New Scalable Methods for Ranking Students in the Presence of Misleading Questions

Gerdus Benadè
Carnegie Mellon University
jbenade@andrew.cmu.edu

Wolfgang Gatterbauer
Northeastern University
wolfgang@ccis.neu.edu

Nam Ho-Nguyen
Carnegie Mellon University
hnh@andrew.cmu.edu

R. Ravi
Carnegie Mellon University
ravi@andrew.cmu.edu

ABSTRACT

We study the problem of ranking students by their abilities and questions by their qualities, solely based on responses to student-sourced multiple-choice questions. This approach addresses the crucial problem of scaling automatic assessment of students to very large class sizes. Existing methods from learning science and Item Response Theory (IRT) have two problems: (1) they assume student responses obey a parameterized model and (2) they were designed for situations with trusted questions and hence suffer from higher inaccuracies in the presence of student-sourced questions that may be misleading. Furthermore, we observe empirically that the running time is quadratic in the number of students. In this paper, we define an axiomatic framework for the requirements of good ranking algorithms, as well as a new model for simulating ill-posed questions. We marry ideas from work in truth discovery with properties from IRT to devise several new algorithms with strong axiomatic guarantees and with linear scalability. We prove that methods for ranking students are not affected by ill-posed questions as long as they obey two natural properties of "translation invariance" and "anti-symmetry"; we then prove that our new methods obey these properties. Our resulting methods solve both prior problems: (1) they scale linearly with the number of students and questions, and (2) they rank students well in the presence of misleading questions.

ACM Reference Format:
Gerdus Benadè, Wolfgang Gatterbauer, Nam Ho-Nguyen, and R. Ravi. 2018. New Scalable Methods for Ranking Students in the Presence of Misleading Questions. In *Proceedings of ACM SIGMOD Conference (SIGMOD'18),* Phillip Bernstein (Ed.). ACM, New York, NY, USA, Article 4, 16 pages. https://doi.org/11.111/111_1

1 INTRODUCTION

In the last decade, technology has had a profound impact on the way students learn. Massive Open Online Courses (MOOCs) allow anyone with an internet connection access to high quality online classes, attracting thousands of students in each class. Large class sizes have created new challenges in creating *scalable assessments* that actively test student comprehension in large courses: While automatically gradable exercises such as Multiple Choice Questions (MCQs) allow scaling the task of grading, creating topical and relevant questions is a labor-intensive task. Motivated by the findings of "learning by explaining" [10], we have been building and using a student-sourced question creation and curation system in our classes.

In this work, we focus on a crucial component of this system: How can we assess students solely based on their responses to questions contributed by other students? A key challenge is that some of these questions may be misleadingly worded or simply wrong. As result, a stronger student may not choose the "correct" answer of such an ill-posed question, while a weaker student may answer the question "correctly." We illustrate an example from our class:

Example 1 (Misleading question). *In an undergraduate class on Strategy & IT, a student created the following multiple-choice question based on a case study:*

> What is the main reason for Zara's market dominance?
> a) Zara's one-week output time versus the six-month industry average and their ability to understand their customers.
> b) Zara's factory location in Spain gives them competitive advantage.
> c) Zara keeping their design and creation in-house instead of outsourcing.
> d) Zara keeping producing fewer products in order to create the appearance of scarcity and increase demand.

Roughly half of the students selected answer a), while one quarter each chose answers c) or d). The question author specified answer a as correct. However, this is incorrect! While Zara is indeed faster than the industry average (the idea behind the answer is correct), Zara's turnaround is 2 weeks, not 1 week. Students who had carefully read the case knew answer a) to be false and selected either c) or d), both of which are good reasons. The question is thus "misleading" as students with better understanding of the material are less likely to choose the answer specified as correct.[1]

Problem. Our goal is to develop a fast and un-supervised grading scheme that ranks students by their abilities and questions by their qualities, solely based on the responses to student-sourced questions. More formally, assume that m students answer a set of n

[1]The "discrimination score" of this question (defined in Section 2.1) is -0.4, showing that better students were less likely to pick the answer defined as correct. Intuitively, answering this question "correctly" should rather deduct than add points to the student score.

SIGMOD'18, 10 – 15 June 2018, Houston, Texas USA

ACM ISBN 111-1111-11-111/11/11…$15.00
https://doi.org/11.111/111_1

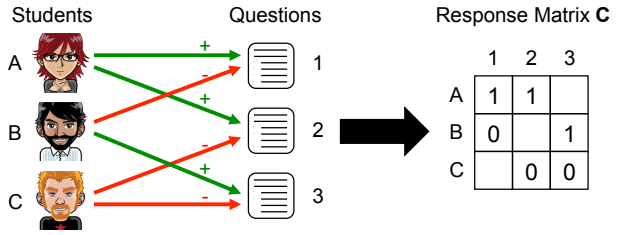


Figure 1: Our approach for student assessment builds upon and extends existing work in *truth discovery* [49]. Students get assigned to questions of unknown qualities (some of which may be misleading) and we want to accurately rank students based on their *responses* to questions (+ as correct, - as incorrect).

questions of unknown quality. Let $\mathbf{C}$ be the $m \times n$ binary response matrix where $C_{ij} = 1$ when student i is presented question j and answers in accordance with the specified "correct" solution, and $C_{ij} = 0$ otherwise. To accommodate large class sizes and potentially thousands of questions, students may be presented with only a constant number of questions independent of the size of n. In this case $\mathbf{C}$ will have missing entries. We will develop our methods for the case where every student attempts every question to simplify notation, and discuss how to extend them to the more realistic setting where students answer only a subset of questions in section 4.7. Note that under this model, students do not have the option of skipping a question: if a presented question is not attempted it is recorded as a 0 in $\mathbf{C}$. The main research question we address in this paper is the following:

> *How can we rank students by their abilities, solely based on the student responses* $\mathbf{C}$*, given that questions are created by students and may be misleading?*

This problem is very general and appears in various forms in many problems of data management, in particular truth discovery (Figure 1). We will build on truth discovery methods while guided by widely held assumptions in Item Response Theory (IRT), although our methods do not assume any generative models.

