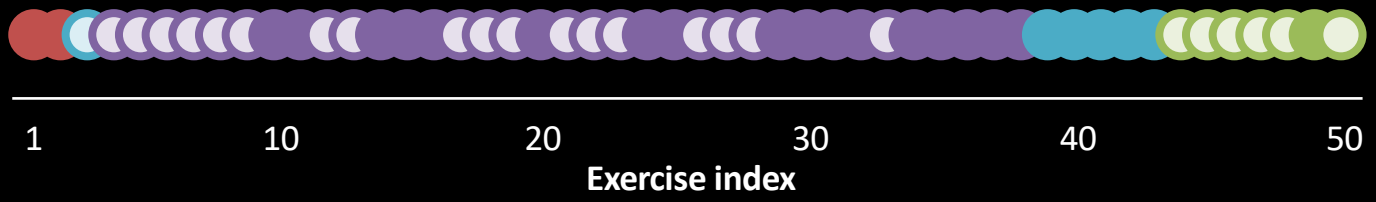
- Solving for x-intercept
- Solving for y-intercept
- Graphing linear equations
- Square roots
- Slope of a line

Answer:

- Correct
- Incorrect

KHAN ACADEMY

Story of Riley



What should Riley do next?

Exercise Type:

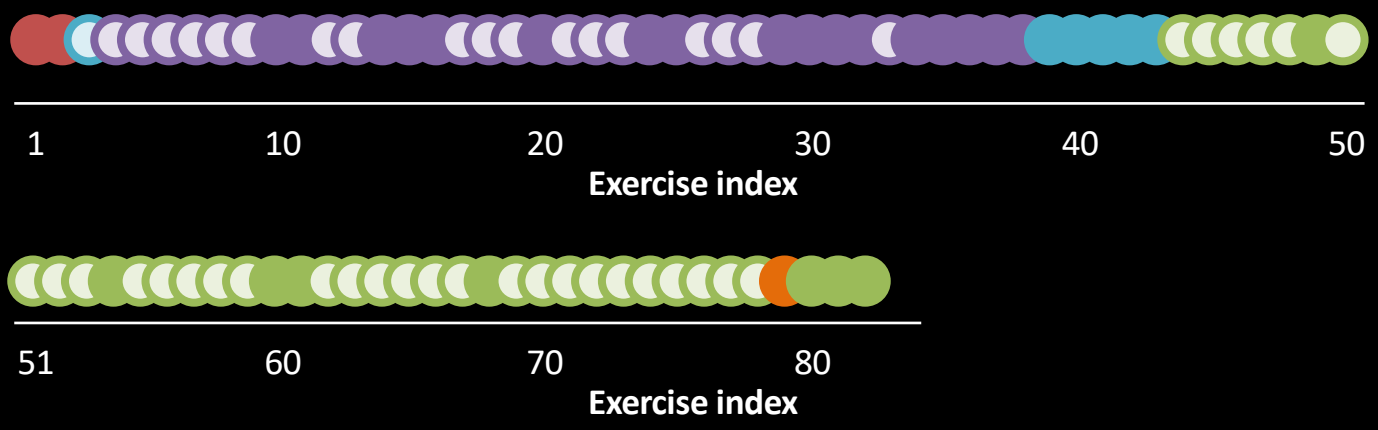
- Solving for x-intercept
- Solving for y-intercept
- Graphing linear equations
- Square roots
- Slope of a line

Answer:

- Correct
- Incorrect

KHAN ACADEMY

Story of Riley



Exercise Type:

- Solving for x-intercept
- Solving for y-intercept
- Graphing linear equations
- Square roots
- Slope of a line

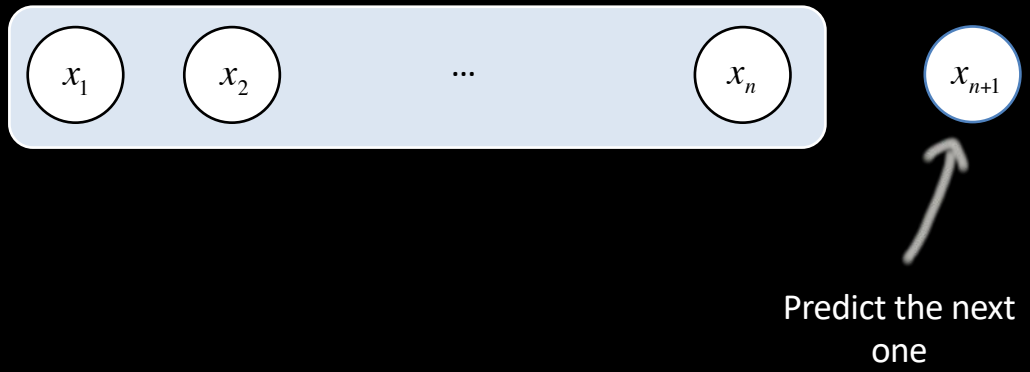
Answer:

- Correct
- Incorrect

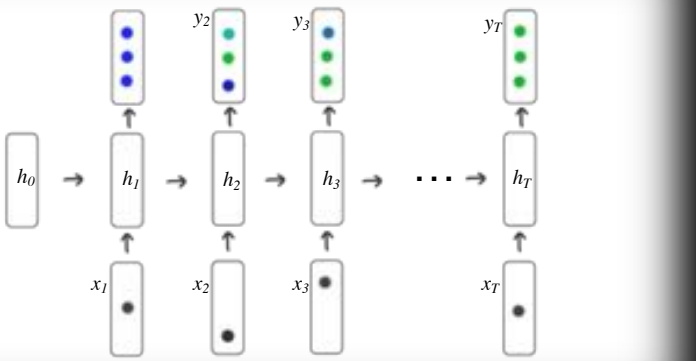
KHAN ACADEMY

Knowledge Tracing for Feedback

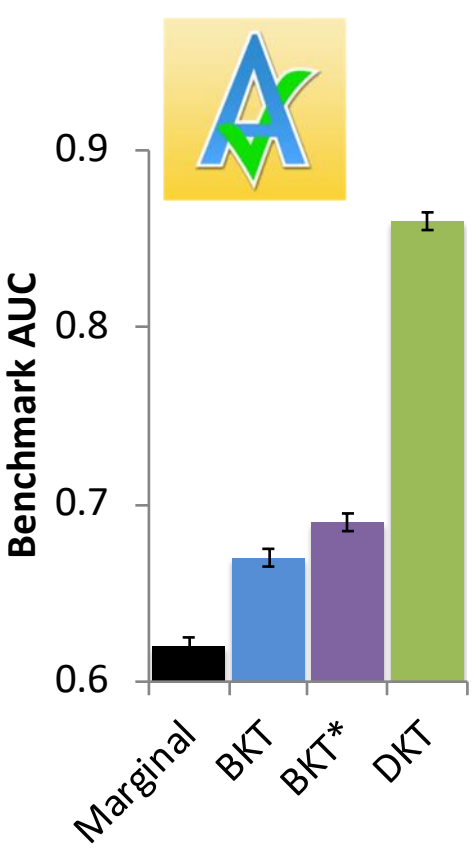
Given *n* historical answers:

