How effective can simple ordinal peer grading be?

Ioannis Caragiannis

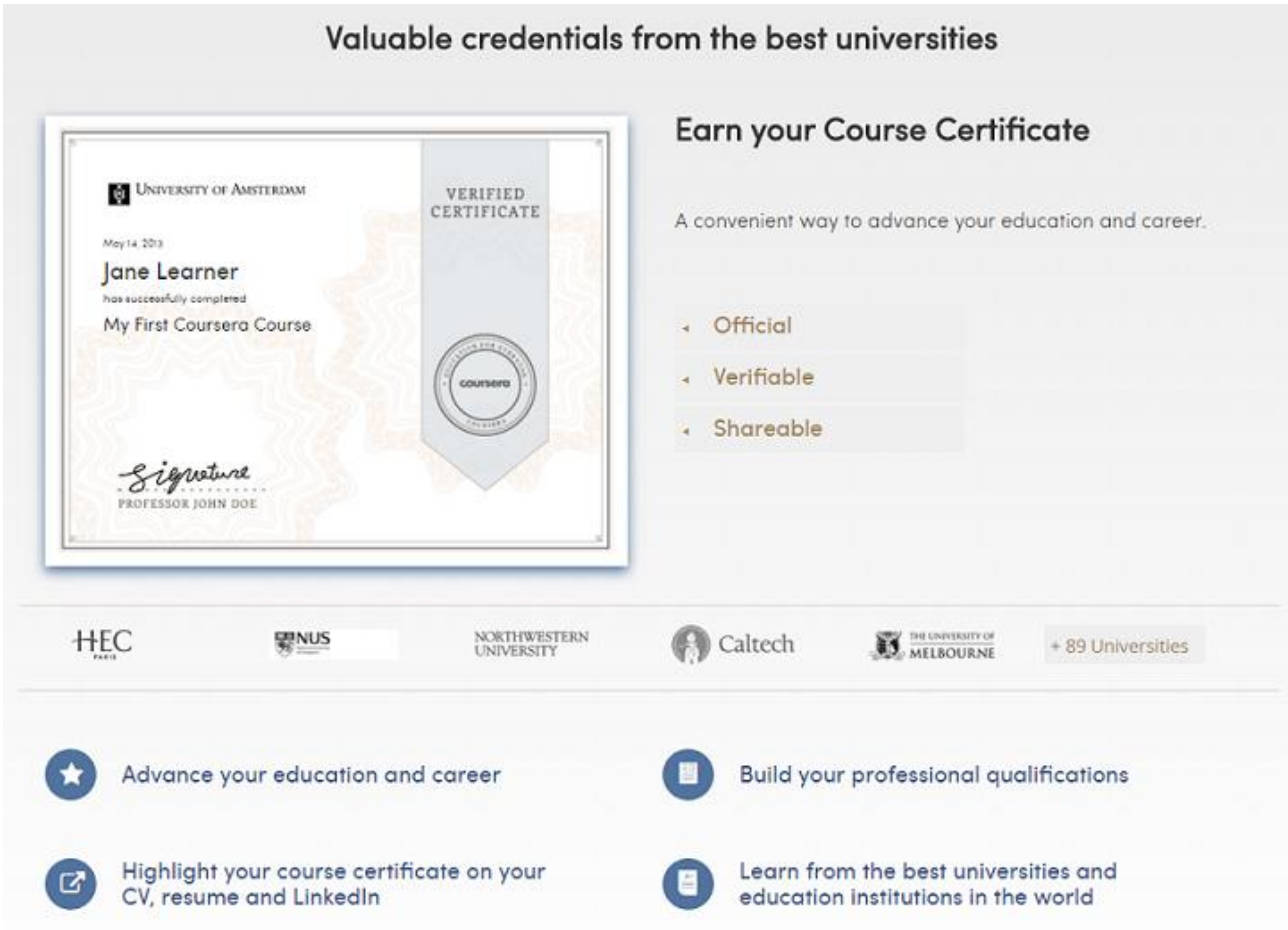
George Krimpas

Alexandros Voudouris

University of Patras

EC 2016

Motivation: verified certificates in MOOCs



The challenge

- The verified certificate should contain **reliable information**
- How can we evaluate the performance of this *huge number* of students in an examination?
- Easy solution: use **closed** type questions (like multiple choice) that can be evaluated automatically
- But … there are courses where students must be examined using **open** type assignments (e.g., solve a math exercise, or write an essay)
- Grading is a typical example of a **human computation** task in such cases
- Limited and costly qualified human resources