Student ranking desiderata. Before discussing our general approach, we suggest criteria for any student ranking approach. This will guide us towards our solution later.

1. *Accuracy*: The most important criterion is high accuracy. The inferred ranking of students should be close to the ranking based on the (hidden) student abilities.
2. *Robustness to misleading questions*: Since questions are student-sourced and potentially misleading, ranking methods should work well even if there is a large fraction of misleading questions.[2]
3. *Resilience to missing responses*: We want low drop in accuracy as the proportion of questions that students do not attempt increases. This is particularly relevant for large class sizes.
4. *Scalability* : We envisage use in an online system which provides students and instructors with a real-time ranking of students and questions and updates every time a student submits a new response. Such a ranking method should be fast and scale to very large class sizes, preferably with a linear dependence on the number of students or questions.
5. *Transparency and fairness*: The calculation of student scores needs to be transparent to the students. We require that a student's score should be the sum of the question scores that she has answered "correctly". If students are presented with different subsets of questions, fairness dictates that a two students who performs as well as possible on their respective questions be ranked equally. In this case, the student scores should be divided by the total number of points achievable.
6. *Identification of misleading questions*: We want our methods to be able to identify misleading questions. This may be useful both for ranking more accurately, and for identifying students who may benefit from additional attention.

Truth Discovery and HITS. Based on these requirements, we can identify a promising algorithm on which to base our methods. The *Hyperlinked Induced Topic Search (HITS)* algorithm of Kleinberg [24] (also known as hubs and authorities) is a popular method in the area of trust and verification of claims [18]. This iterative method first updates hub scores as the sum of the authority scores it is linked to, and then an authority's score as the sum of the hub scores it links to. This is repeated (after normalizing) until convergence. The method is fast and scalable to very large datasets. However, all scores in HITS are nonnegative and therefore cannot clearly identify misleading questions. Furthermore, as we will see later, it is *not robust* to misleading questions in our context (an increasing fraction of misleading question reduces the accuracy of HITS).

Overall Approach. In this paper, we propose several new ranking methods that fulfill our desiderata based on the same framework as HITS. Specifically, starting from an initial vector of student scores $\mathbf{s}_0$, estimate each question's quality from the responses $\mathbf{C}$, captured by an n-dimensional vector of question weights $\mathbf{q}$. Then, compute student i's score s_i to be the sum of the weights of the questions that the student answered correctly (according to the specified solution, which may be incorrect), i.e., $s_i = \sum_j C_{ij} q_j$, or in vector form, $\mathbf{s} = \mathbf{Cq}$. If student i is only presented with a subset of question N_i, compute the percentage of the maximum available score achieved, i.e. $s_i = \sum_{j \in N_i} C_{ij} q_j / \sum_{j \in N_i} \max\{0, q_i\}$. Recalculate question and student scores iteratively until a fixed point is reached. Notice that despite our focus on ranking, this framework provides a numerical score for each student which may be used when assigning grades.

This framework is inherently transparent, since the student scores are computed in a natural and easily understandable way. Also, a question's weight can be interpreted as measure of its quality. This will influence our assessment of both the students who answered it correctly, as well as the student who actually created a question.

Our new methods. We provide a new definition of what it means for a ranking method to be robust to misleading questions. We suggest two properties or ranking functions, *translation invariance* and *antisymmetry*, and show that ranking methods exhibiting them are robust to misleading questions. A simple modification of HITS, called *centered HITS*, satisfies these properties and allows for both positive and negative question scores. cHITS can be extended to an update rule more sensitive to the usefulness of a question in discriminating between students that we call the *biserial update (BSRL)* method – it updates question scores by incorporating the

[2] In one of our classes, we found up to 35% of questions in an early homework to be misleading, i.e. with negative discrimination score.

	Accuracy	*Robustness*	*Resilience*	*Scalability*	*Transparency*	*Identification*
AvgSc				✓✓	✓	
MIRT [9]	✓	✓✓	✓✓			✓
HITS [24]	✓		✓	✓✓	✓	
cHITS	✓✓	✓	✓	✓✓	✓	✓
BSRL	✓✓	✓✓	✓	✓✓	✓	✓
LogR	✓✓	✓✓	✓✓	✓	✓	✓

Figure 2: Overview of the six desirable properties satisfied by the different ranking methods.

point biserial correlation coefficient between the current estimate of student scores and the response vector for a specific question. When examined carefully under a statistical lens, BSRL can be interpreted as a maximum likelihood fit for a linear regression model for the question score values. Since the question responses are binary, this suggests an improvement of the method so that we fit for a binary response model such as the popular logistic regression model, rather than a linear regression model. We term this the *logistic regression (LogR) update* method. In addition to being a better fit for binary question responses, LogR update involves optimizing a likelihood function that is convex and hence is fast and globally optimal. While BSRL is inherently centered, we need to center the student scores in every iteration of LogR. All three methods – cHITS, BSRL and LogR – retain transparency, are iterative in nature and inherently scalable.