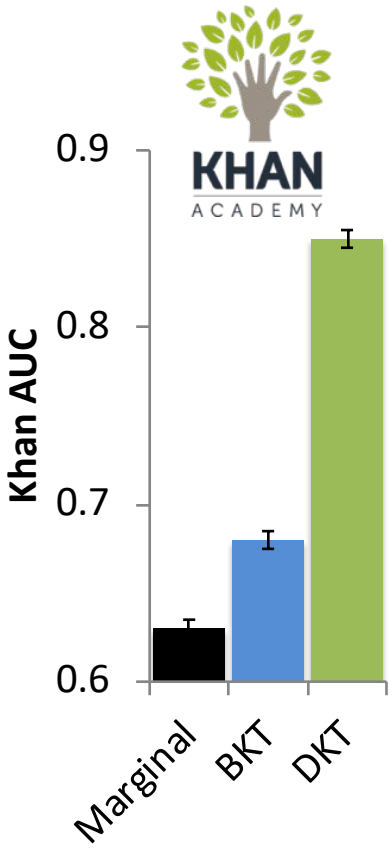


Build the ***first*** deep learning algorithm for human learning

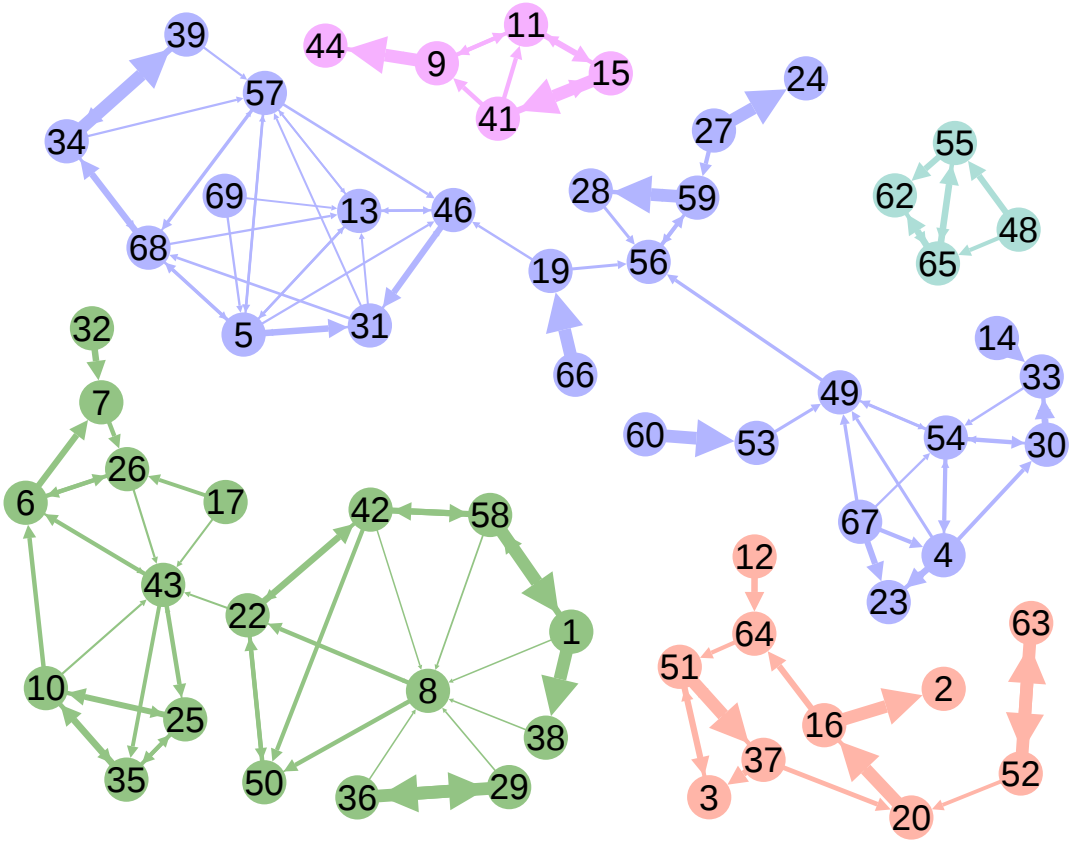


Understanding Students

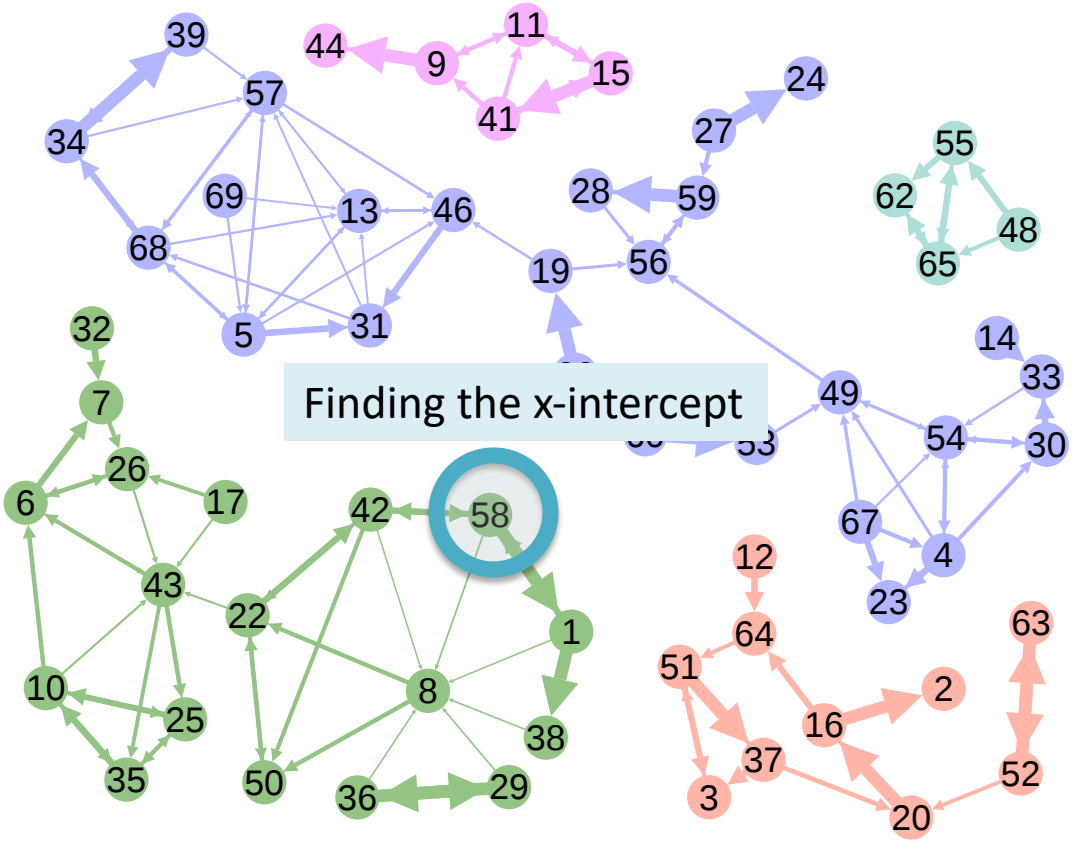




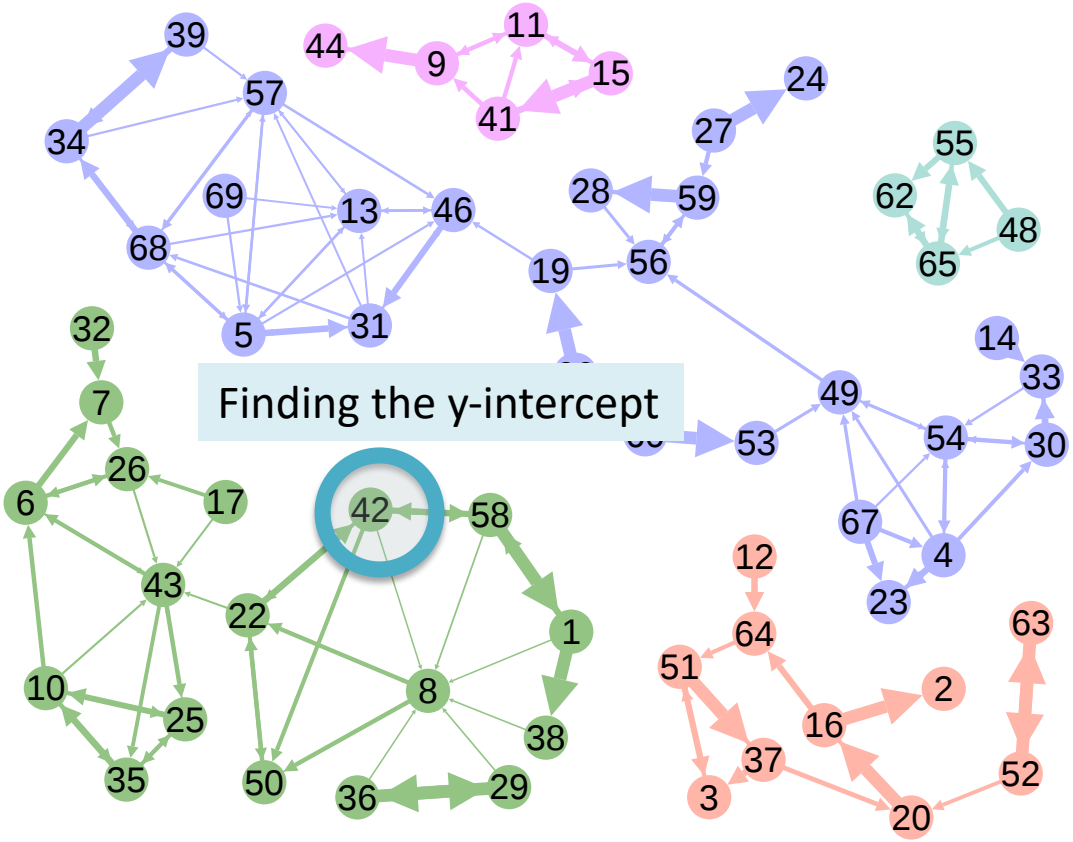
Learns Concept Relationships



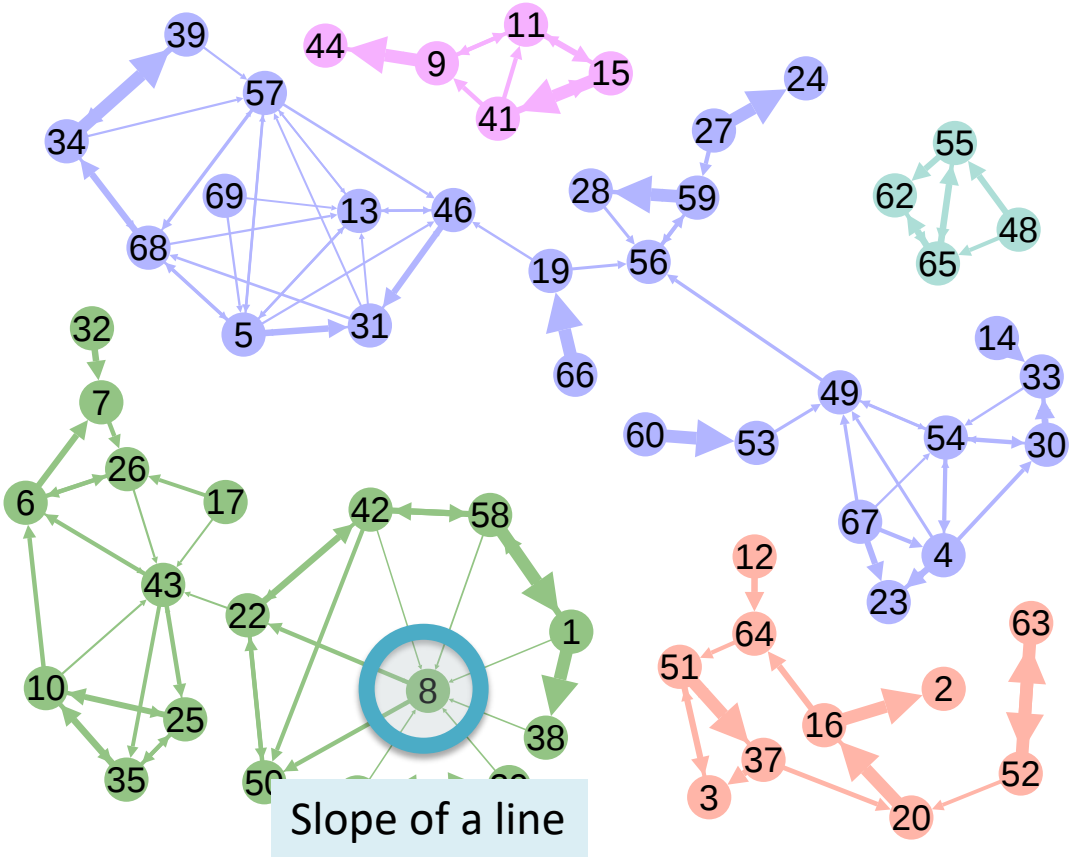
Learns Concept Relationships



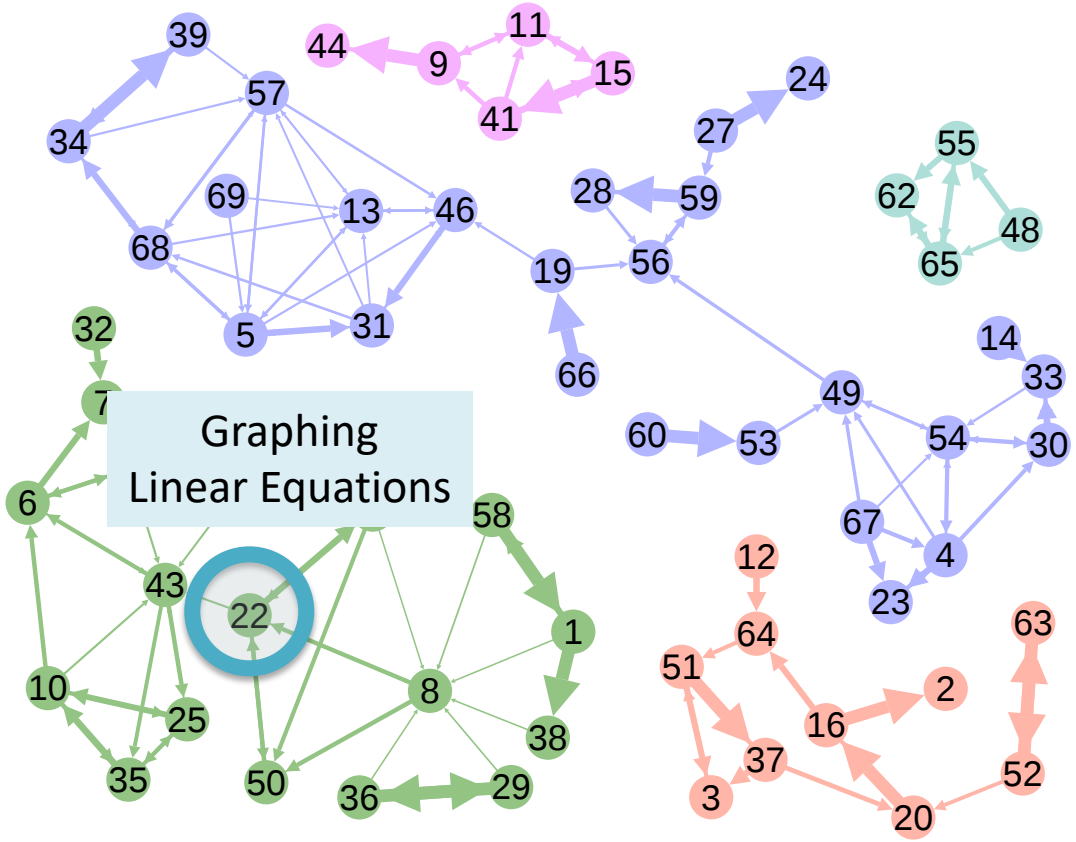
Learns Concept Relationships



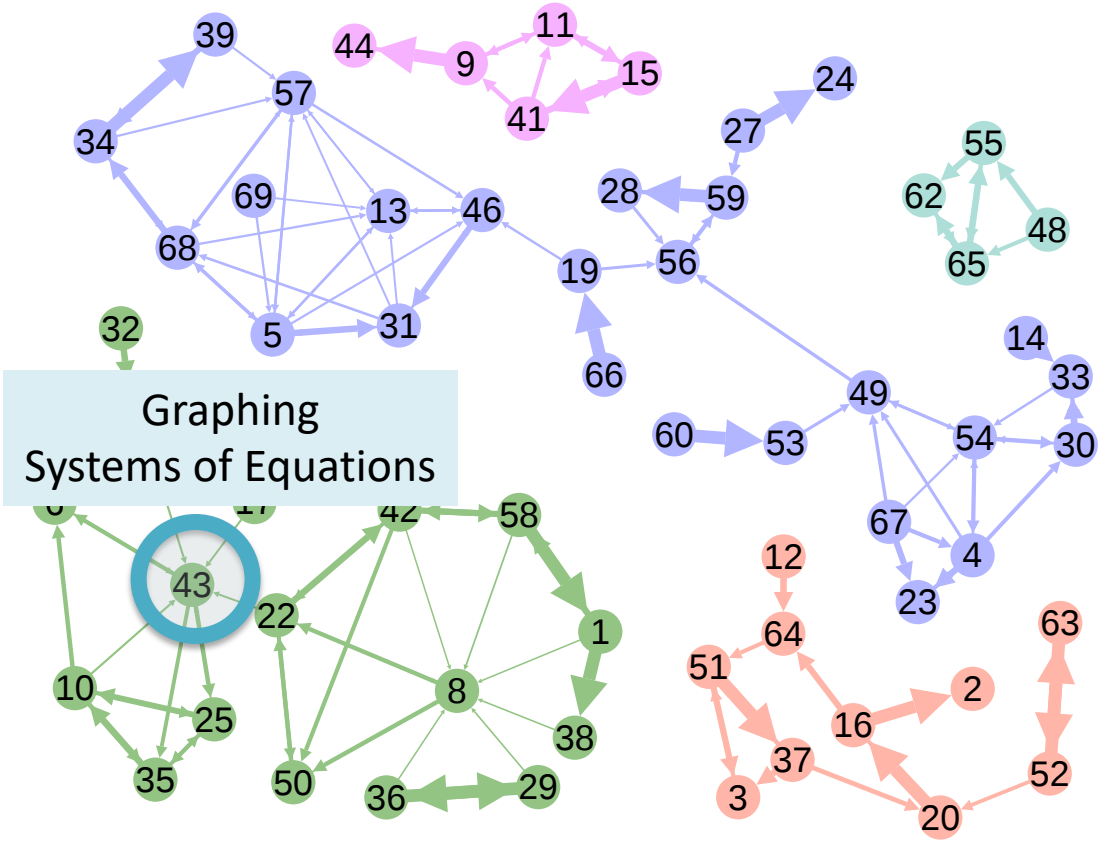
Learns Concept Relationships



Learns Concept Relationships

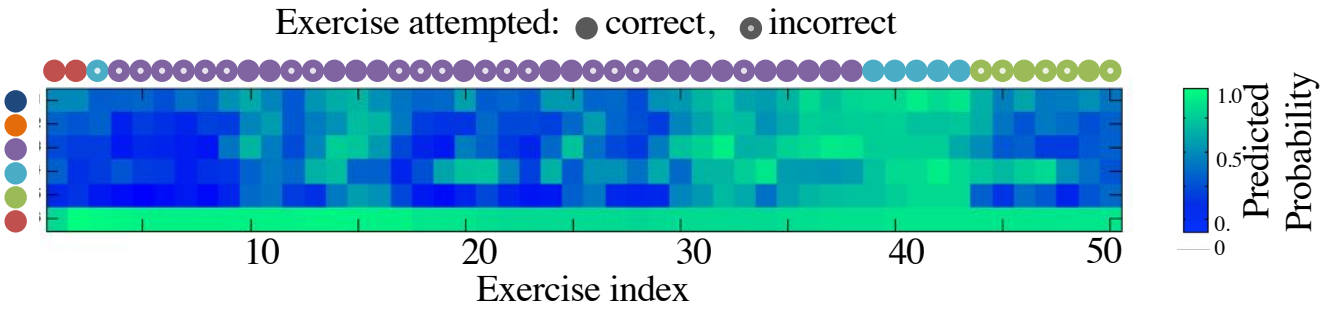


Learns Concept Relationships



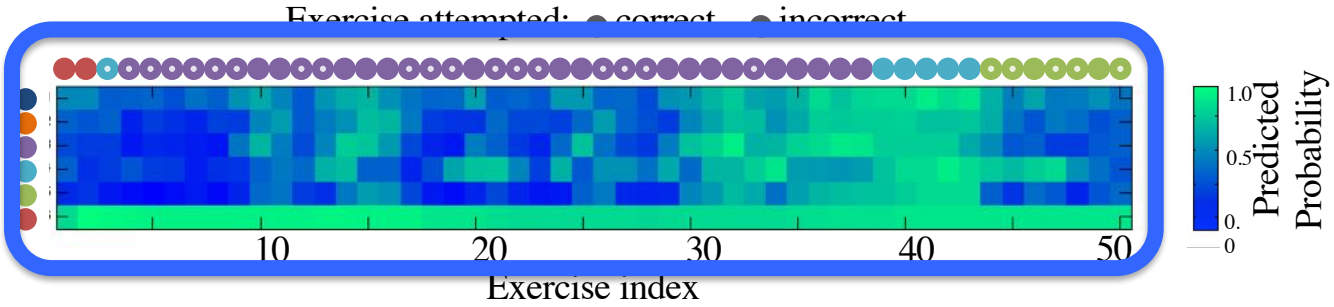
Can it help make decisions?

Predicting Next Item



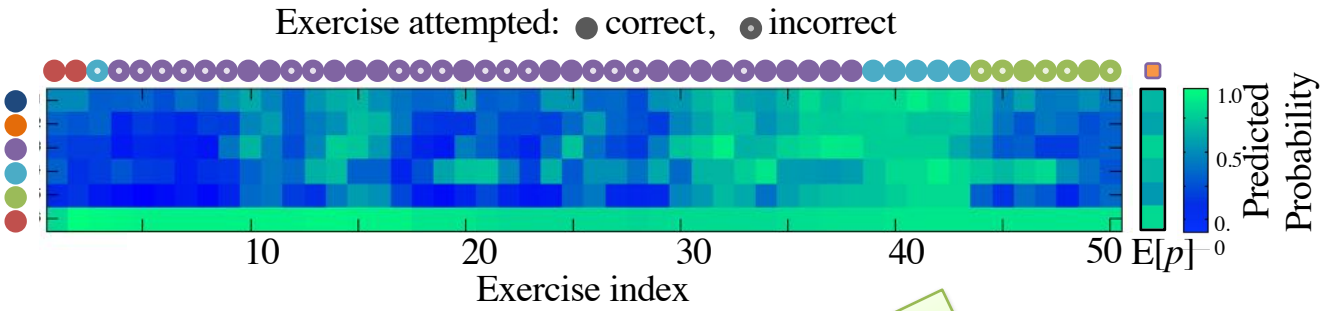
- Line graph intuition
- Slope of a line
- Solving for x-intercept
- Solving for y-intercept
- Graphing linear equations
- Square roots

Predicting Next Item



- Line graph intuition
- Slope of a line
- Solving for x-intercept
- Solving for y-intercept
- Graphing linear equations
- Square roots

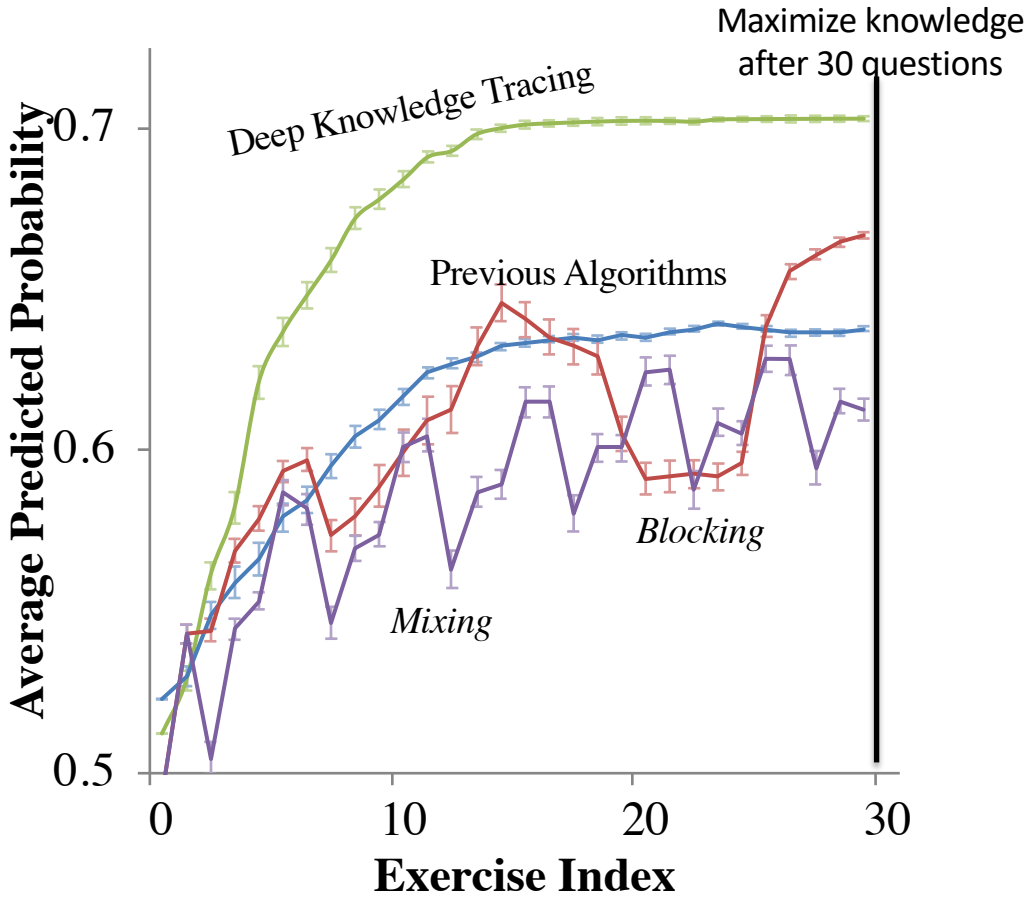
Predicting Next Item



We can find what is the "deal maker"

- Line graph
- Slo
- tercept
- or y-intercept
- raphing linear equations
- Square roots