Overcoming the problem

Peer assessments

In many courses, the most meaningful assignments cannot be easily graded by a computer. That's why we use peer assessments, where learners can evaluate and provide feedback on each other's work. This technique has been shown in many studies to result in accurate feedback for the learner and a valuable learning experience for the grader.

Peer grading

Each student grades a small number of exam papers submitted by other students

Variations

- **Cardinal peer grading**
  - use numerical scores as grades
  - [Piech et al. 2013, Shah et al. 2013, Walsh 2014]

- **Ordinal peer grading** (this work)
  - simply order the exam papers they are given
  - [Raman and Joachims 2014, Caragiannis et al. 2015]

The setting

students (that participated)
Sheldon
Leonard
Penny
Howard
Raj
Bernadette
Amy

The setting

- Each student gets a bundle of *k* exam papers of other students
- Every exam paper belongs in exactly *k* bundles

students (that participated)
Sheldon
Leonard
Penny
Howard
Raj
Bernadette
Amy

bundles (to be graded)
Penny, Howard, Amy
Sheldon, Bernadette, Amy
Leonard, Raj, Amy
Penny, Raj, Bernadette
Leonard, Howard, Bernadette
Sheldon, Leonard, Penny
Sheldon, Howard, Raj

The setting

- Each student gets a bundle of *k* exam papers of other students
- Every exam paper belongs in exactly *k* bundles

students (that participated)
Sheldon
Leonard
Penny
Howard
Raj
Bernadette
Amy

bundles (to be graded)
Penny, Howard, Amy
Sheldon, Bernadette, Amy
Leonard, Raj, Amy
Penny, Raj, Bernadette
Leonard, Howard, Bernadette
Sheldon, Leonard, Penny
Sheldon, Howard, Raj

No one grades their own paper

The setting

- Each student gets a bundle of *k* exam papers of other students
- Every exam paper belongs in exactly *k* bundles

students (that participated)
Sheldon
Leonard
Penny
Howard
Raj
Bernadette
Amy

bundles (to be graded)
Penny, Howard, Amy
Sheldon, Bernadette, Amy
Leonard, Raj, Amy
Penny, Raj, Bernadette
Leonard, Howard, Bernadette
Sheldon, Leonard, Penny
Sheldon, Howard, Raj

The setting

- Each student gets a bundle of k exam papers of other students
- Every exam paper belongs in exactly k bundles
- Each student orders her bundle and produces a partial ranking

students (that participated)	**bundles** (to be graded)	**partial rankings** (outcome of grading)
Sheldon	Penny, Howard, Amy	Penny $\succ$ Howard $\succ$ Amy
Leonard	Sheldon, Bernadette, Amy	Sheldon $\succ$ Bernadette $\succ$ Amy
Penny	Leonard, Raj, Amy	Leonard $\succ$ Raj $\succ$ Amy
Howard	Penny, Raj, Bernadette	Penny $\succ$ Bernadette $\succ$ Raj
Raj	Leonard, Howard, Bernadette	Leonard $\succ$ Bernadette $\succ$ Howard
Bernadette	Sheldon, Leonard, Penny	Penny $\succ$ Sheldon $\succ$ Leonard
Amy	Sheldon, Howard, Raj	Sheldon $\succ$ Howard $\succ$ Raj