We carry out extensive computational evaluations of our three new methods to verify that they satisfy our desiderata. We find that LogR is as accurate as MIRT (the state-of-the-art maximum likelihood inference program based on Item Response Theory, see Section 2.1) across a wide range of parameter settings while being significantly more scalable and able to identify misleading questions, *even when the maximum likelihood estimator knows the IRT model from which the data was generated.* Crucially, LogR performs better than MIRT in the parameter regimes that most closely mimic large classroom scenarios where students answer only a subset of questions that are themselves student-sourced. Additionally, our new methods perform better than MIRT and HITS on experiments with data collected during the instruction of two classes where students attempted only a fraction of the total questions. Figure 2 summarizes our desiderata and our findings.

Summary of Contributions.

1. We propose a model for the problem of ranking students automatically using student-contributed questions and identify desired features that a ranking algorithm must possess.
2. We provide a new definition of what it means for a ranking method to be robust to misleading questions, and model such ill-coded questions as regular questions with responses flipped. We prove that methods with translation invariant and antisymmetric calibration functions are robust to misleading questions (Section 3.3).
3. We identify two simple baseline methods, ranking by the number of correct responses and HITS (Section 4), both of which are consistent.
4. We introduce the centered HITS, biserial and logistic methods for ranking students (Section 4). These are iterative methods that scale linearly and are provably robust to the introduction of misleading questions (Corollary 6, Corollary 9, Theorem 11).
5. We conduct extensive computational experiments in which we generate instances according to the 2PL IRT model. We find that, compared to the state-of-the-art maximum likelihood estimators for the 2PL model, our methods are as robust to ill-coded questions (Figure 5) while being several orders of magnitude faster (Figure 8). Furthermore, when the submitted questions are easy and students answer a constant number of questions independent of the class size, LogR is both significantly better at ranking students than competing methods (Figure 6) and more accurate at identifying misleading questions (**??**).
6. We finally confirm the advantages of our methods on two real world datasets (Table 2).

2 RELATED WORK

We first discuss Item Response Theory and literature pertaining to trust and reputation, which is central to our work before briefly mentioning other work related to computing systems for automatic assessment, peer-review methods and crowdsourcing.

2.1 Item Response Theory (IRT)

Item Response Theory (IRT) [5] is widely used to assess students, e.g. in the Scholastic Aptitude Test (SAT) [29] and Graduate Record Examinations (GRE) [23]. It models the probability of a student providing a correct response as a function of latent traits describing student ability and item factors characterizing the question. In the widely studied two parameter logistic (2PL) model, the *ability* of student i is captured by a single latent trait variable θ_i, and question j has a *difficulty* β_j and a *discrimination factor* α_j. The probability that student i answers questions j correctly is modeled as

$$\mathbf{P}(C_{ij} = 1|\theta_i, \alpha_j, \beta_j) = \frac{1}{1 + e^{-\alpha_j(\theta_i - \beta_j)}}, \tag{1}$$

and responses are assumed to be conditionally independent.

When a student's ability equals the difficulty of a question, then the student answers the problem correctly with probability $\frac{1}{2}$. The discrimination factor a_i determines the maximal slope of the *item characteristic curve* $f_j(\theta) = \mathbf{P}(C_{ij} = 1|\theta, a_j, b_j)$, which occurs at the inflection point where $\theta_i = b_j$. Many descriptions of this model explicitly restrict the discrimination factors to be non-negative [29][44, p. 14] [45, p. 147], while others allow negative discrimination factors [12]. Figure 3 shows how a change in difficulty or discrimination factor changes the likelihood of a correct response.

When $a_j > 0$, the probability of answering a question correctly increases with θ. In other words, better students are more likely to answer such a question correctly. In our context of student-sourced questions, it is important to note that if $a_j < 0$, the probability of answering question j "correctly" decreases with increasing student ability (recall Example 1 with an ill-coded question that has

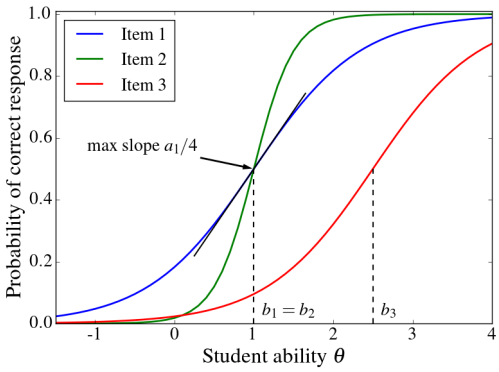


Figure 3: Item-characteristic curves under the 2PL model: Item 1 and 2 have the same difficulty ($b_1 = b_2 = 1$) but item 2 has a higher discrimination ($a_2 = 4$ vs. $a_1 = 1.5$). Item 3 has the same discrimination as item 1, but higher difficulty ($b_3 = 2.5$).

$a_j = -0.4$). The case of $a_j = 0$ corresponds to a completely non-discriminating question that all students are equally likely to answer correctly or incorrectly, independent of their abilities.

Example 1 (Continued). *Recall that for the question in Example 1, the discrimination factor was found to be approximately* -0.4. *Figure 4 shows the item characteristic curve for this question, as well as a representative sample of the responses received. Observe that stronger students were generally less likely to answer this ill-coded question correctly (according to the submitted solution).*

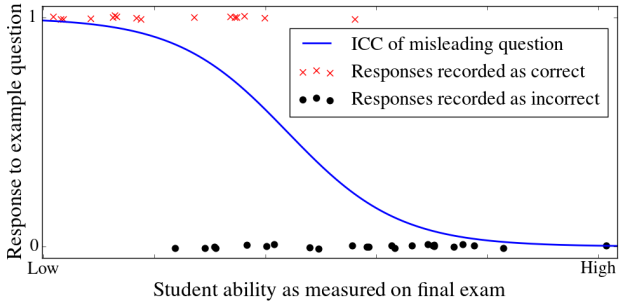


Figure 4: (Example 1 continued) Item-characteristic curve for an ill-coded question with discrimination score -0.4. Students who performed better in the final exam were more likely to answer the question incorrectly.