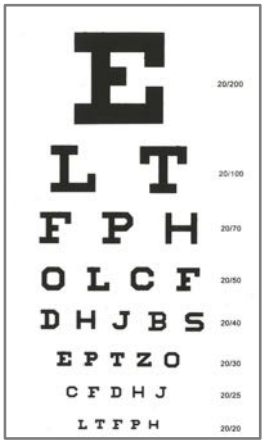
Optimal Teaching



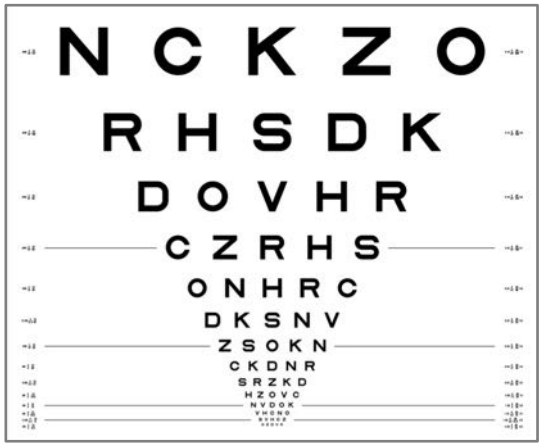
Aside: optimal psychometrics

Visual Acuity Test

Snellen



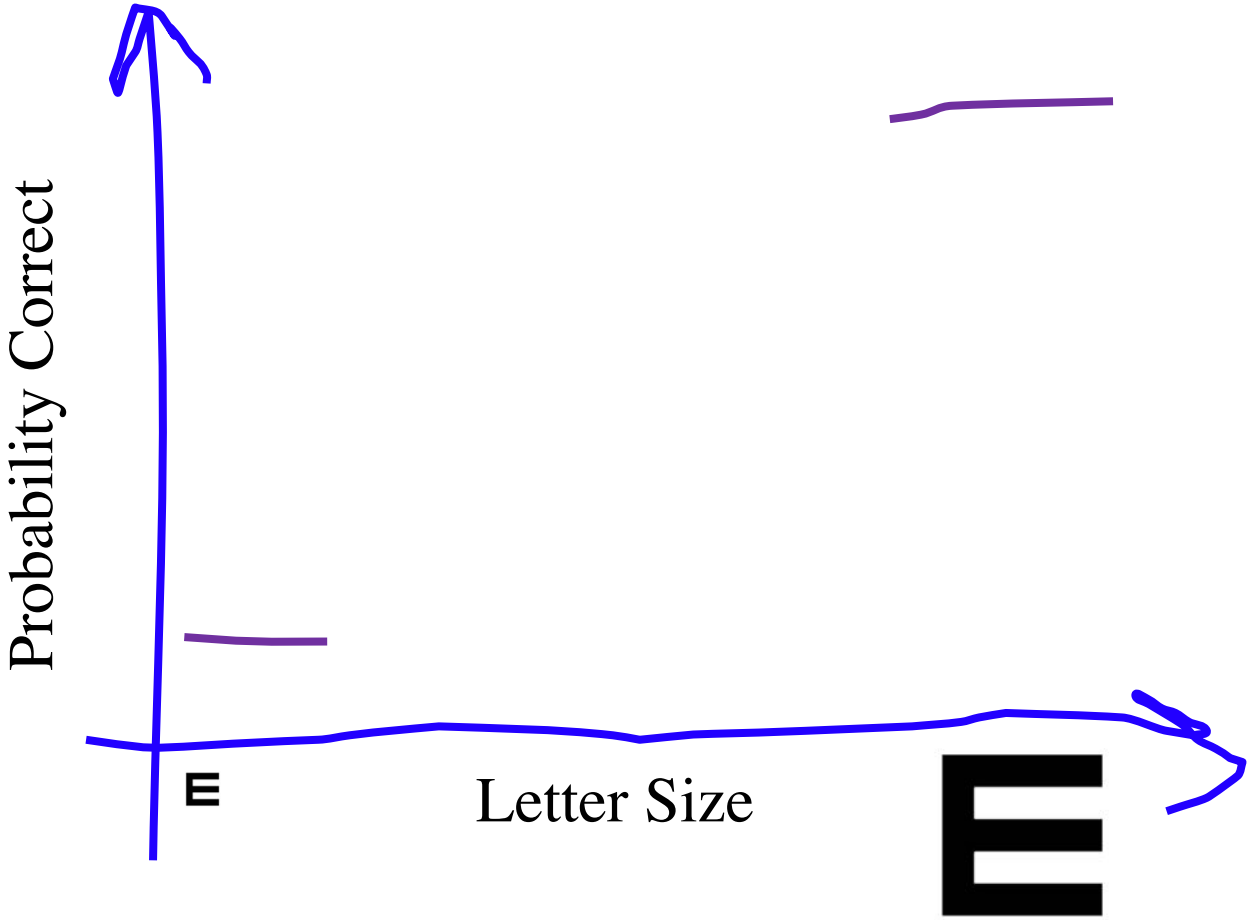
ETDRS



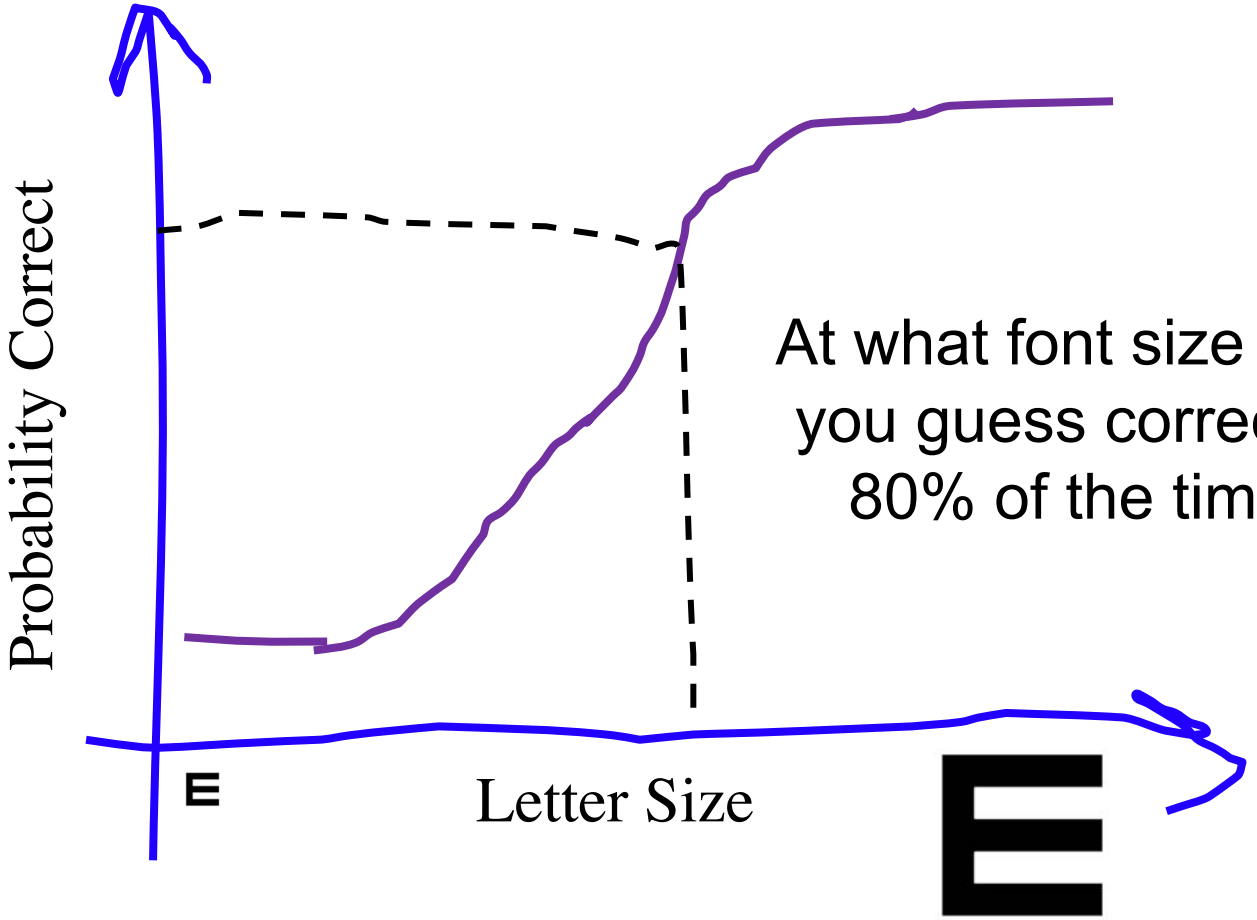
How can we make an accurate, adaptive eye exam?