The setting

- Each student gets a bundle of *k* exam papers of other students
- Every exam paper belongs in exactly *k* bundles
- Each student orders her bundle and produces a partial ranking

students (that participated)	**bundles** (to be graded)	**partial rankings** (outcome of grading)
Sheldon	Penny, Howard, Amy	Penny $\succ$ Howard $\succ$ Amy
Leonard	Sheldon, Bernadette, Amy	Sheldon $\succ$ Bernadette $\succ$ Amy
Penny	Leonard, Raj, Amy	Leonard $\succ$ Raj $\succ$ Amy
Howard	Penny, Raj, Bernadette	Penny $\succ$ Bernadette $\succ$ Raj
Raj	Leonard, Howard, Bernadette	Leonard $\succ$ Bernadette $\succ$ Howard
Bernadette	Sheldon, Leonard, Penny	Penny $\succ$ Sheldon $\succ$ Leonard
Amy	Sheldon, Howard, Raj	Sheldon $\succ$ Howard $\succ$ Raj

Goals

- **Question:** How can we aggregate the partial rankings into a ranking of all students?
- **Simplicity:** Simple aggregation methods, like scoring rules (e.g. Borda)
- **Efficiency:**
  - Assume that there is an underline true ranking (ground truth)
  - How close is the final ranking to the ground truth?
  - Measure: *fraction of correctly recovered pairwise relations*

Type-ordering aggregation rules

- Every exam paper has a **type**
  - vector of ranks it gets in the partial rankings it appears in

Type-ordering aggregation rules

- Every exam paper has a **type**
  - vector of ranks it gets in the partial rankings it appears in

students	**partial rankings**	**type**
Sheldon	Penny $\succ$ Howard $\succ$ Amy	
Leonard	Sheldon $\succ$ Bernadette $\succ$ Amy	
Penny	Leonard $\succ$ Raj $\succ$ Amy	
Howard	Penny $\succ$ Bernadette $\succ$ Raj	
Raj	Leonard $\succ$ Bernadette $\succ$ Howard	
Bernadette	Penny $\succ$ Sheldon $\succ$ Leonard	
Amy	Sheldon $\succ$ Howard $\succ$ Raj	

Type-ordering aggregation rules

- Every exam paper has a **type**
  - vector of ranks it gets in the partial rankings it appears in

students	**partial rankings**	**type**
Sheldon	Penny $\succ$ Howard $\succ$ Amy	
Leonard	Sheldon $\succ$ Bernadette $\succ$ Amy	
Penny	Leonard $\succ$ Raj $\succ$ Amy	
Howard	Penny $\succ$ Bernadette $\succ$ Raj	
Raj	Leonard $\succ$ Bernadette $\succ$ Howard	
Bernadette	Penny $\succ$ Sheldon $\succ$ Leonard	
Amy	Sheldon $\succ$ Howard $\succ$ Raj	

Type-ordering aggregation rules

- Every exam paper has a **type**
  - vector of ranks it gets in the partial rankings it appears in

students	partial rankings	type
Sheldon	Penny ≻ Howard ≻ Amy	(1, 1, 2)
Leonard	Sheldon ≻ Bernadette ≻ Amy	
Penny	Leonard ≻ Raj ≻ Amy	
Howard	Penny ≻ Bernadette ≻ Raj	
Raj	Leonard ≻ Bernadette ≻ Howard	
Bernadette	Penny ≻ Sheldon ≻ Leonard	
Amy	Sheldon ≻ Howard ≻ Raj	

Type-ordering aggregation rules

- Every exam paper has a **type**
  - vector of ranks it gets in the partial rankings it appears in

students	partial rankings	type
Sheldon	Penny $\succ$ Howard $\succ$ Amy	(1, 1, 2)
Leonard	Sheldon $\succ$ Bernadette $\succ$ Amy	(1, 1, 3)
Penny	Leonard $\succ$ Raj $\succ$ Amy	(1, 1, 1)
Howard	Penny $\succ$ Bernadette $\succ$ Raj	(2, 2, 3)
Raj	Leonard $\succ$ Bernadette $\succ$ Howard	(2, 3, 3)
Bernadette	Penny $\succ$ Sheldon $\succ$ Leonard	(2, 2, 2)
Amy	Sheldon $\succ$ Howard $\succ$ Raj	(3, 3, 3)