Interestingly, $\sum_j a_j C_{ij}$ is a sufficient statistic for the 2PL model [29]. A simple assessment method where each question is weighted by its discrimination score can therefore fully capture all the information that the response matrix $\mathbf{C}$ contains about the students, however, this requires accurate knowledge of the question discrimination parameters.

The basic 2PL model has been extended in various ways, e.g. to handle to multiple ("polytomous") responses [6, 40] and multi-dimensional latent traits [2, 14]. For more complex latent trait models, the optimal weight to assign a question is often a function of both the student's ability and the discrimination factor [29]. In practice this leads to an opaque grading scheme in which the weight of a question depends on the ability of the student attempting it.

Several software packages exist to estimate the IRT parameters using maximum likelihood methods and their extensions (BILOG, MULTILOG, PARAM-3PL and LTM [38] and MIRT [9] in R). We collectively refer to the set of such methods as "MIRT" in reference to the R-package MIRT used in our experiments.[3] However, these methods face at least four challenges in our scenario: (1) The validity of these methods relies on the assumption of very specific generative models; (2) Maximum likelihood methods may lack transparency, especially for more complex latent trait models; (3) IRT models face significant challenges in convergence when different questions have appreciable differences in their discrimination factors [1]. Since our questions are student-sourced, we expect a fair amount of variation in the question quality; and (4) For many popular IRT models, the computation of marginal maximum likelihood estimators for student abilities and question parameters involves numerical integration of intractable integrals and assumes that student abilities come from a normal distribution. These computations do not scale gracefully with the number of students.

In contrast, our methods scale linearly with the number of students and are transparent. Our methods are agnostic to the underlying generative model and are provably robust against misleading questions. Surprisingly, our experiments show that our algorithms are comparable to the 2PL maximum likelihood estimators across a wide range of criteria, even when the data is generated with a 2PL model. In the limited real-world data we have available, our methods are superior.

2.2 Truth inference and crowd labelling

The problem of "truth inference" in crowdsourcing assumes that there are workers and tasks, and each task has a true answer. A subset of workers attempt each questions, and the goal is then to infer the binary truth given workers' answers [49]. The assumption is that errors cancel each other out when averaged appropriately and methods vary in the way in which the correct label is guessed by averaging or aggregating individual answers.

Recent work by Shah et al. [41] proposes a model for generating workers' responses for the tasks based on a permutation of the workers (representing their ability) and another on the tasks (representing their difficulty): the probability of any worker correctly answering a task decreases with the rank of the task in the task difficulty ordering; similarly, the probability of any question being answered correctly increases with the rank of the worker in the worker ability ordering. This model generalizes simpler models such as the widely studied Dawid-Skene model [11] in crowd labeling. While the permutations in the model closely resemble the student ability and question difficulty ordering in IRT, it is not hard to verify that IRT models are more general than this permutation model. In particular, when all questions have the same positive discrimination factor a, then the response probabilities generated by the IRT model will satisfy the permutation properties above.

[3] This is not traditional, in the Item Response Literature "MIRT" is often an acronym for "Multi-dimensional Item Response Theory".

However two questions with the same value of difficulty and different discrimination factors a will *flip* the order of two students with higher and lower ability level than the common difficulty value, in terms of their chance of getting the two questions correct. In this way, IRT models are incomparable to the permutation model in [41]. Our work examines IRT and even more general generative models. Furthermore, the model of noise in this line of work assume random noise in the workers' answers, and no noise in the truth of the tasks submitted. In contrast, we examine noise in the questions submitted (since the questions themselves are crowd-sourced rather than just the collection of the answers), and model a noisy or misleading question by flipping the student responses that would have been given to these questions were they correct.

The crowd-labeling approach has been refined for various tasks, such as query answering [16], annotating Twitter data [15], and various other labeling tasks [43, 46]. Furthermore, novel crowdsourcing control methods such as LazySusan have been used to get high scores on the SAT Math test [28]. Together with latent credibility analysis, crowdsourcing has also been used to guess the answers to an IQ test with high accuracy [4].

In contrast, in our problem each individual question (task) is not true or false, but rather more or less discriminating between students. Our goal is to find the appropriate weight for each answer so as to correctly rank the students by their latent abilities. It is thus closer to the related problem of "truth discovery," i.e. solving the problem of extracting information from networks where the trustworthiness of the actors are unknown or uncertain [18, 34]. The most basic model of the problem is to consider a bipartite graph with one side made up of actors, the other side made up of their claims, and edges denoting associations between actors and claims. Furthermore, claims and actors are assumed to have "trustworthiness" and "believability" scores, respectively, with known a priori values. Iterative methods are used to update trust scores of actors to believability scores of claims, and vice versa, until convergence. Variants like PageTrust [22] and EigenTrust [21], attempt to estimate the trustworthiness or quality of sources, which is similar to our problem of identifying good students and high-quality questions. Other variants of these methods (such as Sums, HITS, AverageLog, TruthFinder, Investment, and PooledInvestment) have been extensively studied, are typically extremely fast, and have been proven in practice in a wide variety of settings [3, 17, 18, 24, 31–33, 47, 48]. Our setting is very similar: we have bipartite graph with students in one partition and questions in the other, and an edge exists between student i and question j exactly when $C_{ij} = 1$. It is natural to try to apply existing truth discovery methods to our problem and we will use HITS as an important baseline. Of the other methods mentioned in a survey [18, 31], only 'AverageLog' satisfies our transparency criterion but fails to satisfy a property called translation invariance, discussed in Section 3.2, which is critical for iterative methods to be robust to misleading questions. We will show that our methods are more accurate than HITS, that they are provably robust to misleading questions and can identify misleading questions.