20 adaptively chosen letter sizes

Visual Acuity Response Function

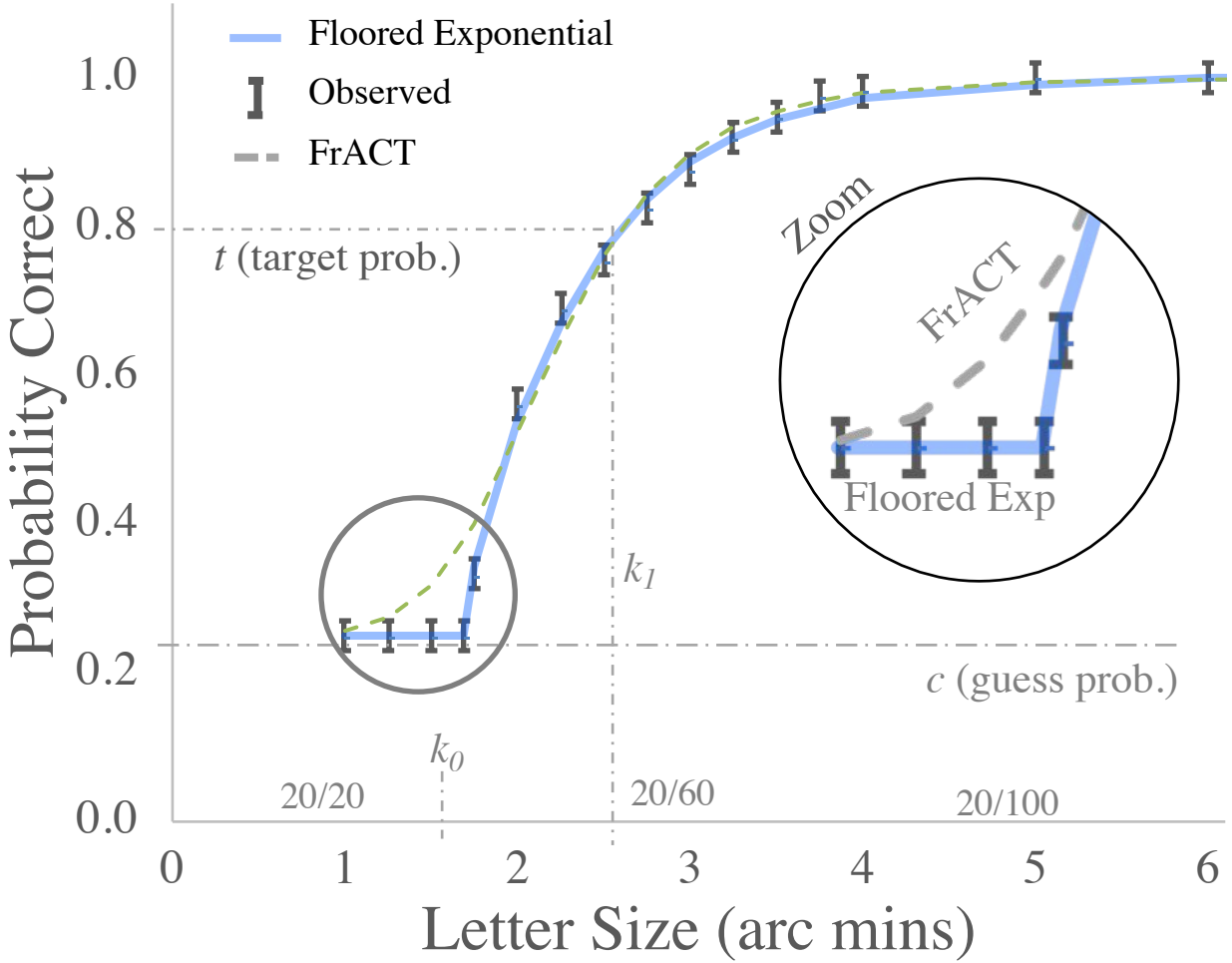


Visual Acuity Response Function

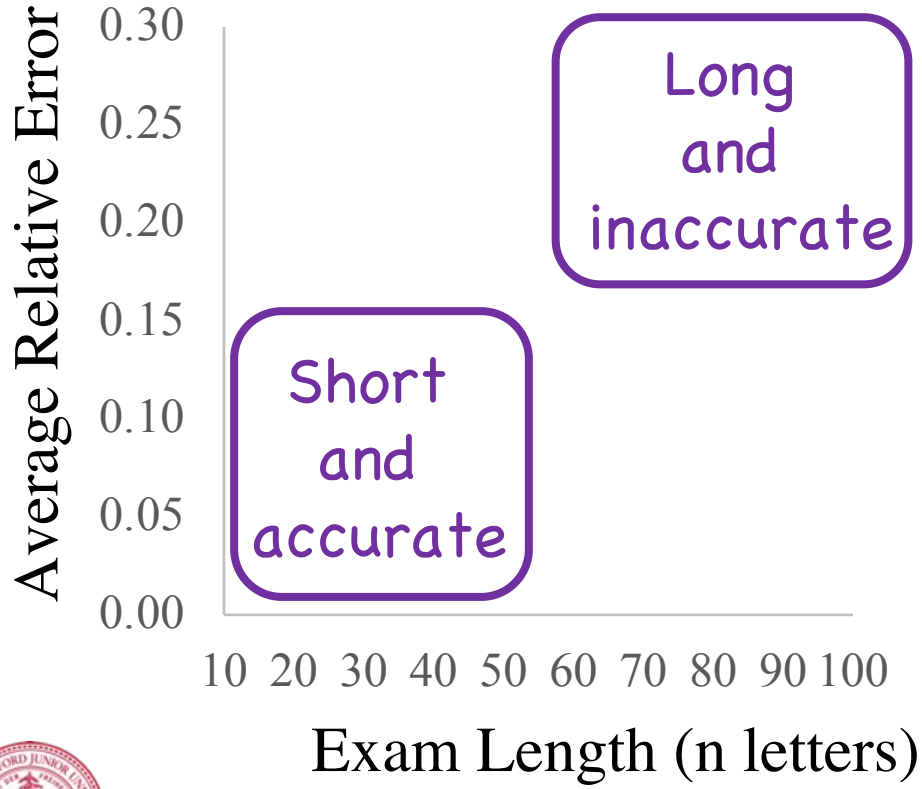


At what font size can you guess correctly 80% of the time

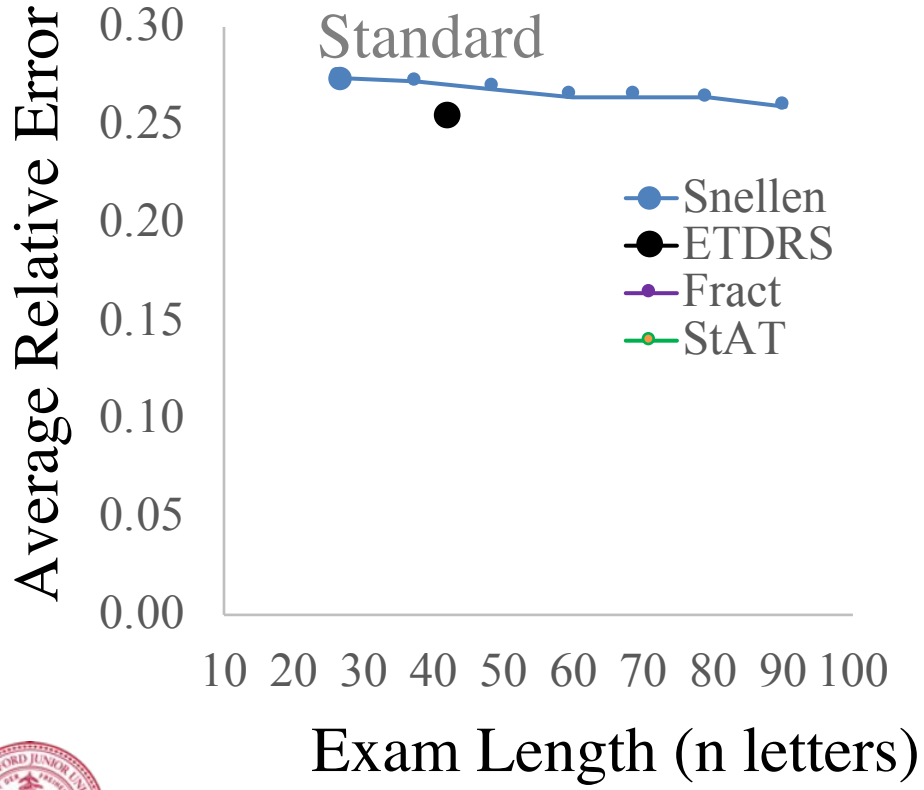
Visual Acuity Response Function



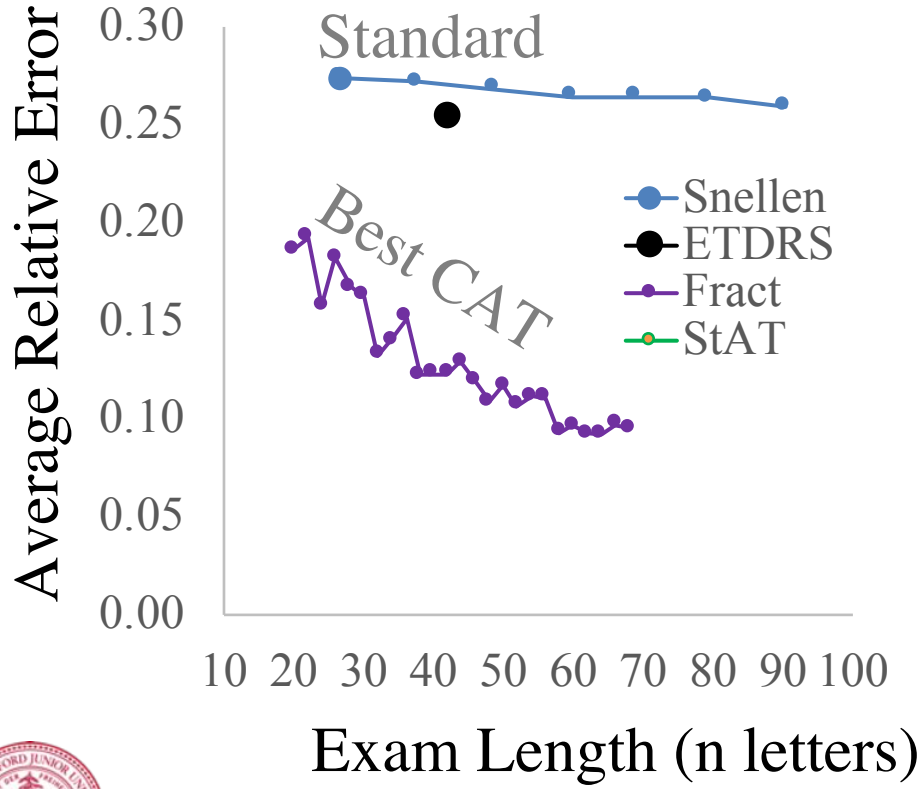
Improved Eye Exam



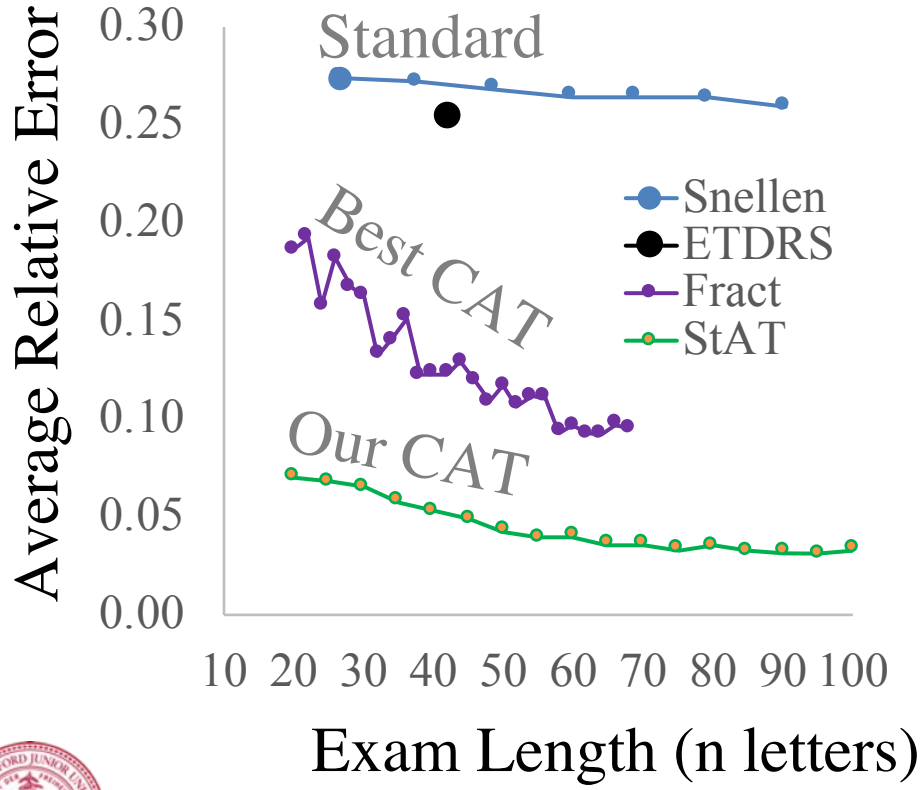
Improved Eye Exam



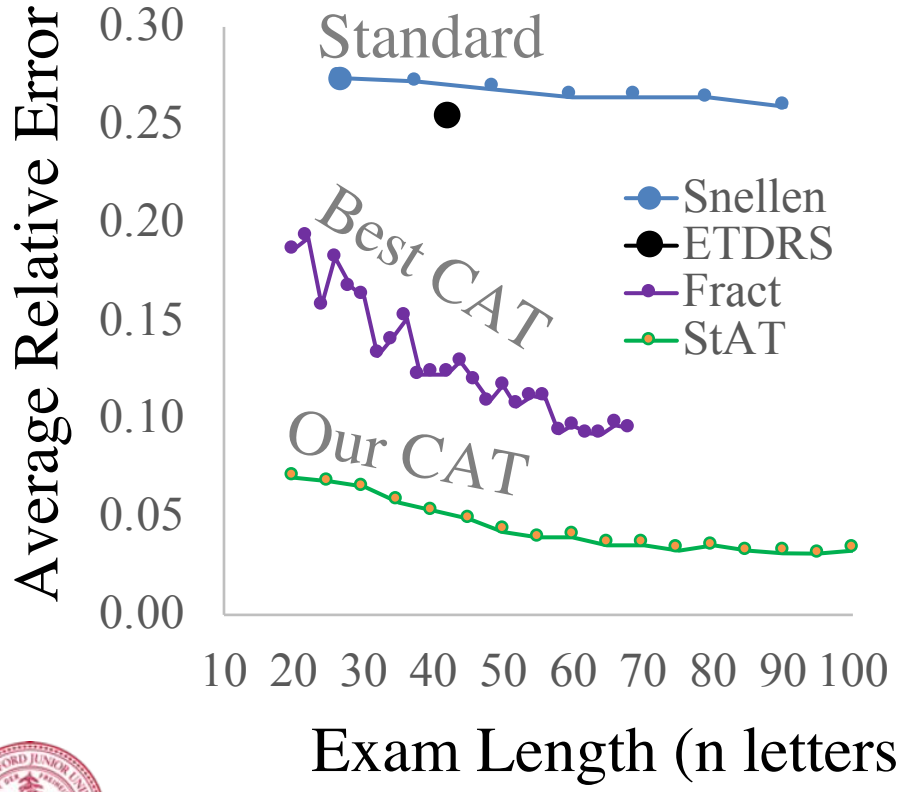
Improved Eye Exam

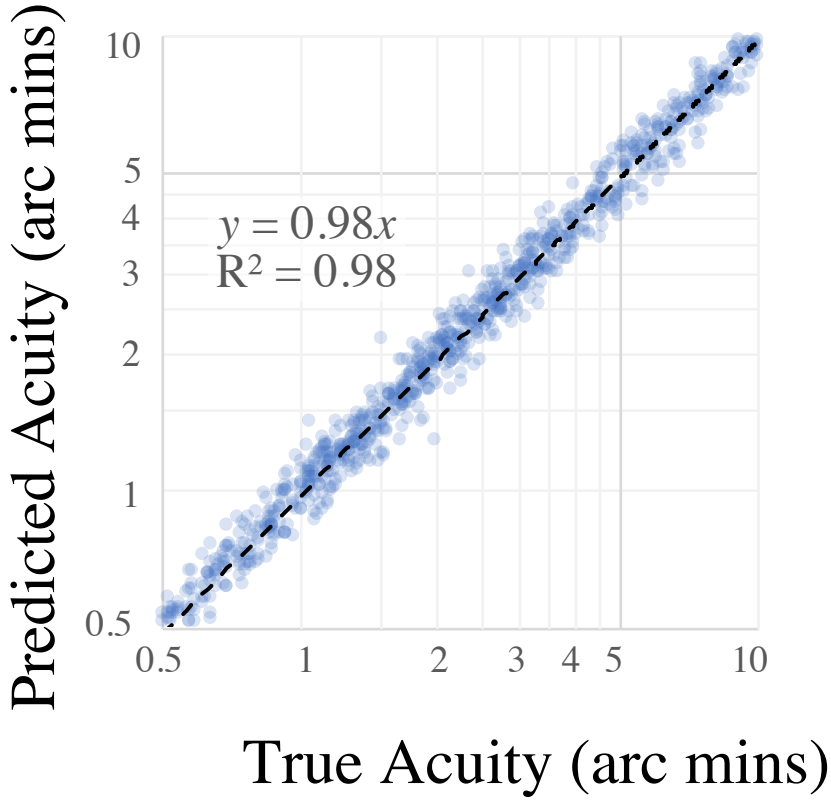


Improved Eye Exam

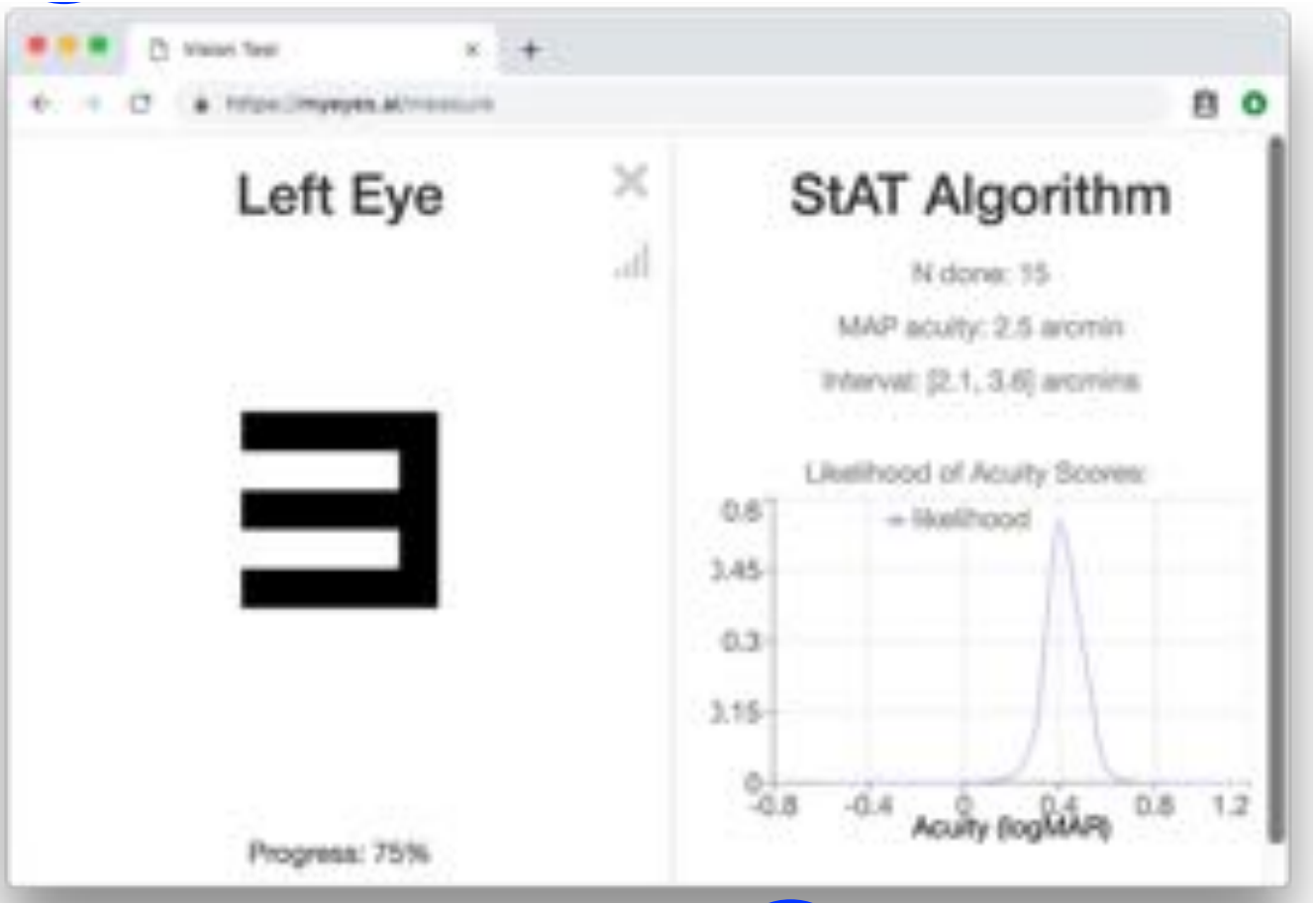


Improved Eye Exam

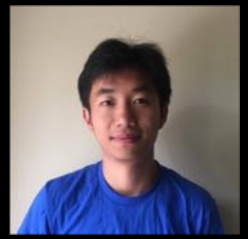




myeyes.ai


Left Eye
StAT Algorithm
N done: 15
MAP acuity: 2.5 arcmin
Interval: [2.1, 3.6] arcmins
Likelihood of Acuity Scores
Likelihood
0.6
0.45
0.3
0.15
0
-0.8
-0.4
0
0.4
0.8
1.2
Acuity (logMAR)
Progress: 75%

Chapter 2: Zero Shot Feedback

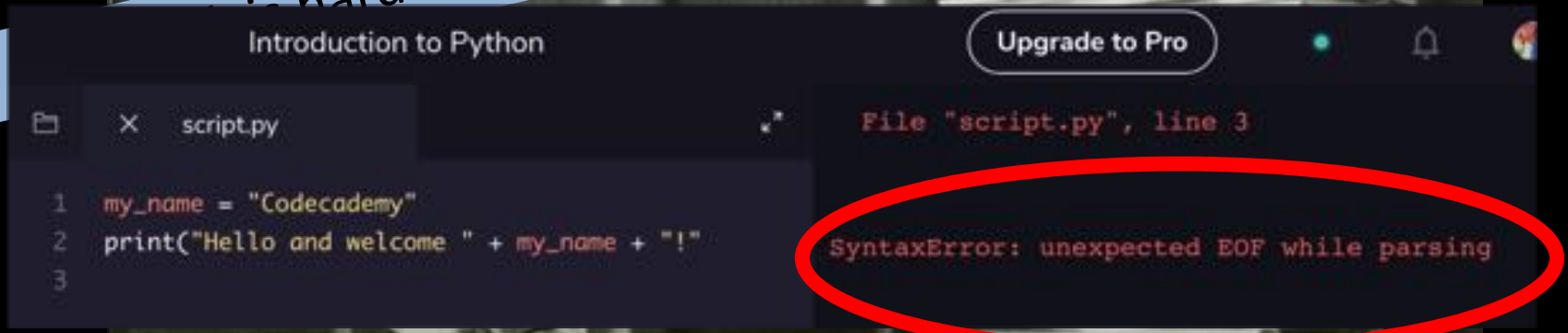


Best student paper at AAAI 2019

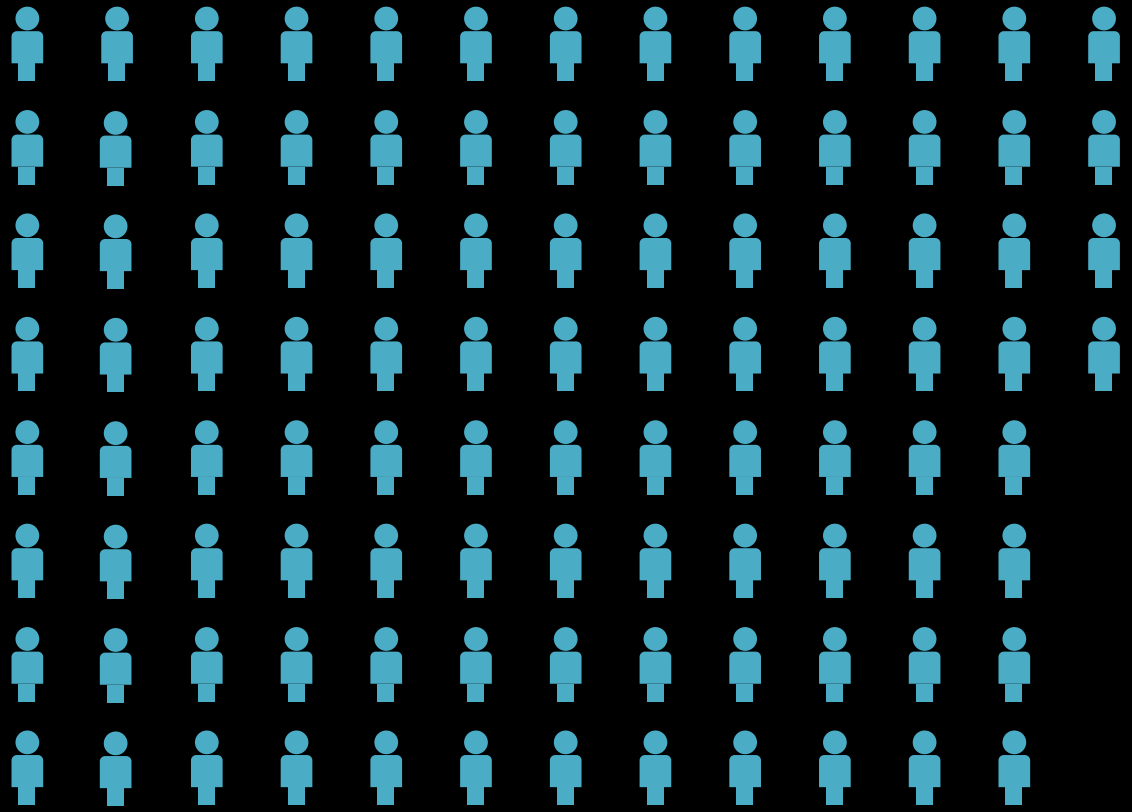
Online Classes Haven't Solved the Problem

Feedback is hard

Online Classes Haven't Solved the Problem

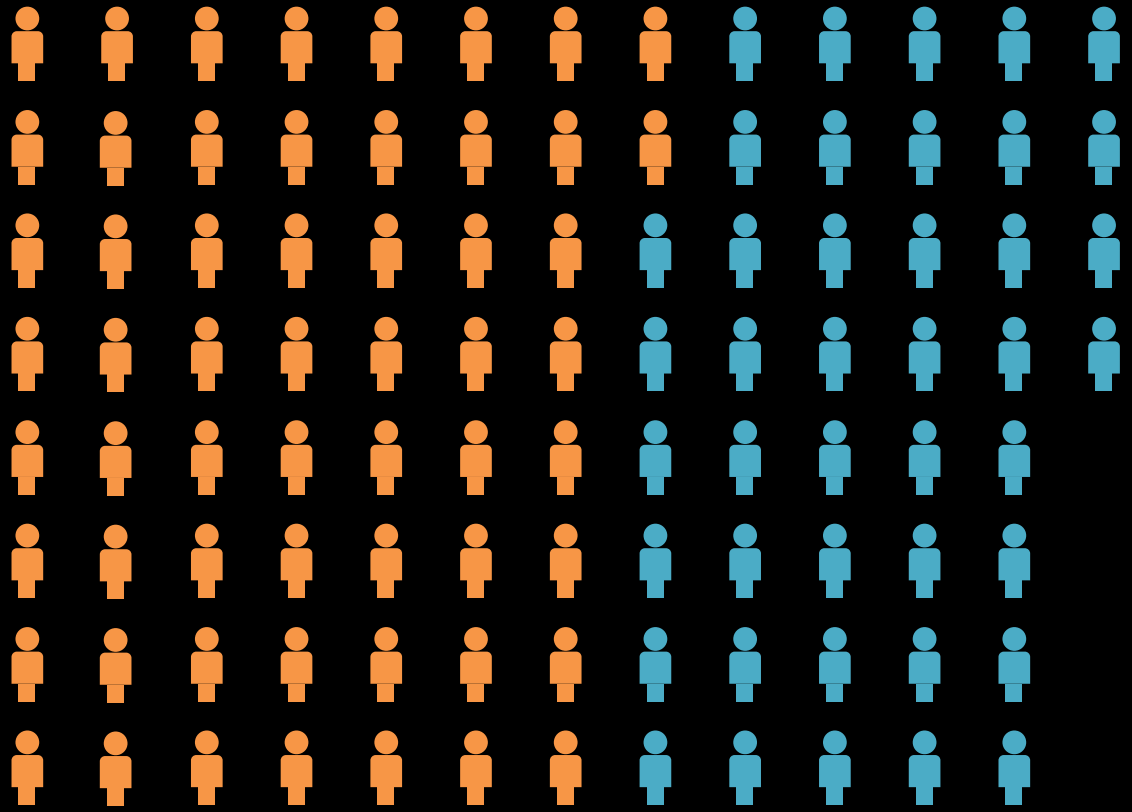


US K12 Students



= 500,000 learners

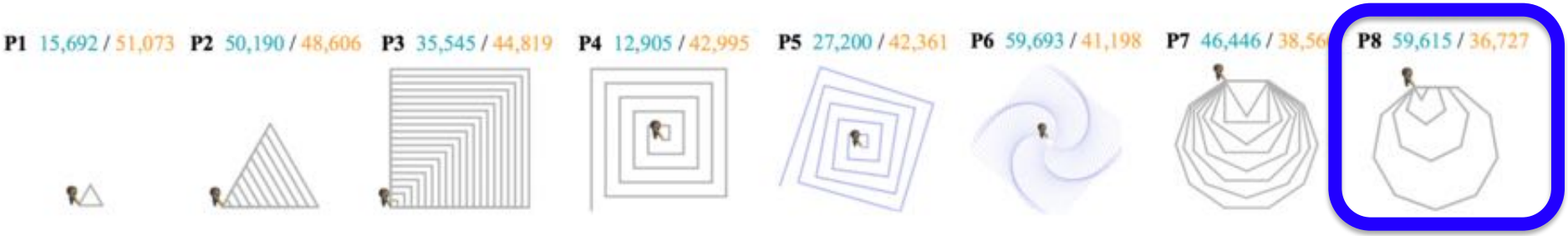
Code.org



= 500,000 learners

The Code.org Dataset

- Students learning **nested loops**
- 50k students with **1.5 million submissions** to a curriculum of **8 exercises**.
- **800 human labels** across 2 of the exercises.