Type-ordering aggregation rules

- Consider an ordering of all possible types

Type-ordering aggregation rules

- Consider an ordering of all possible types

types
(1, 1, 1)
(1, 1, 2)
(1, 1, 3)
(1, 2, 2)
(1, 2, 3)
(2, 2, 2)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

Type-ordering aggregation rules

- Consider an ordering of all possible types

ordering
(1, 1, 1)
(1, 1, 2)
(1, 2, 2)
(1, 1, 3)
(2, 2, 2)
(1, 2, 3)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

Type-ordering aggregation rules

- Consider an ordering of all possible types

ordering
(1, 1, 1)
(1, 1, 2)
(1, 2, 2)
(1, 1, 3)
(2, 2, 2)
(1, 2, 3)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

Borda

- In each partial ranking, the first exam paper gets *k* points, the second get *k*-1 points, and so on
- The exam papers are sorted in descending order w.r.t. their total points

Type-ordering aggregation rules

- Consider an ordering of all possible types
- Order the exam papers according to that type-ordering

ordering
(1, 1, 1)
(1, 1, 2)
(1, 2, 2)
(1, 1, 3)
(2, 2, 2)
(1, 2, 3)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

students
Sheldon
Leonard
Penny
Howard
Raj
Bernadette
Amy

type
(1, 1, 2)
(1, 1, 3)
(1, 1, 1)
(2, 2, 3)
(2, 3, 3)
(2, 2, 2)
(3, 3, 3)

ranking	
1st	
2nd	
3rd	
4th	
5th	
6th	
7th	

Type-ordering aggregation rules

- Consider an ordering of all possible types
- Order the exam papers according to that type-ordering

ordering
$\boxed{(1, 1, 1)}$
(1, 1, 2)
(1, 2, 2)
(1, 1, 3)
(2, 2, 2)
(1, 2, 3)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

students
Sheldon
Leonard
Penny
Howard
Raj
Bernadette
Amy

type
(1, 1, 2)
(1, 1, 3)
$\boxed{(1, 1, 1)}$
(2, 2, 3)
(2, 3, 3)
(2, 2, 2)
(3, 3, 3)

ranking	
1st	
2nd	
3rd	
4th	
5th	
6th	
7th	

Type-ordering aggregation rules

- Consider an ordering of all possible types
- Order the exam papers according to that type-ordering

ordering
(1, 1, 1)
(1, 1, 2)
(1, 2, 2)
(1, 1, 3)
(2, 2, 2)
(1, 2, 3)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

students
Sheldon
Leonard
Penny
Howard
Raj
Bernadette
Amy

type
(1, 1, 2)
(1, 1, 3)
(1, 1, 1)
(2, 2, 3)
(2, 3, 3)
(2, 2, 2)
(3, 3, 3)

ranking	
1st	Penny
2nd	
3rd	
4th	
5th	
6th	
7th	

Type-ordering aggregation rules

- Consider an ordering of all possible types
- Order the exam papers according to that type-ordering

ordering
(1, 1, 1)
(1, 1, 2)
(1, 2, 2)
(1, 1, 3)
(2, 2, 2)
(1, 2, 3)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

students	type
Sheldon	(1, 1, 2)
Leonard	(1, 1, 3)
Penny	(1, 1, 1)
Howard	(2, 2, 3)
Raj	(2, 3, 3)
Bernadette	(2, 2, 2)
Amy	(3, 3, 3)

ranking	
1st	Penny
2nd	Sheldon
3rd	Leonard
4th	Bernadette
5th	Howard
6th	Raj
7th	Amy

Type-ordering aggregation rules

- Consider an ordering of all possible types
- Order the exam papers according to that type-ordering

ordering
(1, 1, 1)
(1, 1, 2)
(1, 2, 2)
(1, 1, 3)
(2, 2, 2)
(1, 2, 3)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

students	type
Sheldon	(1, 1, 2)
Leonard	(1, 1, 3)
Penny	(1, 1, 1)
Howard	(2, 2, 3)
Raj	(2, 3, 3)
Bernadette	(2, 2, 2)
Amy	(3, 3, 3)

ranking	
1st	Penny
2nd	Sheldon
3rd	Leonard
4th	Bernadette
5th	Howard
6th	Raj
7th	Amy

ground truth	
1st	Sheldon
2nd	Penny
3rd	Leonard
4th	Howard
5th	Raj
6th	Bernadette
7th	Amy