Input: Response matrix $\mathbf{C}$, initial student scores $\mathbf{s}_0$
Output: Student scores $\mathbf{s}$, question weights $\mathbf{q}$

```
s ← s_0            // initialize student scores
repeat
    q ← f(C, s)    // update question weights
    s ← Cq         // update student scores
until convergence or iteration limit
```

Algorithm 1 Framework for transparent student rankings

2.3 Automatic student assessment

There has been a large body of recent work on computational approaches to assessing students in the presence of unreliable test questions [4, 13, 25–27, 30, 35]. Shah et al. [42] build a theoretical model where peer-grading at scale is unable to grade all but a vanishing fraction of students correctly if there is a small constant chance of wrong grading, despite having an instructor grade a constant fraction of the students. They propose a clustering based solution using dimension-reduction techniques to identify situations when this can be avoided. The models of quality and error they consider are highly stylized and unlikely to apply to our setting of crowd-sourced MCQs.

3 A FRAMEWORK FOR RANKING STUDENTS

In this section we discuss our model, which is broad enough to encompass several known models including those from IRT , and our solution approach in more detail.

We mention the characteristics we expect of a ranking method suitable for an online classroom and identify a framework for iterative algorithms inspired by the HITS algorithm which has the potential to fulfill these characteristics. Finally, we show that if an iterative ranking algorithm calibrates question weights in a *translation invariant* and *antisymmetric* way, then the method is robust to the introduction of misleading questions.

3.1 Desiderata and Problem Definition

We would like to develop a ranking algorithm that fulfills our desiderata: (1) it should be *accurate*; (2) it should be *robust* and work even if a subset $J \subseteq [n]$ of questions are ill-coded (i.e. students with higher abilities are less likely to answer those questions correctly); (3) the method should be *resilient* and remain accurate when students answer only a fraction k/n of all questions; (4) it should be *scalable* and scale linearly with the number of student-question observations; (5) it should be *transparent* and *fair* to students; and (6) it should clearly *identify* misleading questions (see Figure 2).

Our problem can be formalized as follow.

Definition 2 (Student ranking). *Given a partially filled response matrix* $\mathbf{C}$*, find weights for the questions so that the weighted answer scores for students reflect the true order in student abilities.*

3.2 An Iterative Framework for Ranking

Our task is to to infer an n-dimensional vector of question weights $\mathbf{q}$, and an m-dimensional vector of student test scores $\mathbf{s}$. Our proposed ranking framework defines a solution by a system of multiple constraints. We use iterative updates to find a fixed point $(\mathbf{s}, \mathbf{q})$. The solution is defined by three parts:

1. *Initialization*: Start with an initialization of the student scores $\mathbf{s}_0$. For some methods, the choice of $\mathbf{s}_0$ does not affect the resulting fixed point. In the absence of additional information, one may initialize the student scores with the number of questions a student answered correctly: $\mathbf{s}_0 = \mathbf{C}\mathbf{1}_n$.
2. *Question weights*: The question weights $\mathbf{q}^{(t)}$ are calibrated to give higher weights to "better" questions. We utilise a calibration function $f : \{0,1\}^{m\times n} \times \mathbb{R}^m \to \mathbb{R}^n$ that calculates the question weights based on the student scores and the response matrix:
$$\mathbf{q}^{(t)} \leftarrow f(\mathbf{C}, \mathbf{s}^{(t-1)}).$$
Every method we discuss is defined by a different function f and our goal will be to find functions with appropriate mathematical properties. Furthermore, although not necessary in general, the functions f we discuss all decompose into identical functions $g : \{0,1\}^m \times \mathbb{R}^m \to \mathbb{R}$, and the update rule for $q_j^{(t)}$ uses only the jth column $\mathbf{C}_j$:
$$q_j^{(t)} \leftarrow g(\mathbf{C}_j, \mathbf{s}^{(t-1)}). \tag{2}$$
3. *Student scores*: Transparency demands that student test scores are the weighed sum of the questions correctly answered by the students. For each question, we use the question weights calculated in the previous step to update the student scores.
$$\mathbf{s}^{(t)} \leftarrow \mathbf{C}\,\mathbf{q}^{(t)}. \tag{3}$$

Iterative updates are performed until the student and question scores converge, or until an iteration limit is reached. (See Algorithm 1).

Notice that while we require student scores to be assigned in a transparent way, (2) allows the question weights to be determined in arbitrarily complex ways through appropriate choices of a function g. Still, we argue that a reasonable calibration function g should possess the following two properties:

1. Let $g(\mathbf{c}, \mathbf{s})$ be the computed question weight based on the student scores $\mathbf{s}$ and student responses $\mathbf{c}$ (corresponding to a column in $\mathbf{C}$). Then the question weight should not change if all student scores are increased by the same number. More formally, we say that g is *translation invariant* if
$$\forall \mathbf{c} \in \{0,1\}^m, \mathbf{s} \in \mathbb{R}^m, \alpha \in \mathbb{R}.\ g(\mathbf{c}, \mathbf{s} + \alpha\mathbf{1}) = g(\mathbf{c}, \mathbf{s}) \tag{4}$$
2. If the question response vector $\mathbf{c} \in \{0,1\}^m$ is flipped to $\mathbf{1} - \mathbf{c}$, then the question score $g(\mathbf{1} - \mathbf{c}, \mathbf{s})$ should reflect this. We say that g is *antisymmetric* if
$$\forall \mathbf{c} \in \{0,1\}^m, \mathbf{s} \in \mathbb{R}^m.\ g(\mathbf{1}-\mathbf{c}, \mathbf{s}) = -g(\mathbf{c}, \mathbf{s}). \tag{5}$$

Intuitively, translation invariance states that, since we are concerned with student rankings, adding a constant to each student score should not change the calibration score, since the underlying ranking stays the same. Antisymmetry ensures that the calibration function g is sensitive to a reversal of the student responses to a particular question (as may occur in the case of an ill-coded question) provided that the vector of student scores s that we are using to calibrate the question scores stays the same. Flipping the sign of the calibration function for flipped responses is motivated by the use of negative scores to flag ill-coded questions.