Let's put it all together!

Using your knowledge of `for` loops and the `counter` variable, create this drawing where each shape has two more sides than the last. Make sure that each side is 10 times as long as the number of sides in the polygon.

Very little of the code has been provided for you.

P1 15,692 / 51,073 **P2** 50,190 / 48,606 **P3** 35,545 / 44,819 **P4** 12,905 / 42,995 **P5** 27,200 / 42,361 **P6** 59,693 / 41,198 **P7** 46,446 / 38,56 **P8** 59,615 / 36,727

Let's put it all together!

Using your knowledge of `for` loops and the `counter` variable, create this drawing where each shape has two more sides than the last. Make sure that each side is 10 times as long as the number of sides in the polygon.

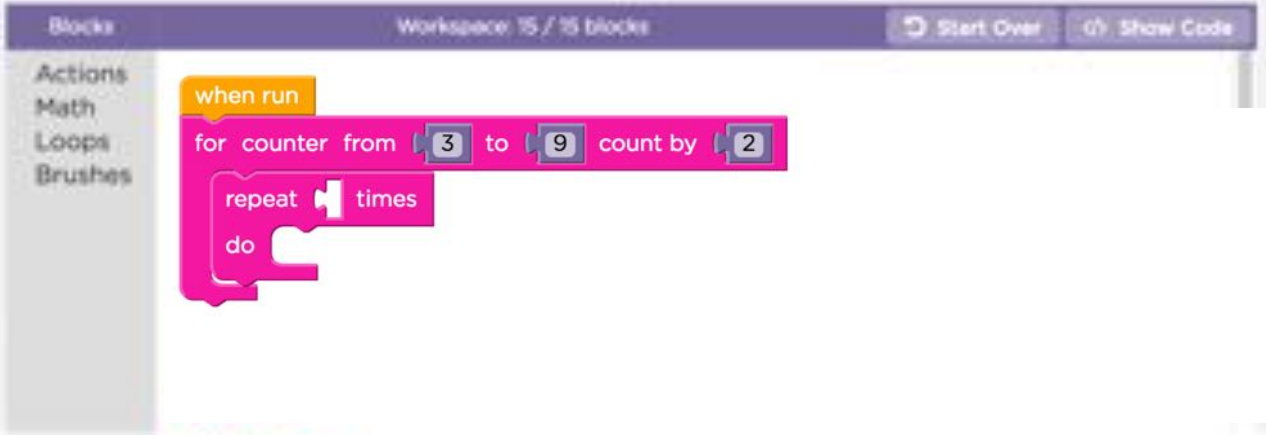
Very little of the code has been provided for you.

P1 15,692 / 51,073 **P2** 50,190 / 48,606 **P3** 35,545 / 44,819 **P4** 12,905 / 42,995 **P5** 27,200 / 42,361 **P6** 59,693 / 41,198 **P7** 46,446 / 38,56 **P8** 59,615 / 36,727

Let's put it all together!

Using your knowledge of `for` loops and the `counter` variable, create this drawing where each shape has two more sides than the last. Make sure that each side is 10 times as long as the number of sides in the polygon.

Very little of the code has been provided for you.



P1 15,692 / 51,073 **P2** 50,190 / 48,606 **P3** 35,545 / 44,819 **P4** 12,905 / 42,995 **P5** 27,200 / 42,361 **P6** 59,693 / 41,198 **P7** 46,446 / 38,56 **P8** 59,615 / 36,727

Let's put it all together!

Using your knowledge of `for` loops and the `counter` variable, create this drawing where each shape has two more sides than the last. Make sure that each side is 10 times as long as the number of sides in the polygon.

Very little of the code has been provided for you.

Blocks | Workspace: 15 / 15 blocks | Start Over | Show Code

Actions
Math
Loops
Brushes

```
when run
for counter from 3 to 9 count by 2
    repeat counter times
    do move forward by counter * 10 pixels
       turn right by 360 / counter degrees
```

Run

P1 15,692 / 51,073 **P2** 50,190 / 48,606 **P3** 35,545 / 44,819 **P4** 12,905 / 42,995 **P5** 27,200 / 42,361 **P6** 59,693 / 41,198 **P7** 46,446 / 38,56 **P8** 59,615 / 36,727

Human Labelled 800

Top Secret

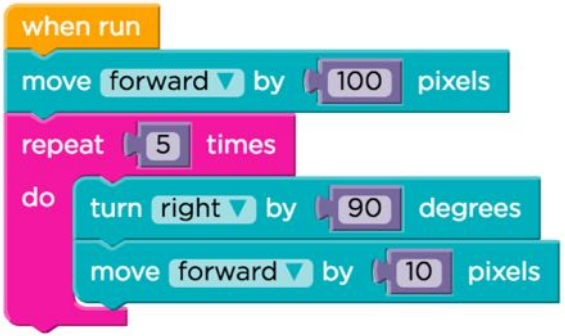
Not Secure stanford.edu/~cpiech/restricted/codeDotOrgLabels/#/c/p

Question Solution

Instructions

- If there are many moves, focus on the first one
- `Random code` strategy is for when the student seems to be trying things randomly
- Lookout for students who don't get nesting or pre/post conditions. Often extra blocks in a body is an indication that they don't get that the post of the loop has to match the precondition

Question



Student 0

Label Console

✔ Num Done: 8273

Strategy

- ○ Beeper Boundry (most people do this)
- ◉ Triangle Strategy
- ○ Recursive Strategy

Looping

- ☐ Correct use of looping
- ☐ Doesn't use a while
- ☑ Doesn't have correct stop condition
- ☐ Body is missing statements
- ☐ Body has extra statements
- ☐ Body order is incorrect
- ☐ Sets up initial precondition
- ☐ Does not get nesting
- ☑ Loop post condition doesn't match preconditi
- ☐ Repetition of bodies

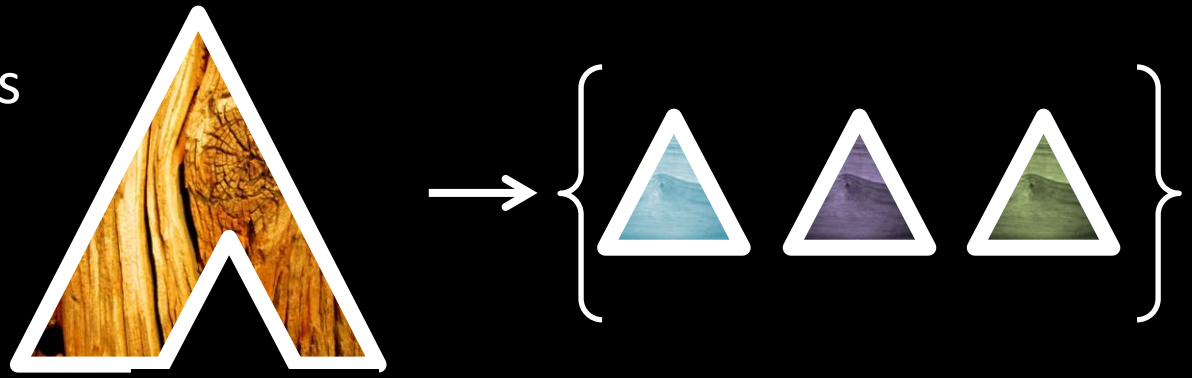
Cleanup

Record label

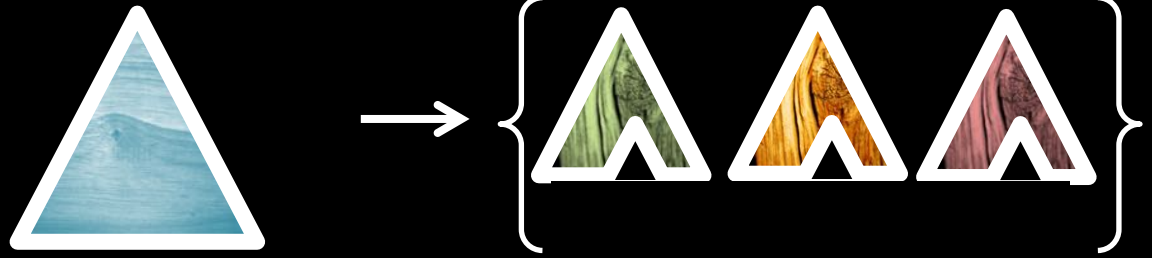
It is a very hard problem

Terribly Clever Static Analysis

A student's program



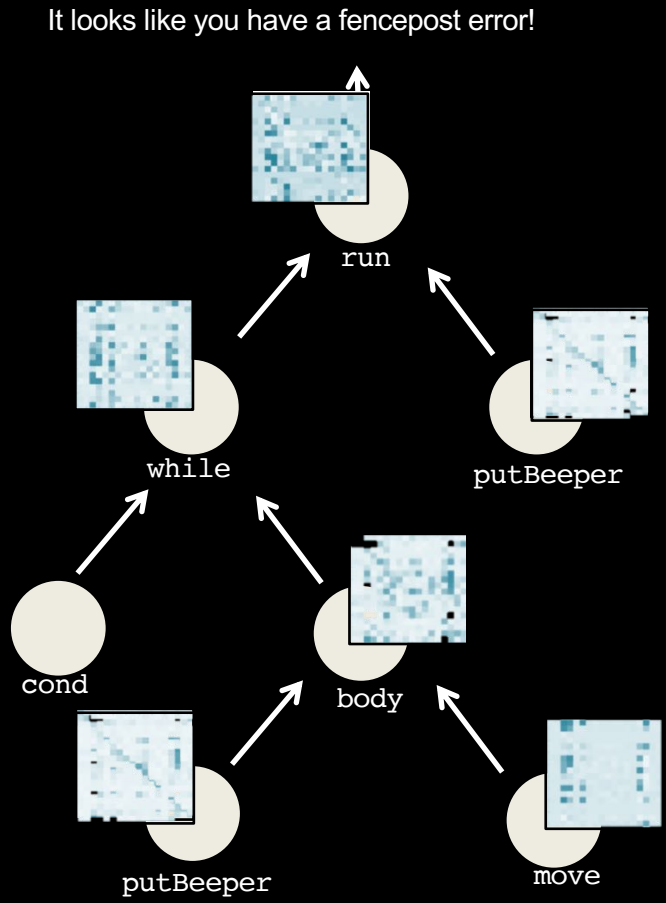
A bug



Nguyen et Al, WWW 2014

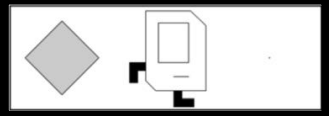
Encode a Program

```
// User defined method
private void run() {
   while(isClear()){
      putBeeper();
      move();
   }
   putBeeper();
}
```

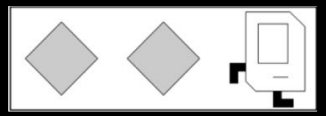


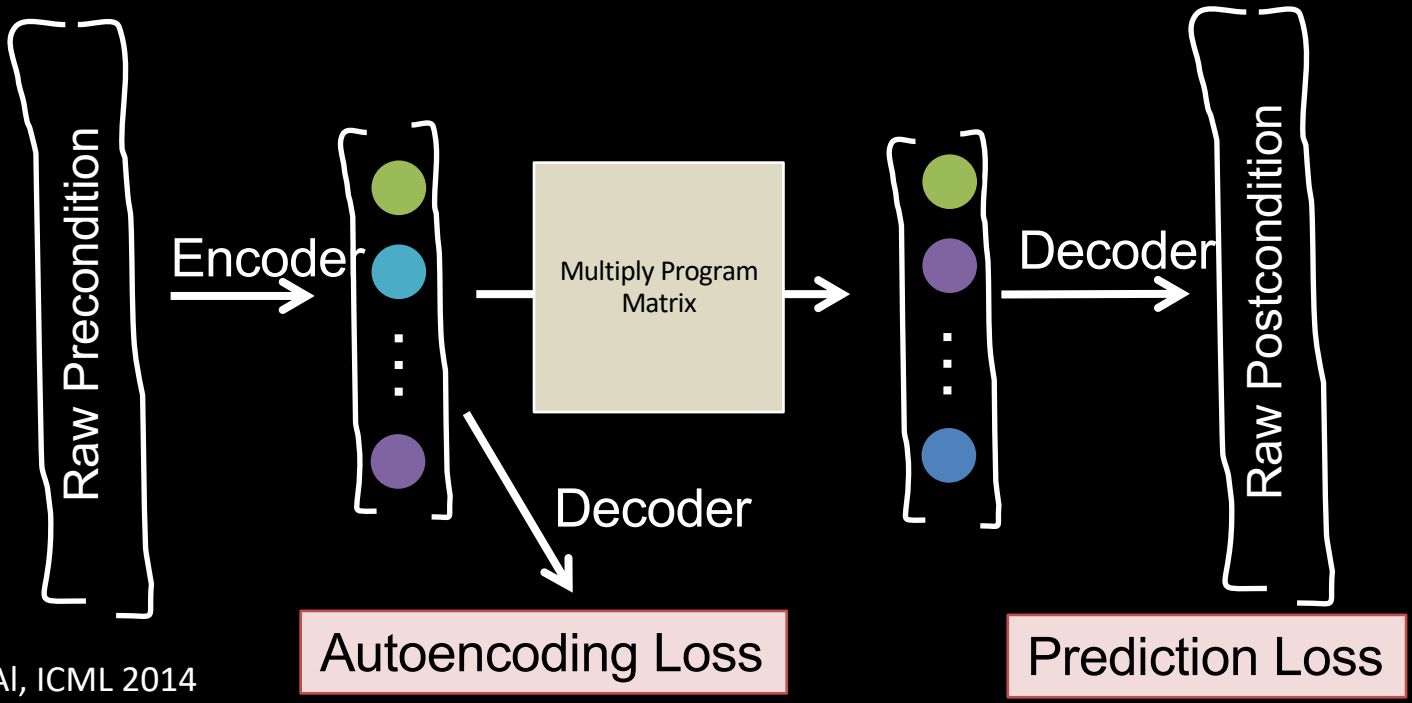
Piech et Al, ICML 2014

Neural Network for Programs

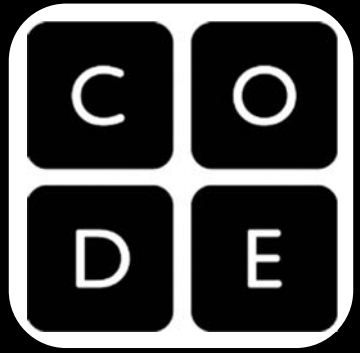


```
method step() {
    putBeeper();
    move();
}
```





Piech et Al, ICML 2014



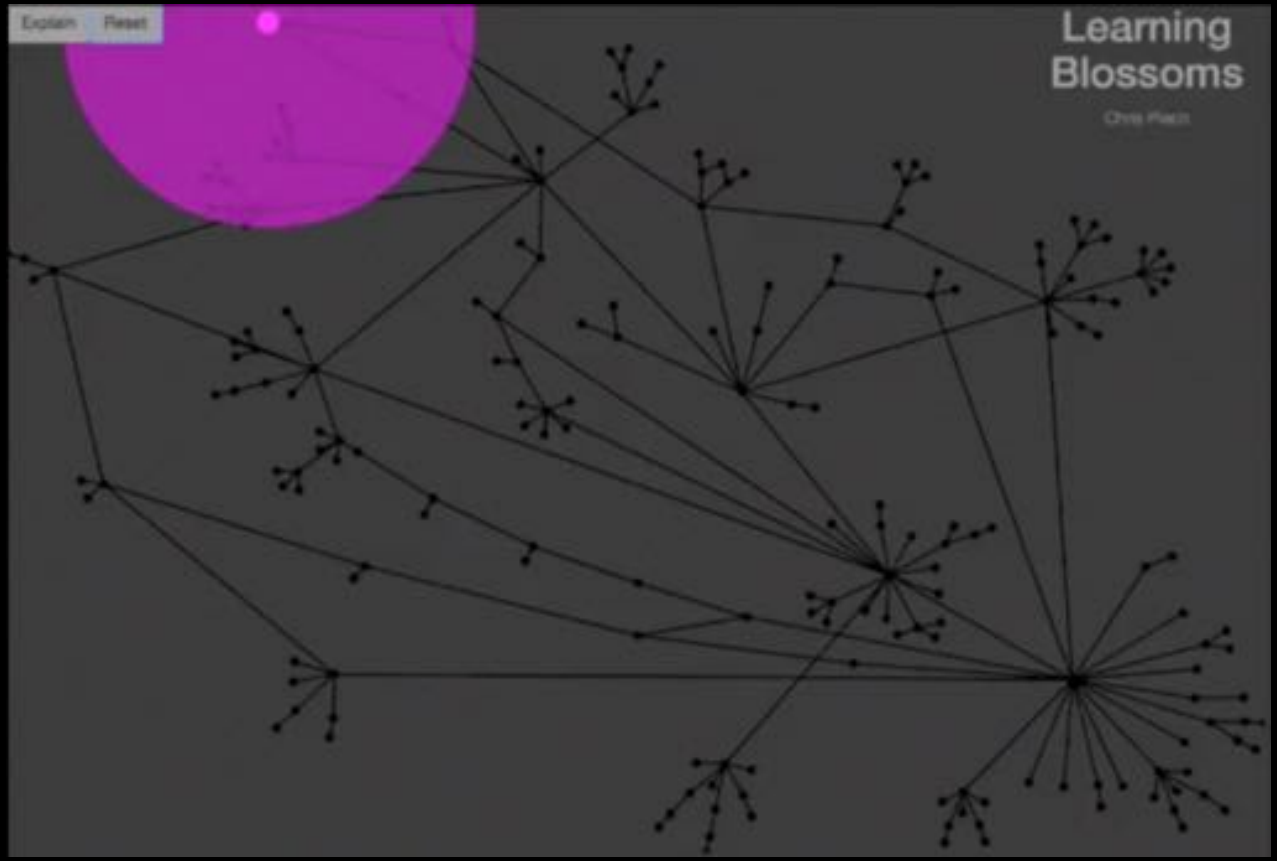
Can we provide feedback by dynamic analysis?