Type-ordering aggregation rules

- Consider an ordering of all possible types
- Order the exam papers according to that type-ordering

ordering
(1, 1, 1)
(1, 1, 2)
(1, 2, 2)
(1, 1, 3)
(2, 2, 2)
(1, 2, 3)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

students
Sheldon
Leonard
Penny
Howard
Raj
Bernadette
Amy

type
(1, 1, 2)
(1, 1, 3)
(1, 1, 1)
(2, 2, 3)
(2, 3, 3)
(2, 2, 2)
(3, 3, 3)

ranking	
1st	Penny
2nd	Sheldon
3rd	Leonard
4th	Bernadette
5th	Howard
6th	Raj
7th	Amy

ground truth	
1st	Sheldon
2nd	Penny
3rd	Leonard
4th	Howard
5th	Raj
6th	Bernadette
7th	Amy

NOT correctly recovered

correctly recovered

Type-ordering aggregation rules

- Consider an ordering of all possible types
- Order the exam papers according to that type-ordering

ordering
(1, 1, 1)
(1, 1, 2)
(1, 2, 2)
(1, 1, 3)
(2, 2, 2)
(1, 2, 3)
(2, 2, 3)
(2, 3, 3)
(3, 3, 3)

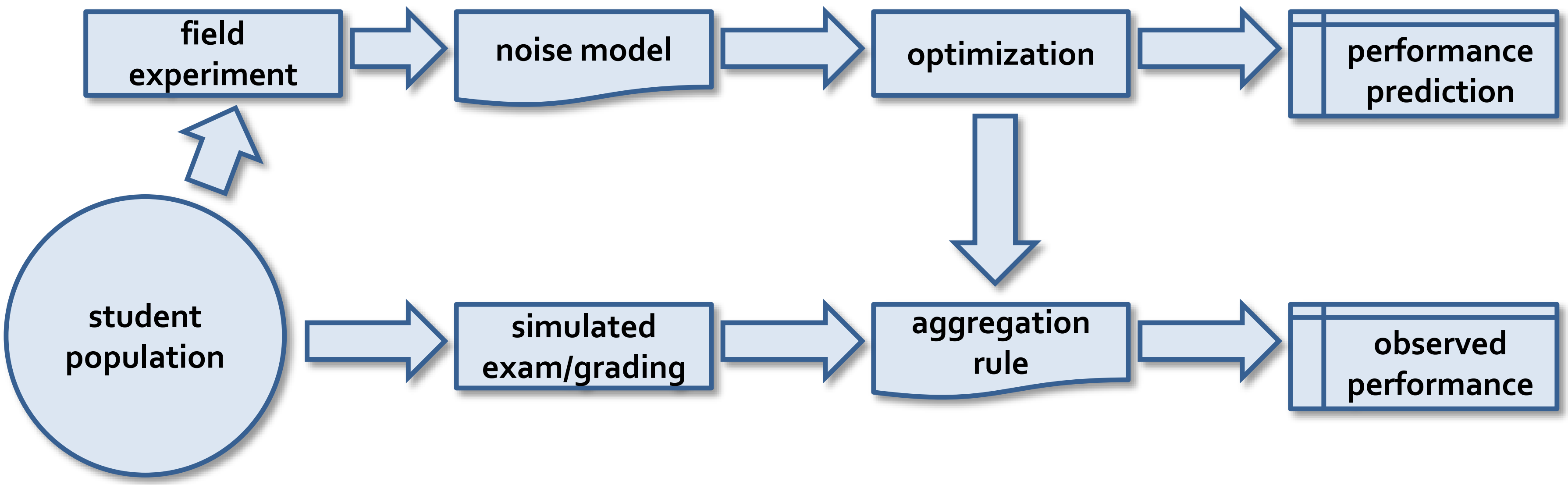
students	type
Sheldon	(1, 1, 2)
Leonard	(1, 1, 3)
Penny	(1, 1, 1)
Howard	(2, 2, 3)
Raj	(2, 3, 3)
Bernadette	(2, 2, 2)
Amy	(3, 3, 3)

ranking	
1st	Penny
2nd	Sheldon
3rd	Leonard
4th	Bernadette
5th	Howard
6th	Raj
7th	Amy

ground truth	
1st	Sheldon
2nd	Penny
3rd	Leonard
4th	Howard
5th	Raj
6th	Bernadette
7th	Amy