In Section 4, we discuss two baseline and three novel ranking methods derived by modifying the calibration functions g in order to satisfy properties (4) and (5). Before that, we show that these properties help satisfy one of our desiderata: robustness to misleading questions.

3.3 A Response-Flipping Model for Misleading Questions

Let $\mathbf{C}$ be a given response matrix and let $\mathbf{C}_j$ be a short form for $\mathbf{C}_{:j}$, referring to the j-th column of $\mathbf{C}$. Further, let $J \subseteq [n]$ be a subset of misleading questions. We model such questions as those whose responses have been inverted $\mathbf{C}'_j = \mathbf{1}_m - \mathbf{C}_j$.

This is a particularly simple form of generating misleading questions from good questions that is well justified: Consider a good multiple-choice question j with two potential answers and the student response vector $\mathbf{C}_j$. Now assume an alternative world in which the correct answer was erroneously specified as the incorrect answer. The response vector for this misleading question would be exactly $\mathbf{C}'_j = \mathbf{1} - \mathbf{C}_j$. Note that this response-flipping model is also consistent with the 2PL generative model if we interpret the misleading version of a question to be the same question with its discrimination value flipped in sign, since $\mathbf{P}(C_{ij} = 1|\boldsymbol{\theta}, a_j, b_j) = 1 - \mathbf{P}(C_{ij} = 1|\boldsymbol{\theta}, -a_j, b_j)$.

Formally, let $\mathbf{C}'$ be the matrix with these inverted responses replacing the corresponding columns in $\mathbf{C}$. The following result shows that if g is antisymmetric, then the score vector calculated from the current question vector $\mathbf{q}$ will imply the same student ranking, regardless whether we use $\mathbf{C}$ or $\mathbf{C}'$.

Lemma 3 (Rank invariance to response-flipping). *Suppose we have a function $g : \{0,1\}^m \times \mathbb{R}^m \to \mathbb{R}$ that satisfies antisymmetry (5). Let $\boldsymbol{\sigma} \in \mathbb{R}^m$ be a student score vector, and $\mathbf{C}_j \in \{0,1\}^m$, $j \in [n]$, be a set of binary vectors. Fix an arbitrary subset $J \subseteq [n]$ and define two vectors $\mathbf{s}, \mathbf{s}' \in \mathbb{R}^m$ from using the two update rules (2) and (3) with $\mathbf{C}$ and $\mathbf{C}'$ respectively:*
$$s_i = \sum_{j\notin J} C_{ij} g(\mathbf{C}_j, \boldsymbol{\sigma}) + \sum_{j\in J} C_{ij} g(\mathbf{C}_j, \boldsymbol{\sigma})$$
$$s'_i = \sum_{j\notin J} C_{ij} g(\mathbf{C}_j, \boldsymbol{\sigma}) + \sum_{j\in J} (1 - C_{ij}) g(\mathbf{1} - \mathbf{C}_j, \boldsymbol{\sigma}).$$
Then the elements of $\mathbf{s}$ and $\mathbf{s}'$ differ by the same constant and, hence, imply the same student ordering.

Proof. We rewrite s'_i using antisymmetry of g:
$$\begin{aligned} s'_i &= \sum_{j\notin J} C_{ij} g(\mathbf{C}_j, \boldsymbol{\sigma}) + \sum_{j\in J} (1 - C_{ij}) g(\mathbf{1} - \mathbf{C}_j, \boldsymbol{\sigma}) \\ &= \sum_{j\notin J} C_{ij} g(\mathbf{C}_j, \boldsymbol{\sigma}) + \sum_{j\in J} (C_{ij} - 1) g(\mathbf{C}_j, \boldsymbol{\sigma}) \\ &= s_i - \sum_{j\in J} g(\mathbf{C}_j, \boldsymbol{\sigma}). \end{aligned}$$

The result now follows since each s'_i is simply s_i offset by the same constant $\sum_{j\in J} g(\mathbf{C}_j, \boldsymbol{\sigma})$. □

We next show that if g is translation invariant (4) in addition to being antisymmetric (5), then the rankings resulting from using $\mathbf{C}$ and $\mathbf{C}'$, respectively, will remain the same across multiple iterations.