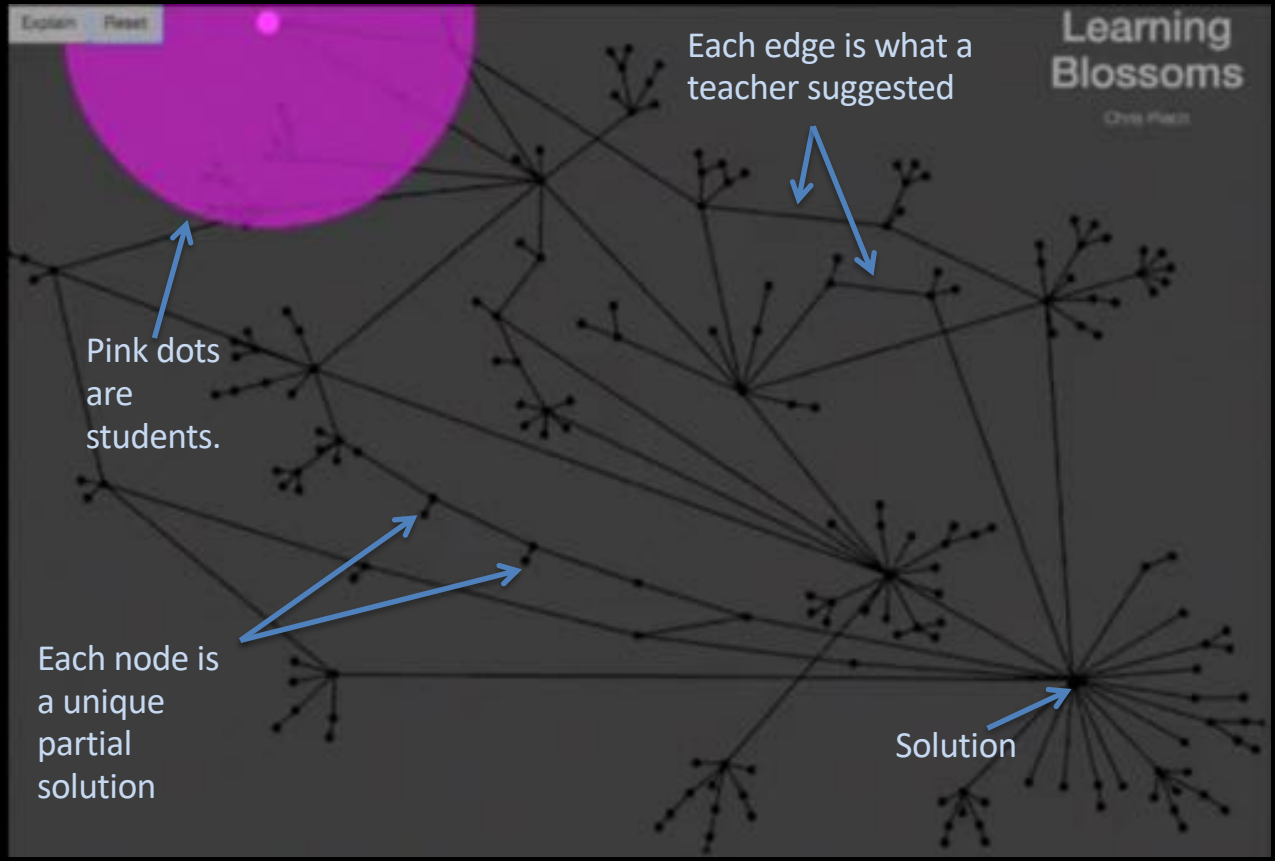
Starter code

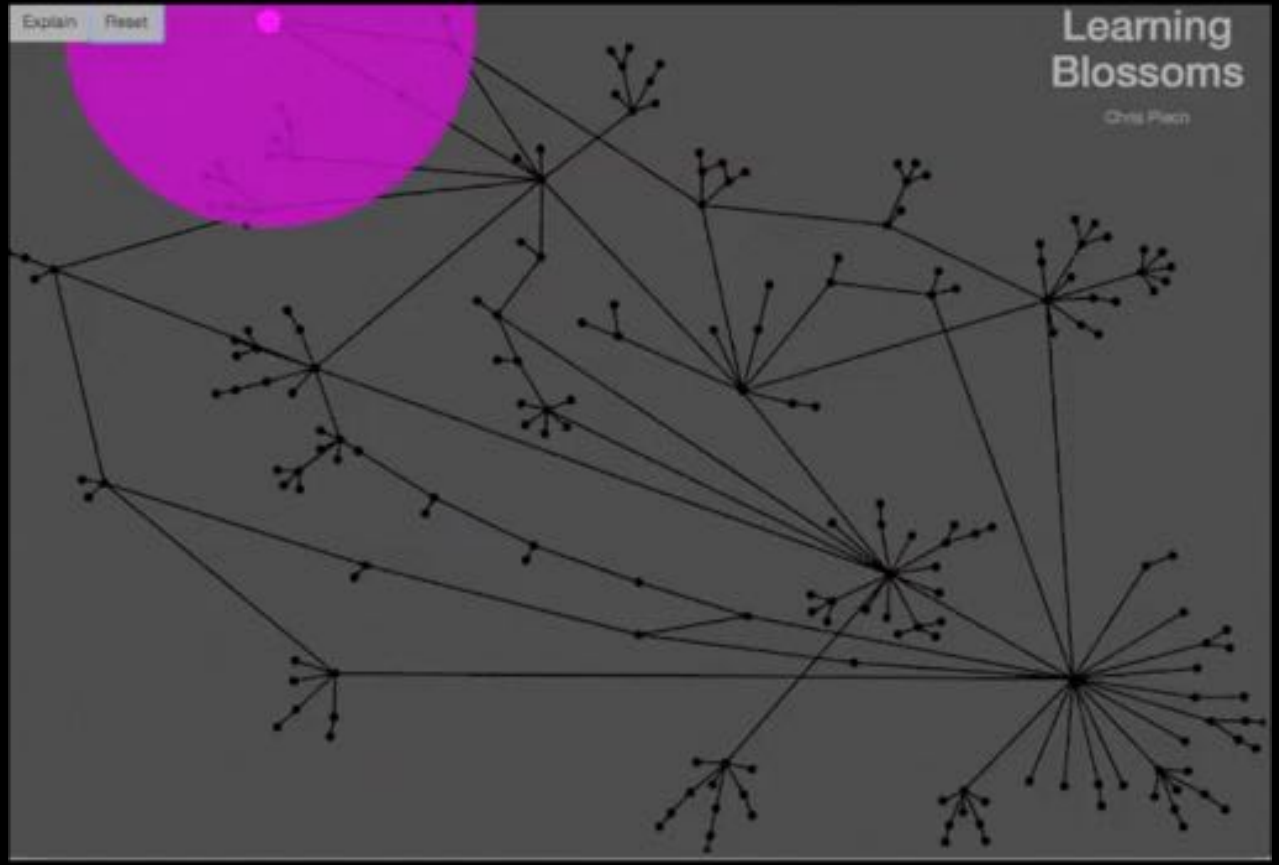
First attempt

• • •

Final solution


Explain
Reset
Learning
Blossoms


Explain
Reset
Learning Blossoms
Each edge is what a teacher suggested
Pink dots are students.
Each node is a unique partial solution
Solution


Explain
Reset
Learning
Blossoms
Chris Plech

The Crowd is Un-wise

Temporal methods tried:
Shortest path
Min Time
Expected Success
Reinforcement learning
Most Common Next
Most Popular Path

18%

45%

12%

State of the Art of this well studied problem

SOTA uses RNNs to vectorize programs and classify among K feedback classes.

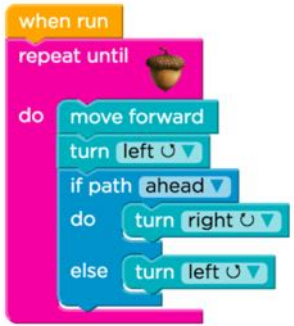
These models are ...

- *Far* from human accuracy.
- *Uninterpretable*... why pick the feedback it did?
- Require *lots* of labeled examples.

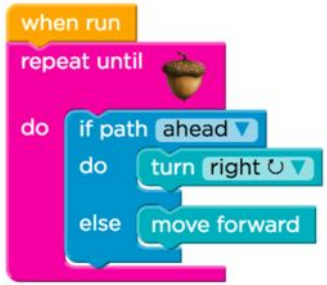
Wang et al, EDM 2017

What Matters for Students?

1. Two compound errors



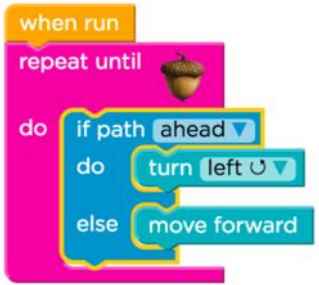
2. Solves first error



3. Starts reasonable attempt

4. Completes attempt

5. Backtracks

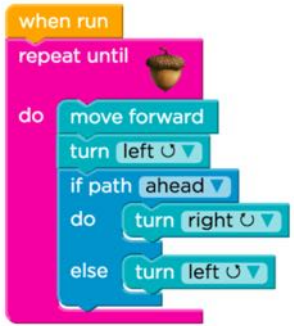


6. Finds solution

Lisa Wang, Chris Piech. Deep Knowledge Tracing on Programming Exercises. L@S 2017

Highly Rates Grit

1. Two compound errors



2. Solves first error

3. Starts reasonable attempt

4. Completes attempt

5. Backtracks

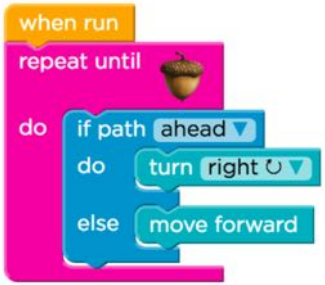
6. Finds solution

Lisa Wang, Chris Piech. Deep Knowledge Tracing on Programming Exercises. L@S 2017

Highly Rates Grit

1. Two compound errors

2. Solves first error



3. Starts reasonable attempt

4. Completes attempt

5. Backtracks

6. Finds solution

Lisa Wang, Chris Piech. Deep Knowledge Tracing on Programming Exercises. L@S 2017

Highly Rates Grit

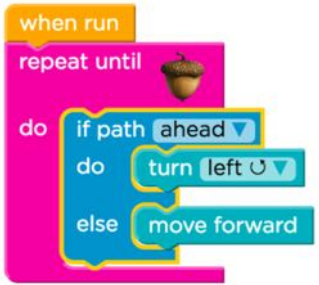
1. Two compound errors

2. Solves first error

3. Starts reasonable attempt

4. Completes attempt

5. Backtracks



6. Finds solution

Lisa Wang, Chris Piech. Deep Knowledge Tracing on Programming Exercises. L@S 2017

Highly Rates Grit

1. Two compound errors

2. Solves first error

3. Starts reasonable attempt

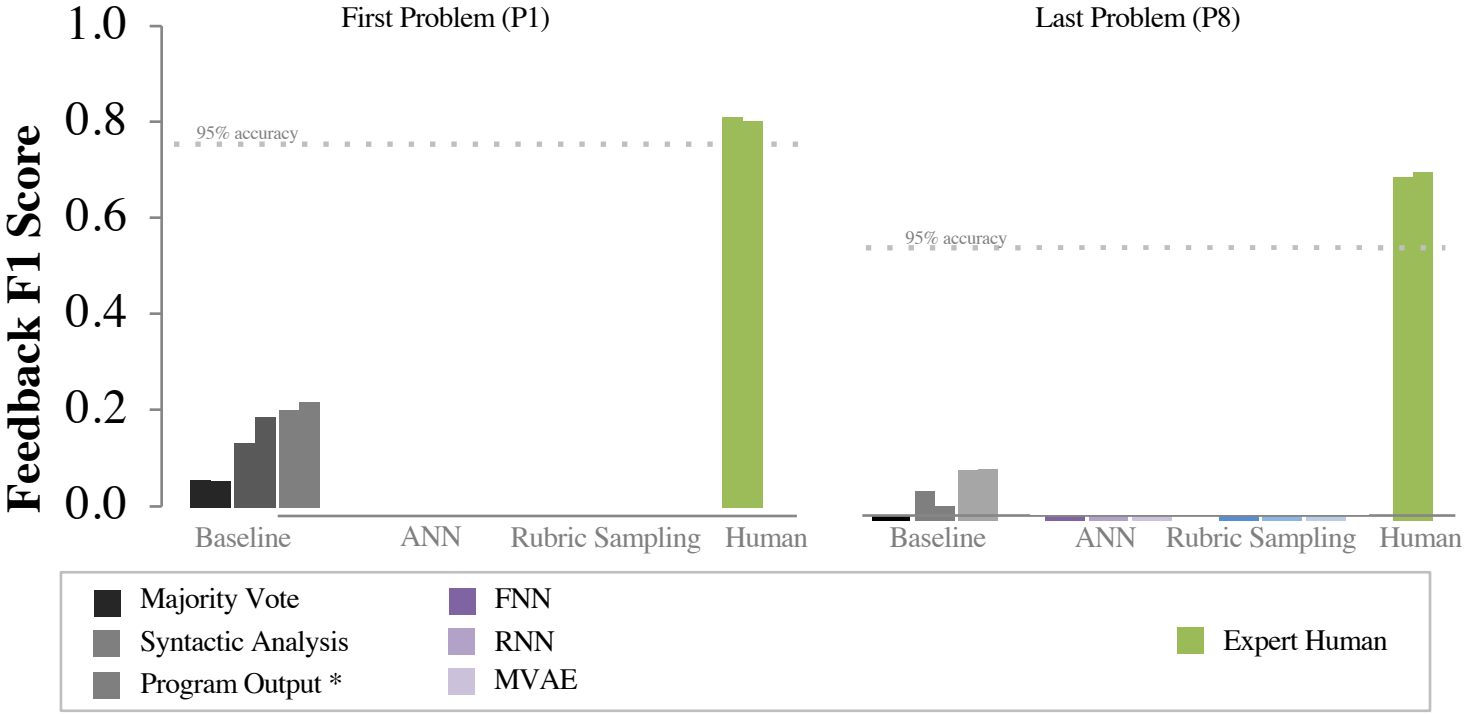
4. Completes attempt

5. Backtracks

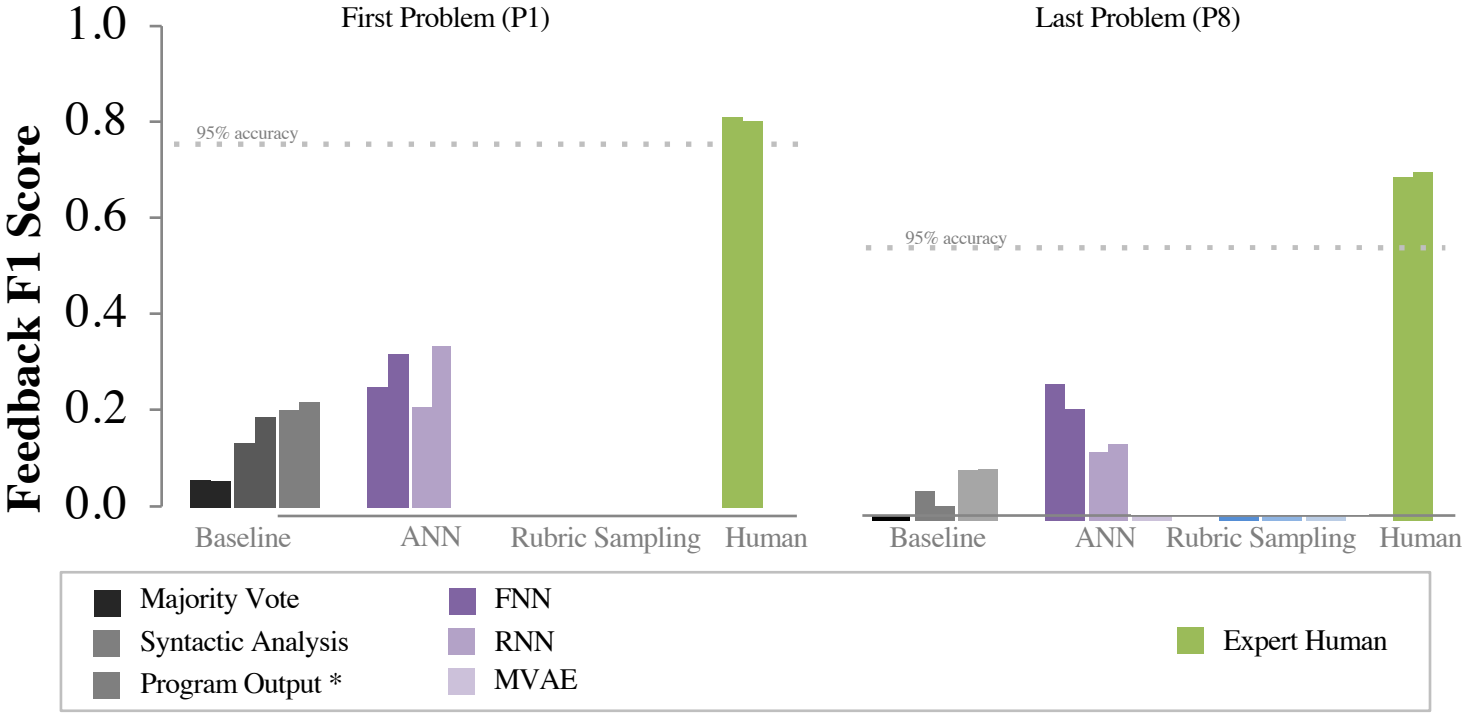
6. Finds solution

Lisa Wang, Chris Piech. Deep Knowledge Tracing on Programming Exercises. L@S 2017