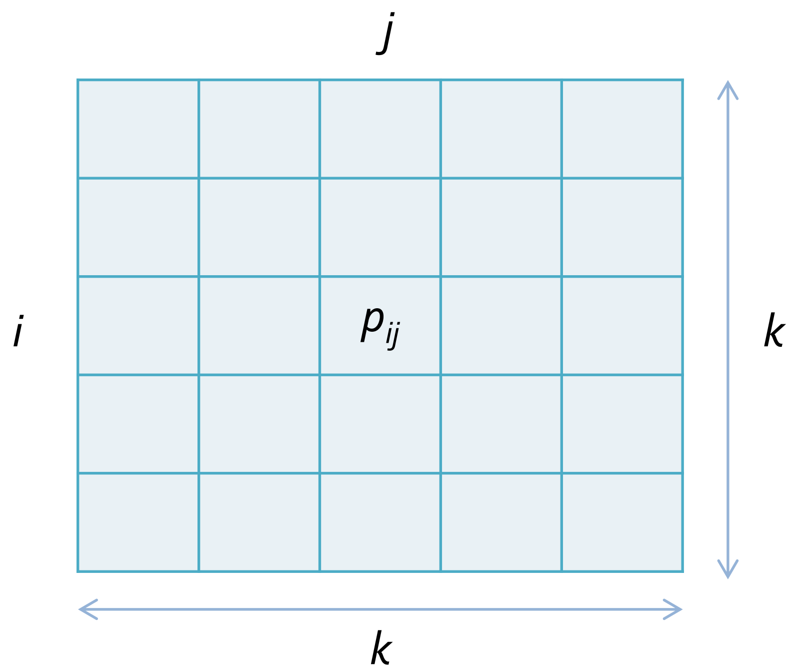
efficiency = 16/21 = 76.19%

Overview of our approach



Grading behavior (noise model)

- **Noise (stochastic) matrix** $P = (p_{ij})_{i,\, j = 1 \ldots k}$



- p_{ij} = Pr[student ranks an exam paper at position i | the correct position of this exam paper in the bundle is j]

A noise model built from real data

0.463	0.257	0.102	0.058	0.058	0.058
0.205	0.316	0.227	0.110	0.066	0.073
0.161	0.191	0.257	0.205	0.132	0.051
0.102	0.117	0.191	0.242	0.279	0.066
0.044	0.066	0.139	0.220	0.301	0.227
0.022	0.051	0.080	0.161	0.161	0.522

Realistic model

- Field experiment at our university
- Data collected from 136 undergraduate students
- Each student ordinal graded a bundle of k=6 exam papers

Assessing the quality of type-ordering aggregation rules

- Assumption: number of students tends to infinity
  - The ranks in the ground truth are real numbers in [0,1]

$$
\begin{aligned}
C(\succ) &= \int_0^1 \int_x^1 \left(\sum_{\sigma,\sigma' : \sigma \succ \sigma'} \Pr[x \rhd \sigma \text{ and } y \rhd \sigma'] \right) \mathrm{d}y \, \mathrm{d}x \\
&\approx \sum_{\sigma,\sigma' : \sigma \succ \sigma'} \int_0^1 \int_x^1 \Pr[x \rhd \sigma] \cdot \Pr[y \rhd \sigma'] \, \mathrm{d}y \, \mathrm{d}x \\
&= \sum_{\sigma,\sigma' : \sigma \succ \sigma'} W(\sigma, \sigma')
\end{aligned}
$$