Theorem 4 (Iterative rank invariance). *Let g be a calibration rule satisfying* (4) *and* (5). *Let $\mathbf{q}^{(t)}, \mathbf{s}^{(t)}$ and $\boldsymbol{\gamma}^{(t)}, \boldsymbol{\sigma}^{(t)}$ be question and student scores at iteration t generated from $\mathbf{C}$ and $\mathbf{C}'$ respectively, both starting from the same $\mathbf{s}^{(0)} \in \mathbb{R}^m$. Then, at every iteration t, the elements of $\mathbf{s}^{(t)}$ and $\boldsymbol{\sigma}^{(t)}$ differ by the same constant $\alpha^{(t)}$ and, hence, imply the same student ordering.*

Proof. We prove this by induction on t. The base case $t = 1$ follows by applying Lemma 3 with $\mathbf{s}^{(0)} = \mathbf{s}_0$. Suppose now that it holds for $t-1 \geq 1$, that is, $\boldsymbol{\sigma}^{(t-1)} = \mathbf{s}^{(t-1)} + \alpha^{(t-1)}\mathbf{1}_m$. By translation invariance (4), the question scores $\gamma_j^{(t-1)} = g(\mathbf{C}'_j, \boldsymbol{\sigma}^{(t-1)}) = g(\mathbf{C}'_j, \mathbf{s}^{(t-1)})$. It follows from Lemma 3 with $\mathbf{s} = \mathbf{s}^{(t-1)}$ that $\boldsymbol{\sigma}^{(t)}$ differs from $\mathbf{s}^{(t)}$ by a constant. This completes the induction hypothesis, and the proof. □

The important implication of this result is that a calibration function g that is both *translation invariant* and *antisymmetric* will lead to a transparent method for ranking students that is *robust to misleading questions*. Concretely, starting from an initialization vector $\mathbf{s}^{(0)}$, and flipping any subset J of columns will not change the student ranking implied by the student scores after any fixed number of iterations.

Interestingly, our results suggest that an appropriate calibration function g will give a robust ranking method for *any* fraction of misleading questions, *if given the same starting* s_0. Loosely, this states that the response vectors $\mathbf{C}_j$ and $\mathbf{C}'_j$ provide the same amount of information about students' relative abilities. This holds irrespective of the generative model underlying the creation of $\mathbf{C}$.

In practice, however, if too many questions are misleading, it might affect our initial vector $\mathbf{s}_0$, which would result in a different ranking. For example, if $\mathbf{s}_0 = \mathbf{C}\mathbf{1}_n$ and $|J| > n/2$, we expect $\mathbf{s}'_0$ to invert the ranking given by s_0.

4 FIVE METHODS FOR RANKING

In this section, we first review two simple baseline methods: first, using the number of correct responses, and second, the HITS update method [24]. We will show that they do not satisfy translation invariance (4) nor asymmetry (5). Motivated by this observation, we then modify these baseline methods to derive three novel update rules satisfying properties (4) and (5). Finally, we discuss the computational complexity of these methods and how to modify the algorithms when students only answer $k < n$ questions. We validate our findings in the next section with our experimental results.

4.1 Number of correct responses (AvgSc)

The simplest way to rank students is by the *number of recorded correct responses* (AvgSc). This means that all questions have equal weight, so we set

$$g^{\text{AvgSc}}(\mathbf{c}, \mathbf{s}) := 1.$$

No iteration is required.

In the presence of questions of different qualities it may be beneficial to weigh better questions more heavily and to identify misleading questions. Clearly, g^{AvgSc} does not satisfy (5).

4.2 Hubs and Authorities (HITS)

The popular HITS algorithm [24] forms the basis from which various variants have been applied and studied in truth discovery (Section 2.2).

In our setting, we formulate a bipartite graph with $m + n$ nodes, with student nodes on one side, and question nodes on the other. The $\mathbf{C}$ matrix gives the edge structure, with $C_{ij} = 1$ indicating the existence of an edge from j pointing to i. The HITS algorithm computes hub scores *only* for questions, and authority scores *only* for students (since edges only point from questions to students). This translates to saying that good questions are those which are answered by strong students, and strong students are answering the good questions correctly.

In our formalism, the inner product recalibration rule

$$g^{\text{HITS}}(\mathbf{c}, \mathbf{s}) := \mathbf{c}^\intercal \mathbf{s} \qquad \text{(HITS)}$$

together with (2), (3) captures the HITS algorithm.

We can write the update rule for the student scores $\mathbf{s}^{(t)}$ without the question scores:

$$\mathbf{s}^{(t)} = \mathbf{C}\mathbf{C}^\intercal \mathbf{s}^{(t-1)} = (\mathbf{C}\mathbf{C}^\intercal)^t \mathbf{s}_0,$$

where the second equality is form recursively applying the first equality. The HITS update rule for $\mathbf{s}^{(t)}$ corresponds to applying an un-normalised version of the well-studied power method for computing leading eigenvalues of the matrix $\mathbf{C}\mathbf{C}^\intercal$ [39]. In fact, $\mathbf{s}^{(t)}/\|\mathbf{s}^{(t)}\|$ directly corresponds to iterations of the power method. In other words, as long as the initial student score vector $\mathbf{s}_0$ is not orthogonal to the leading eigenspace, $\mathbf{s}^{(t)}/\|\mathbf{s}^{(t)}\|$ converges to a leading eigenvector of $\mathbf{C}\mathbf{C}^\intercal$. Thus, since $\mathbf{s}^{(t)}/\|\mathbf{s}^{(t)}\|$ and $\mathbf{s}^{(t)}$ give the same ranking of the students, the final ranking will be given by the leading eigenvector of $\mathbf{C}\mathbf{C}^\intercal$.

It is possible to show that as the number of questions increase to infinity (while m stays fixed), the probability of the leading eigenvector of $\mathbf{C}\mathbf{C}^\intercal$ ranking students correctly converges to 1. We include this result in the full version.

The update rule g^{HITS} does not satisfy translation invariance (4) or antisymmetry (5), so we do not expect it to perform well in the presence of misleading questions. Notice also that if the initial student score vector $\mathbf{s}_0$ is non-negative, $\mathbf{q}^{(t)}$ is also non-negative since $\mathbf{C}\mathbf{C}^\intercal$ has only non-negative entries. This means that HITS will always give positive scores to (even misleading) questions and is not able to identify misleading questions with negative question scores, which is one of the desiderata discussed in Section 1.