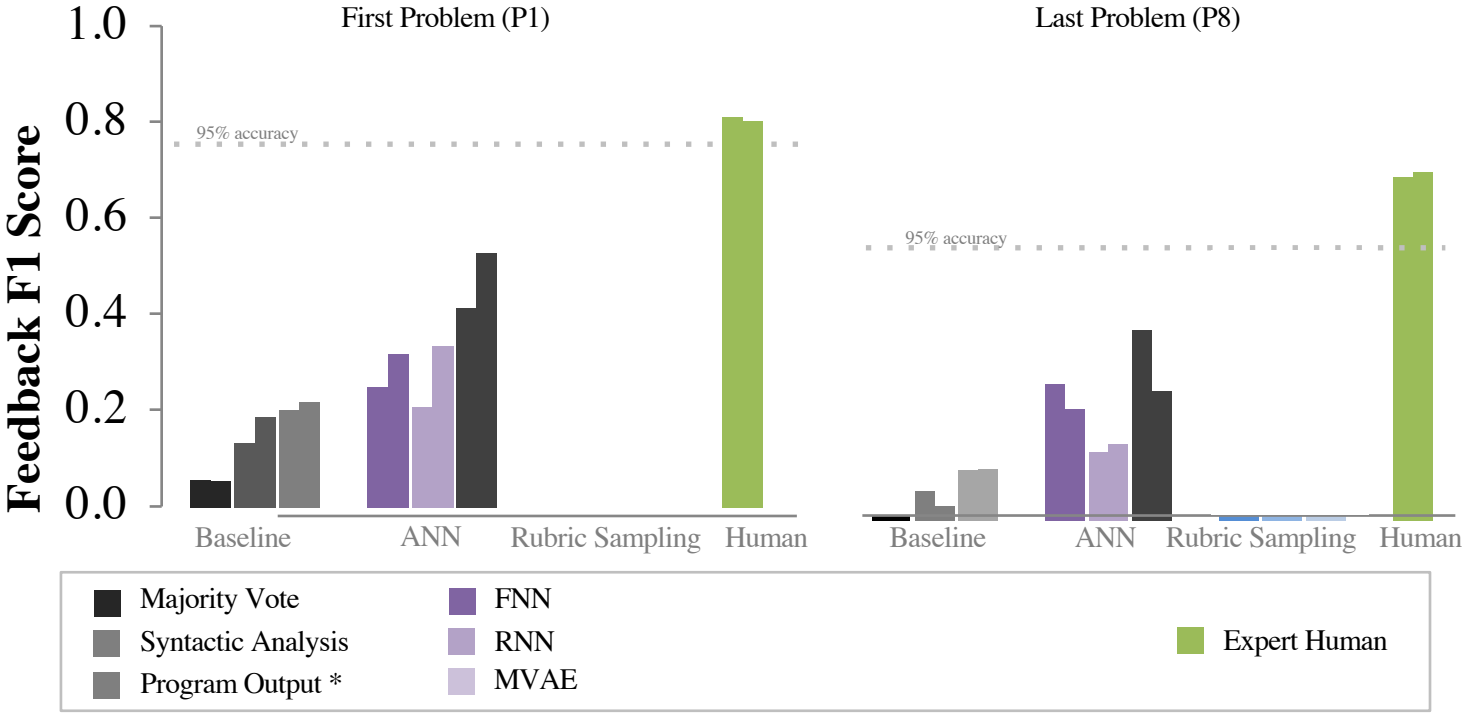
Baselines Don’t Work



Neural Networks Don’t Work



Inaccurate, Uninterpretable, and Data Hungry

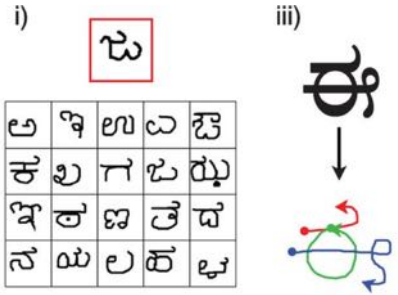


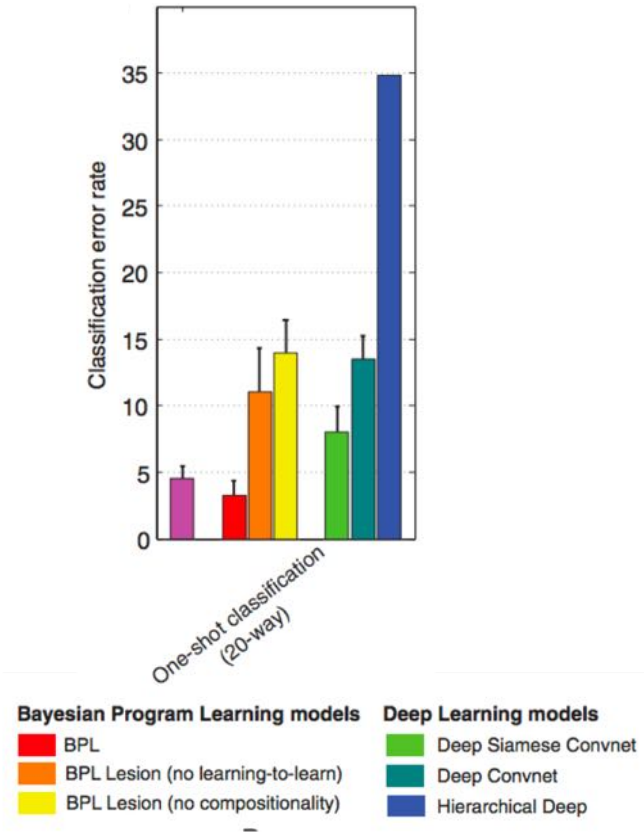
Zero Shot Feedback Challenge

Real impact needs to be zero shot = *zero historical data + zero expert labels*.

[suspense]

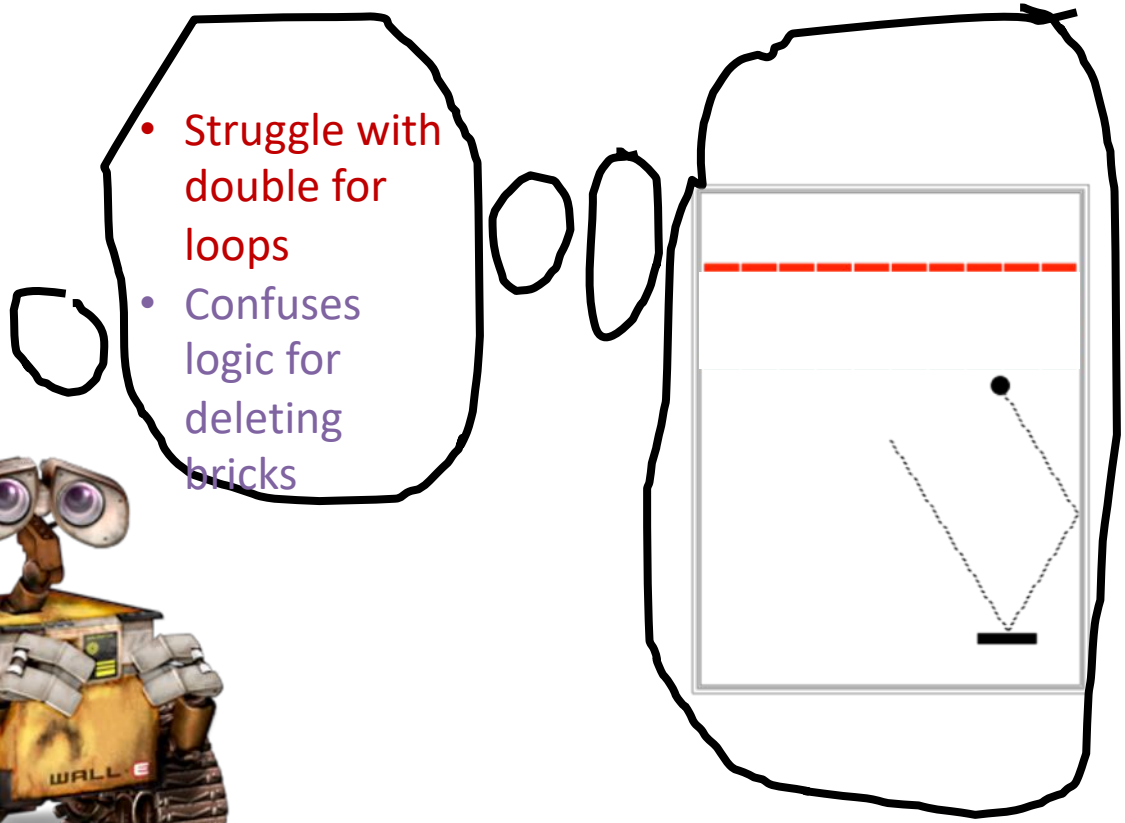
One Shot Learning with Characters





Imagine Students

- Struggle with double for loops
- Confuses logic for deleting bricks



Imagine Students

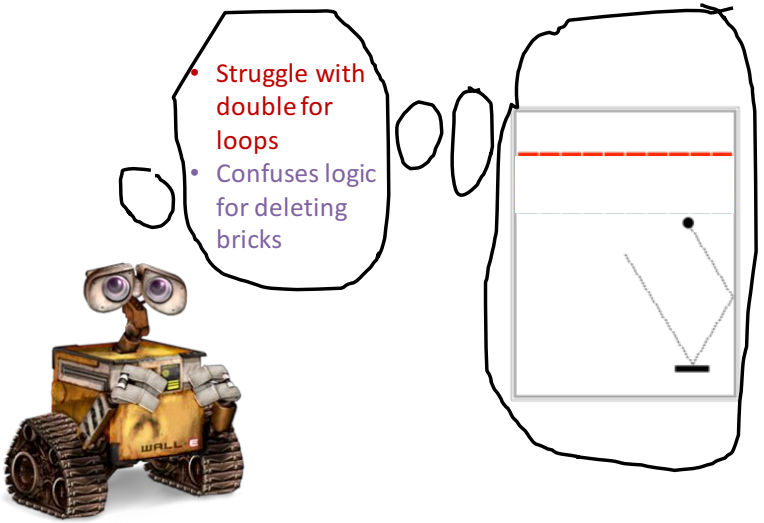
Misconceptions

Priors on knowledge

Program methods

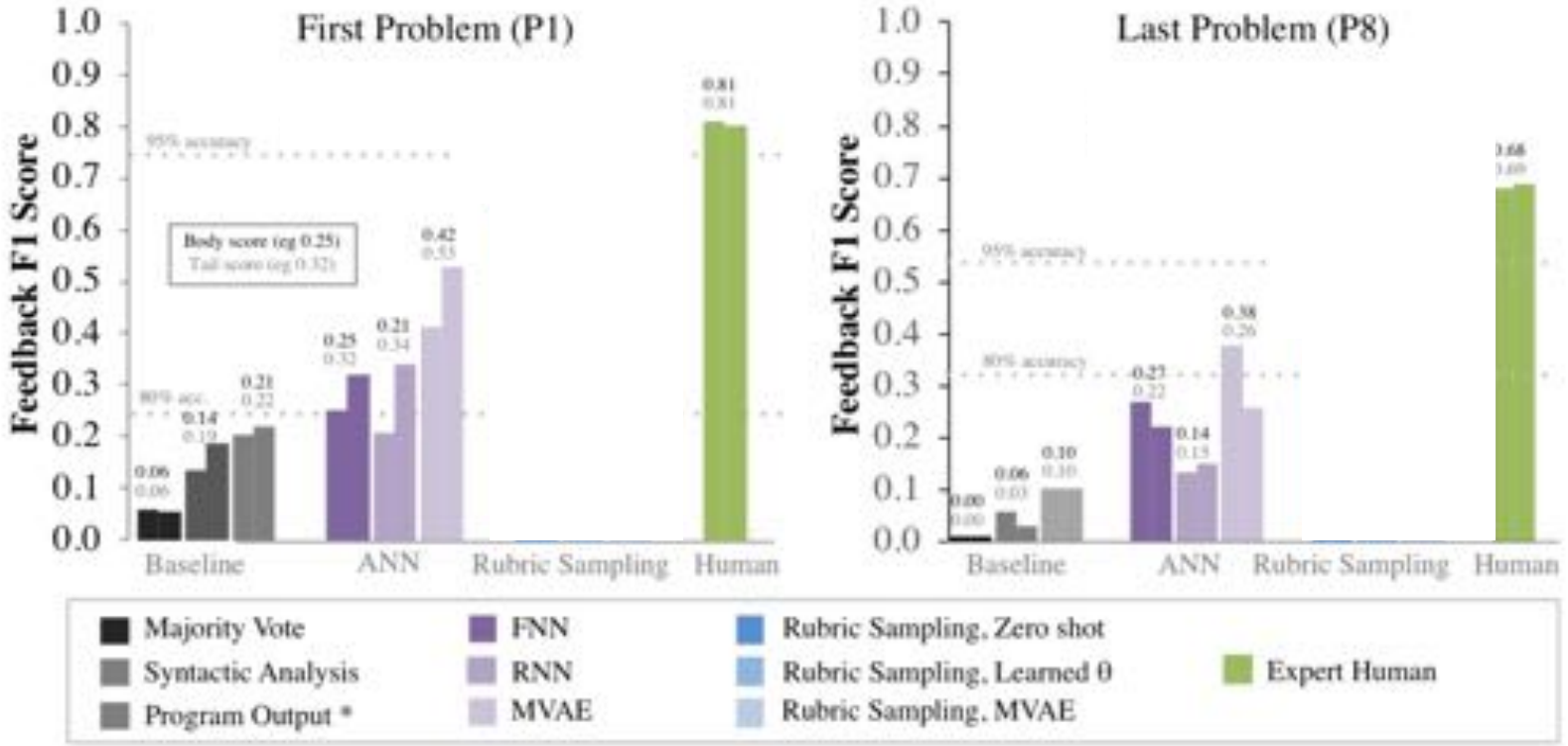
$$P(\psi, \theta^{(1)}, \ldots, \theta^{(M)}, I^{(1)}, \ldots, I^{(M)})$$

$$= P(\psi) \prod_{m=1}^{M} P(I^{(m)} | \theta^{(m)}) P(\theta^{(m)} | \psi) \quad *$$