Assessing the quality of type-ordering aggregation rules

- Assumption: number of students tends to infinity
  - The ranks in the ground truth are real numbers in [0,1]

$$C(\succ) = \int_0^1 \int_x^1 \left(\sum_{\sigma,\sigma':\sigma\succ\sigma'} \Pr[x \rhd \sigma \text{ and } y \rhd \sigma'] \right) \mathrm{d}y\, \mathrm{d}x$$

$$\approx \sum_{\sigma,\sigma':\sigma\succ\sigma'} \int_0^1 \int_x^1 \underbrace{\Pr[x \rhd \sigma] \cdot \Pr[y \rhd \sigma']}_{\textbf{Neglect dependencies}} \mathrm{d}y\, \mathrm{d}x$$

$$= \sum_{\sigma,\sigma':\sigma\succ\sigma'} W(\sigma,\sigma')$$

Optimization

- **Optimization problem:** compute an ordering of the types such that the sum of weights is maximized

Optimization

- **Optimization problem:** compute an ordering of the types such that the sum of weights is maximized

  ⇒ **FEEDBACK ARC SET** (NP-hard) …

- Turns out to be fairly easy in practical scenarios (k=6)

A theoretical result

Theorem

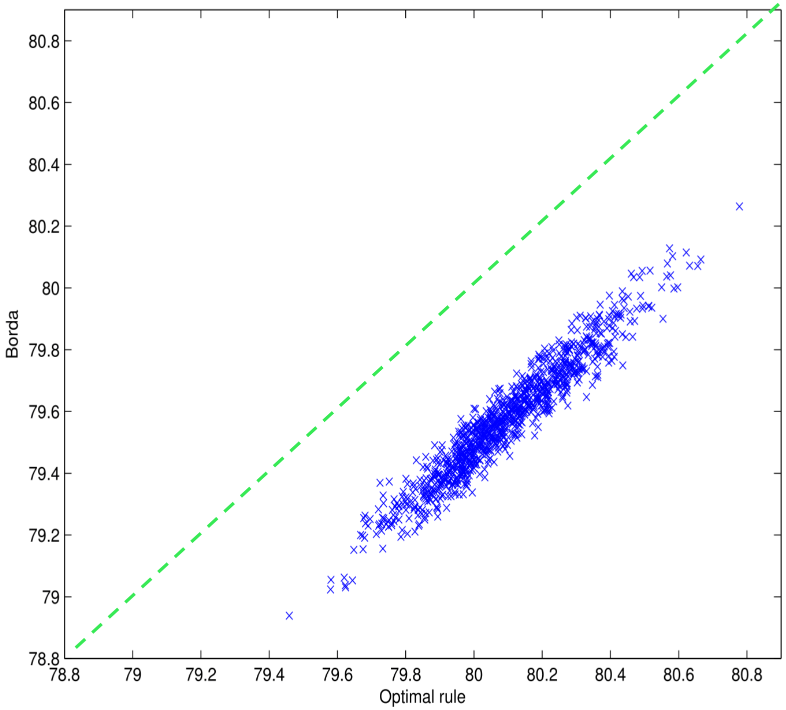
In the perfect grading model, Borda (with any tie-breaking rule) is the optimal type-ordering aggregation rule