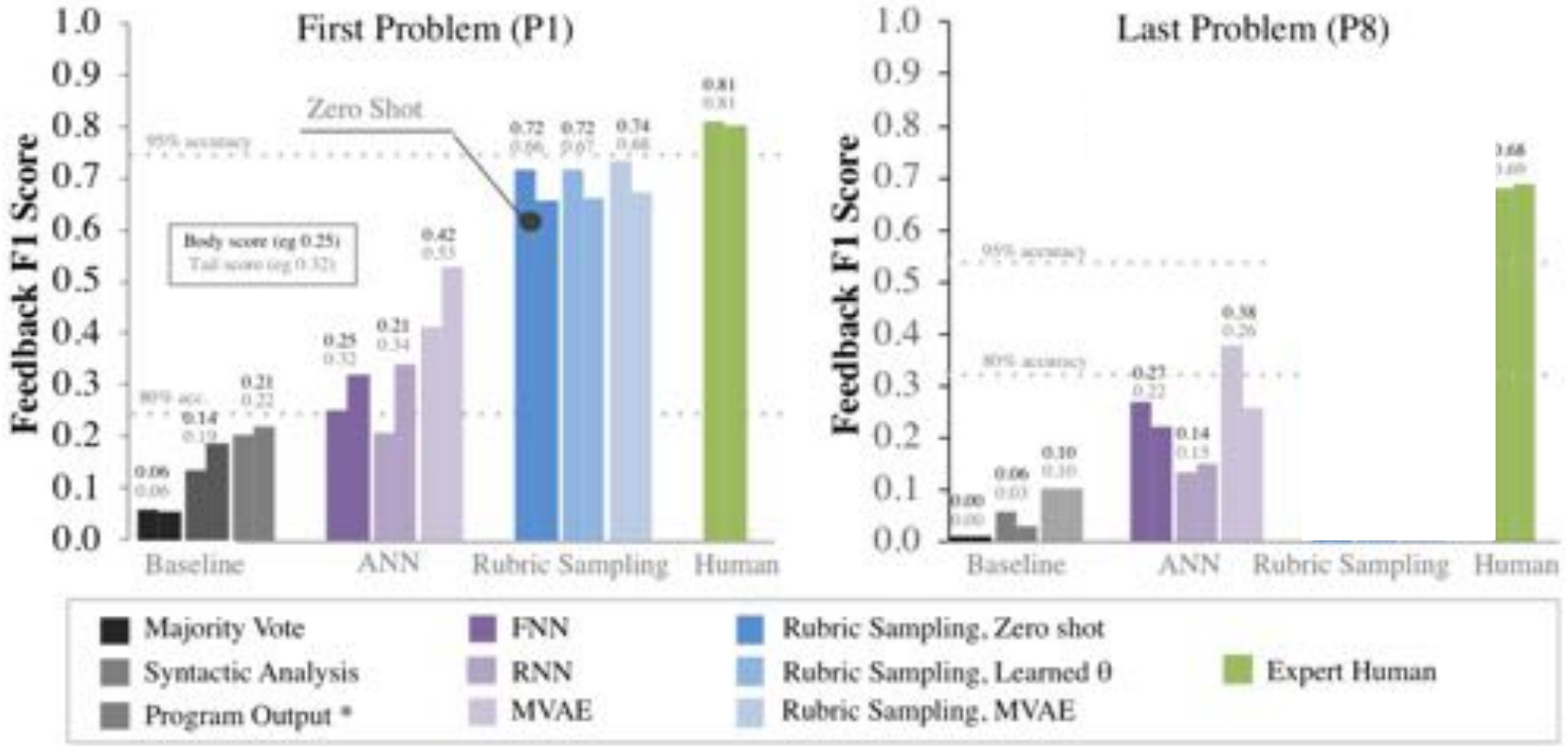


* Encoded as a Bayesian Meta Program

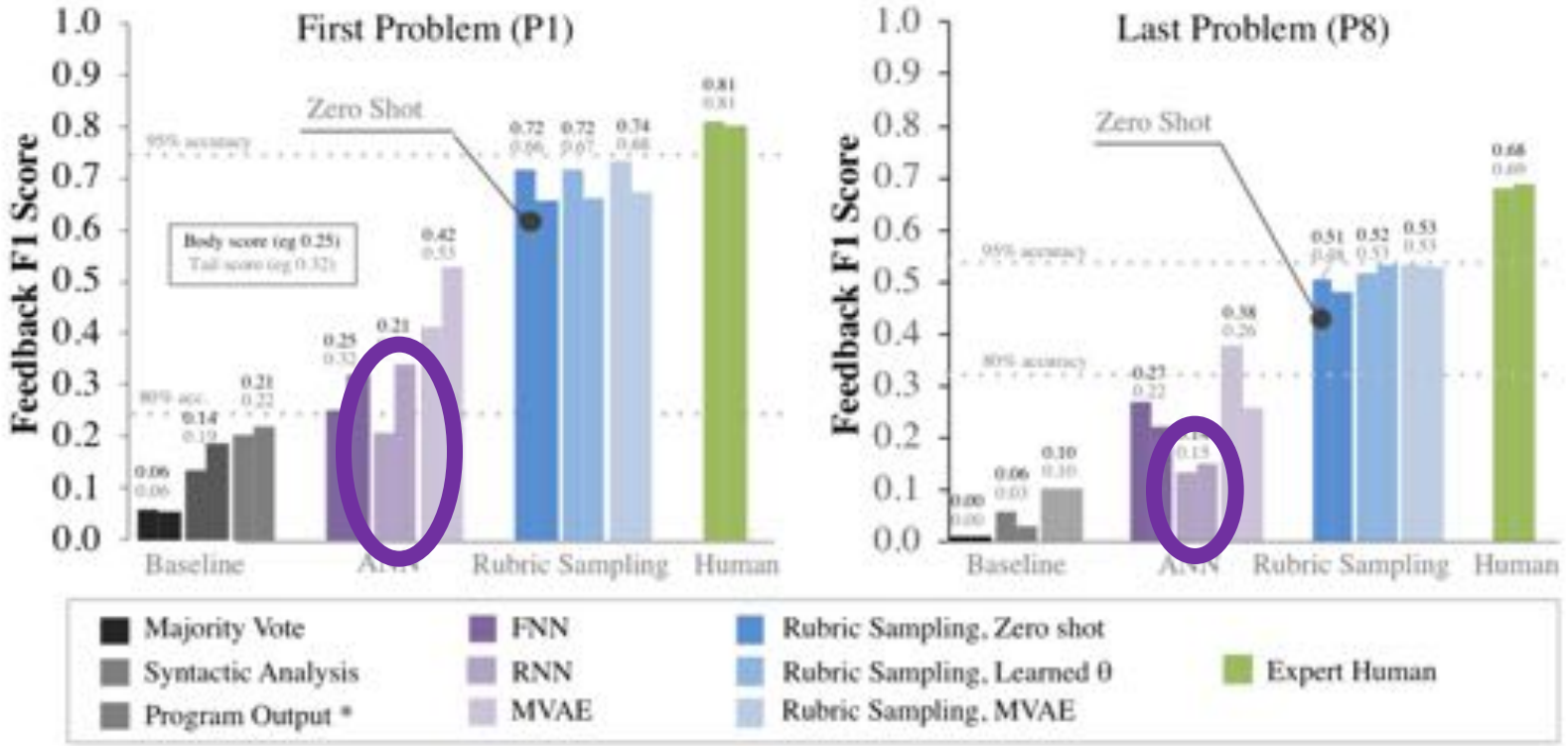
Setting a new state-of-the-art



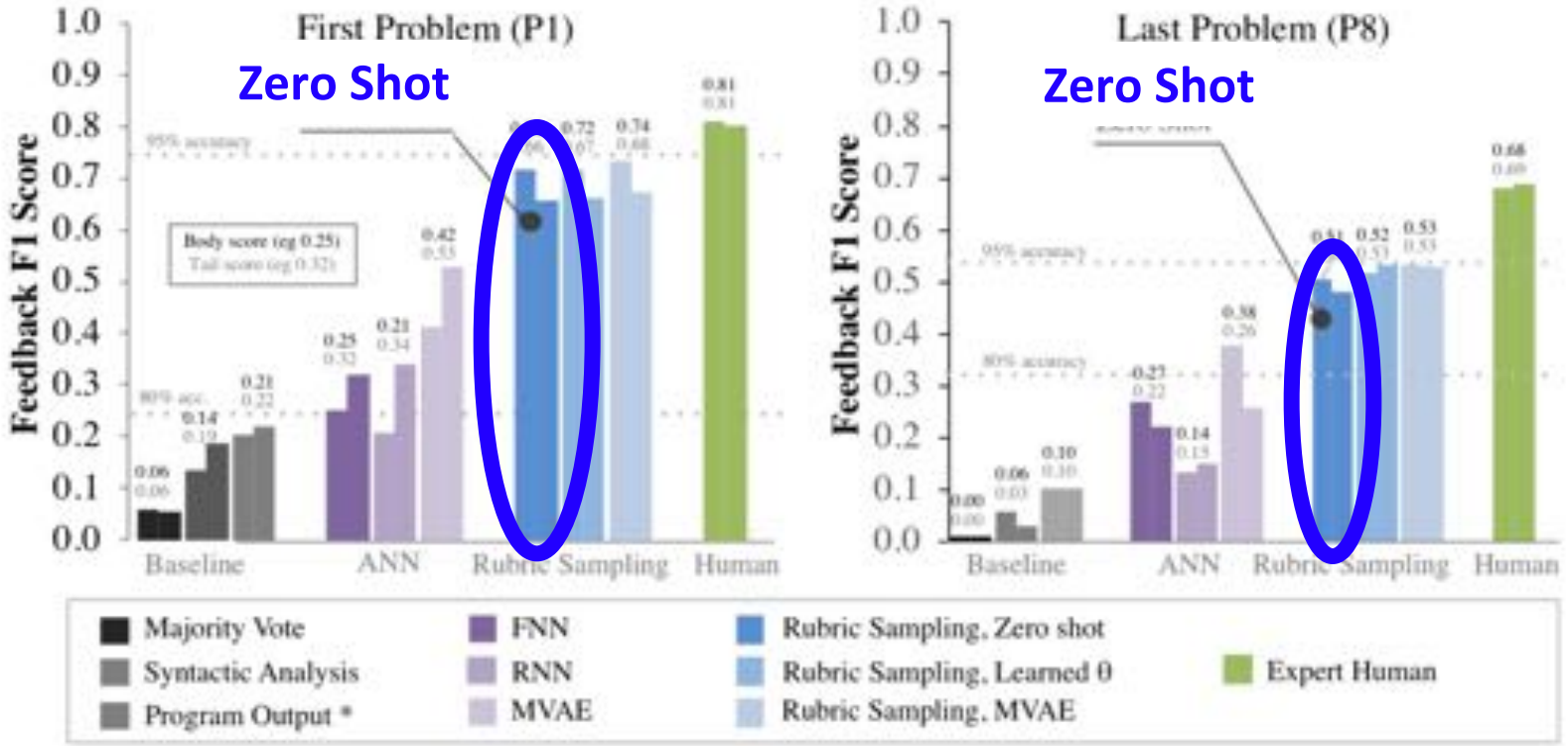
Setting a new state-of-the-art



Setting a new state-of-the-art



Setting a new state-of-the-art



Rubrics are interpretable

Inference is easy in a context free grammar: use A*.

We can highlight parts of the code that caused the model to give a particular feedback.

Replace **compiler messages** with **code highlighting**.

Can we **grade** my CS1 **midterm** more accurately than Stanford TAs?

Chapter 3: Helping Student Teachers

Presented at SIGCSE 2019

Lisa Yan

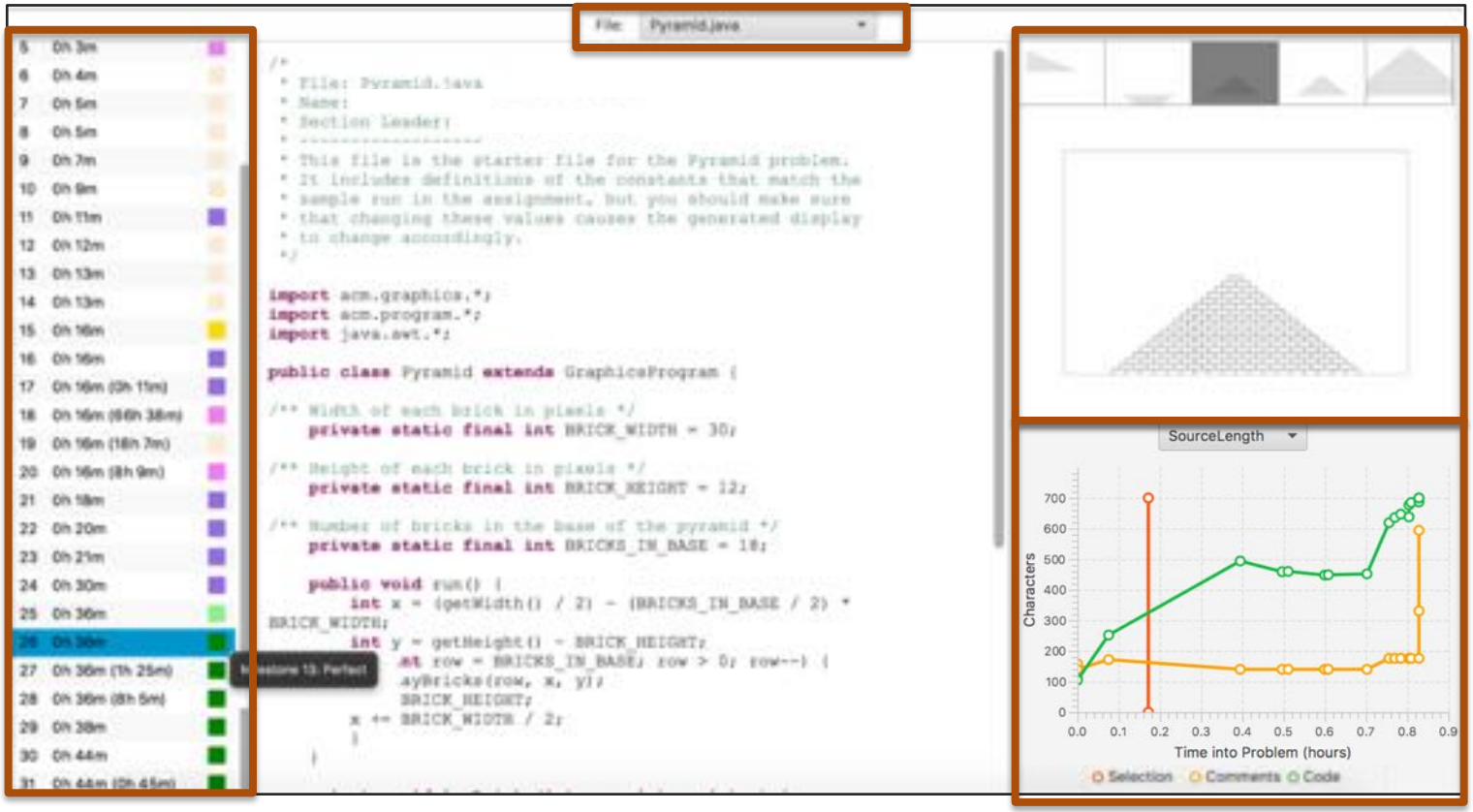
Annie Hu

Students should learn
the *process* of how to solve
programming problems.

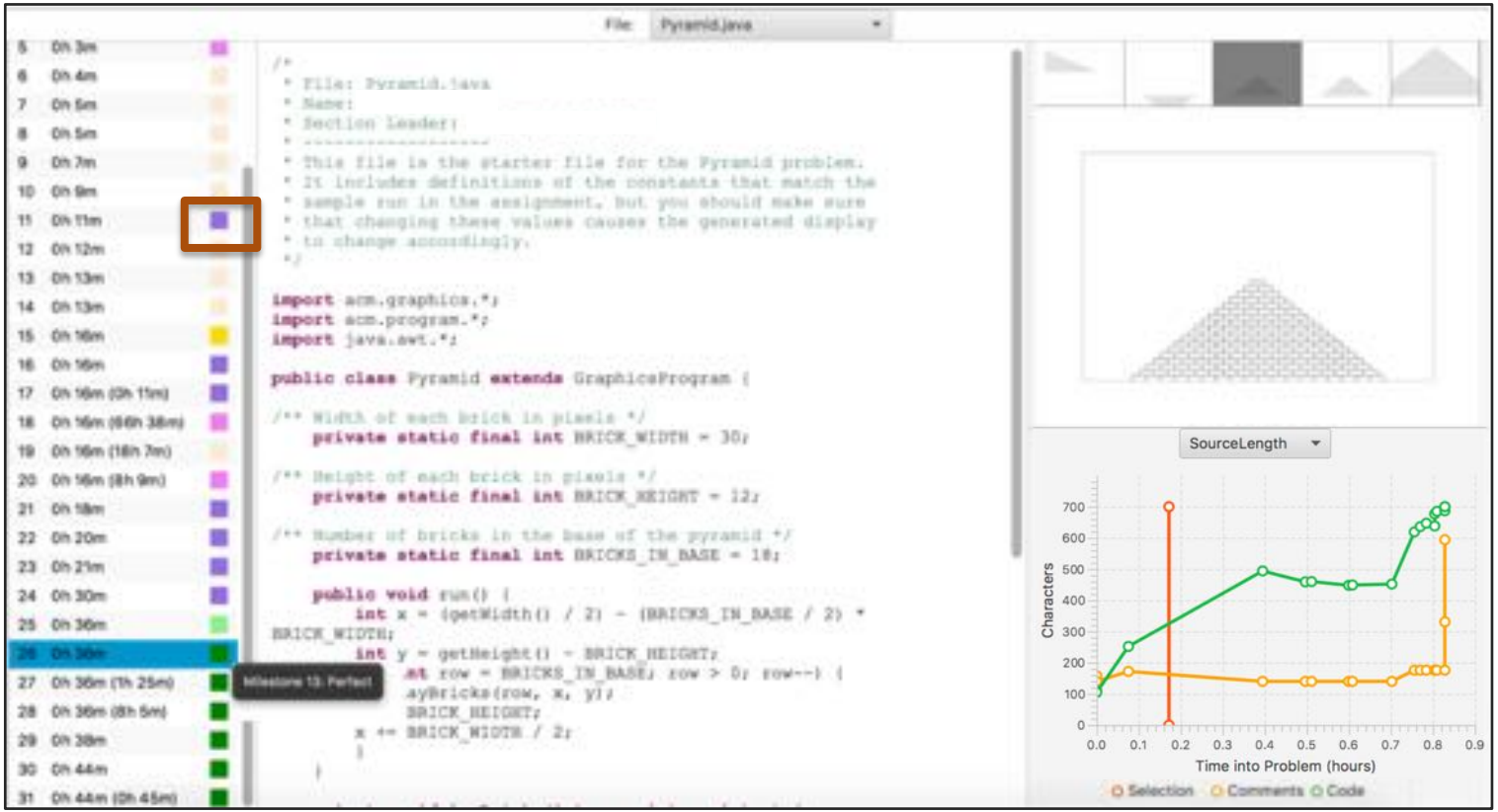
But we aren’t providing feedback on *process*.

(We are providing feedback on final solutions)

Pensieve Tool



Pensieve Tool



Students work faster and learn more