Predicted *vs.* observed efficiency

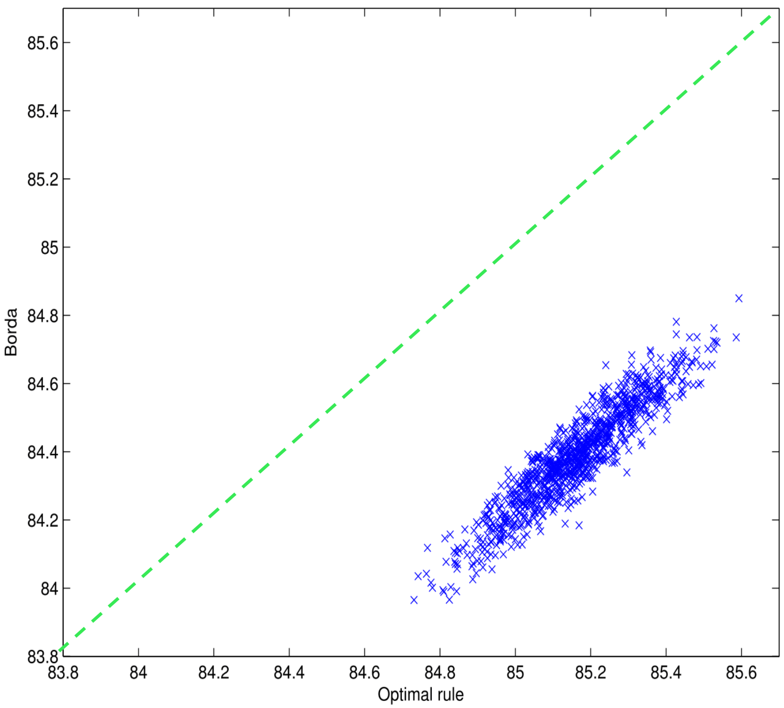
	perfect	realistic		mallows	
	Borda (optimal)	optimal	Borda	optimal	Borda
predicted	92.01 %	80.01 %	79.57 %	85.15 %	84.38 %
observed	92.02 %	80.09 %	79.57 %	85.16 %	84.39 %

- The observed efficiency (almost) coincides with the expected theoretical one
- Borda is always close to optimal … but *not* optimal!

Borda *vs.* optimal



Realistic

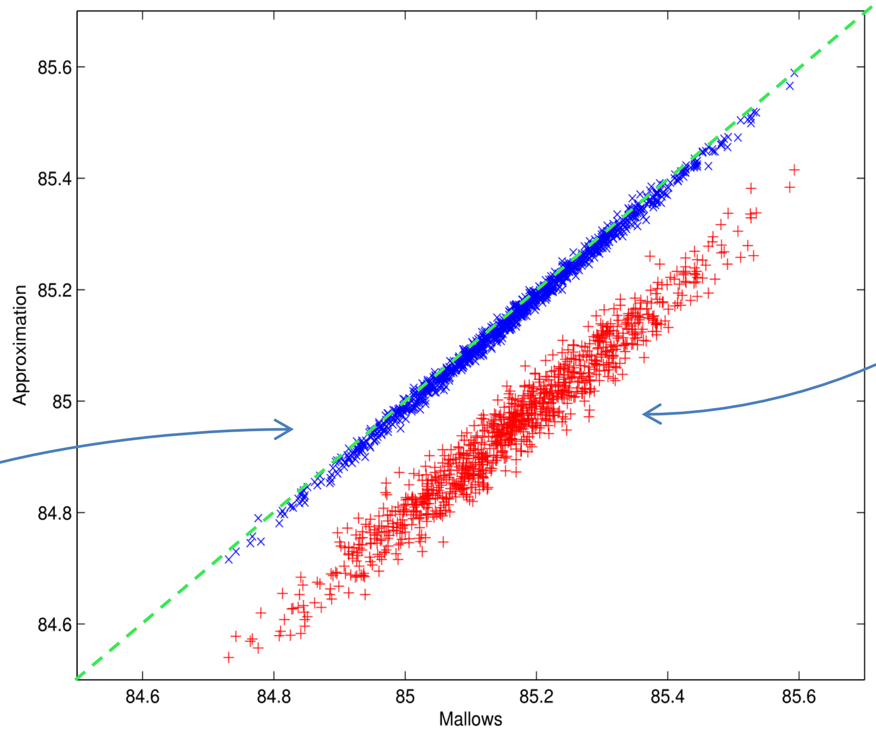


Mallows

- Each point corresponds to a simulated exam with 10000 students

Are 136 samples enough?

- Test for Mallows (for which we know the actual model)



100-sample approximation

1000-sample approximation

Future work

- More real-world field experiments to produce realistic models
  - The one we performed was without training
  - What if we trained the students to ordinal grading first?
  - Try different values of *k*?

- Real-world ordinal peer grading experiments
  - http://co-rank.ceid.upatras.gr/

- Game theoretic or adversarial extensions


Thanks!