4.3 Centered HITS (cHITS)

We next add a simple correction to HITS that gives us translation invariance (4) and brings us closer to our goal of a ranking scheme that is robust to misleading questions. Specifically, we first "center" the student score vector, then compute the question weights as before. This results in the update

$$g^{\text{cHITS}}(\mathbf{c}, \mathbf{s}) := g^{\text{HITS}}(\mathbf{c}, \mathbf{s} - \bar{s}\mathbf{1}_m) = \mathbf{c}^\intercal (\mathbf{s} - \bar{s}\mathbf{1}_m), \qquad \text{(cHITS)}$$

where $\bar{s} := \mathbf{1}_m^\top \mathbf{s}/m$ is the mean. By construction, this new update rule satisfies translation invariance (4). Remarkably, this simple fix to HITS also gives us antisymmetry (5).

Lemma 5 (Antisymmetry of cHITS). *Let* $\mathbf{s} \in \mathbb{R}^m$ *and* $\mathbf{c} \in \{0,1\}^m$ *be arbitrary. Then*

$$g^{\text{cHITS}}(\mathbf{1}_m - \mathbf{c}, \mathbf{s} - \bar{s}\mathbf{1}_m) = -g^{\text{cHITS}}(\mathbf{c}, \mathbf{s} - \bar{s}\mathbf{1}_m).$$

Proof. Observe that $\mathbf{1}_m^\top(\mathbf{s} - \bar{s}\mathbf{1}_m) = 0$. The result follows. □

It now follows trivially from Theorem 4 and lemma 5 that the cHITS update rule is robust to misleading questions.

Corollary 6 (Robustness of cHITS). *Let* $\mathbf{s}^{(t)}$ *and* $\boldsymbol{\sigma}^{(t)}$ *denote the student scores of the cHITS algorithm on* $\mathbf{C}$ *and* $\mathbf{C}'$*, respectively. If* $\mathbf{s}_0 = \boldsymbol{\sigma}_0$*, then for every iteration* t*,* $\mathbf{s}^{(t)}$ *and* $\boldsymbol{\sigma}^{(t)}$ *give the same ranking.*

Additionally, (cHITS) allows negative question scores, as opposed to (HITS), and has the potential to identify misleading questions.

Finally, we show that cHITS is equivalent to an eigenvector problem, which also shows it is convergent. Denoting $\mathbf{W} = \mathbf{I}_m - \mathbf{1}_m\mathbf{1}_m^\top/m$, we can write the updates in matrix form as

$$\mathbf{q}^{(t-1)} = \mathbf{C}^\top\mathbf{W}\mathbf{s}^{(t-1)}, \text{and } \mathbf{s}^{(t)} = \mathbf{C}\mathbf{q}^{(t-1)}.$$

For the student scores $\mathbf{s}^{(t)}$ this becomes

$$\mathbf{s}^{(t)} = \mathbf{C}\mathbf{C}^\top\mathbf{W}\mathbf{s}^{(t-1)} = (\mathbf{C}\mathbf{C}^\top\mathbf{W})^t\mathbf{s}_0, \tag{6}$$

where the second equality follows from recursively applying the first. Like the updates for HITS, (6) is simply an un-normalised power method update, except that $\mathbf{C}\mathbf{C}^\top\mathbf{W}$ is not symmetric. However, we can still show that the iterations converge to a transformed eigenvector of a symmetric matrix. The full proof is deferred to the appendix.

Theorem 7. *Compute* $\mathbf{s}^{(t)}$ *according to* (6)*. Then* $\mathbf{s}^{(t)}/\|\mathbf{s}^{(t)}\| \to \mathbf{C}\mathbf{C}^\top\mathbf{e}$*, where* $\mathbf{e}$ *is the eigenvector of the largest eigenvalue of the symmetric positive definite matrix* $\mathbf{W}^\top\mathbf{C}\mathbf{C}^\top\mathbf{W}$*, not orthogonal to* $\mathbf{s}_0$*.*

4.4 Biserial update (BSRL)

We now explore a possible improvement to cHITS. Consider two questions, one is answered correctly by half of the students and the other by only a small fraction of students. It seems reasonable that a correct answer to the latter question should be worth more points than a correct answer to the former. A similar argument suggests weighing a question answered correctly by the vast majority of students more heavily places a larger relative penalty on the few students answered it incorrectly. A natural way to do this is to scale a question's response vector by its standard deviation. Let $f_j := \mathbf{1}_m^\top\mathbf{C}_j/m$ denote the fraction of students who answered question j correctly. The resulting update rule is

$$q_j^{(t+1)} = \frac{\mathbf{C}_j^\top\left(\mathbf{s}^{(t)} - \bar{s}^{(t)}\mathbf{1}_m\right)}{\|\mathbf{C}_j - f_j\mathbf{1}_m\|} = \frac{(\mathbf{C}_j - f_j\mathbf{1}_m)^\top\left(\mathbf{s}^{(t)} - \bar{s}^{(t)}\mathbf{1}_m\right)}{\|\mathbf{C}_j - f_j\mathbf{1}_m\|}. \tag{7}$$

The second equality holds because $f_j\mathbf{1}_m^\top\left(\mathbf{s}^{(t)} - \bar{s}^{(t)}\mathbf{1}_m\right) = 0$.

This update rule has close ties to the *cosine similarity* between $\mathbf{C}_j$ and $\mathbf{s}^{(t)}$ and the least-squares solution when fitting a linear curve to the relationship between $\mathbf{C}_j$ and $\mathbf{s}^{(t)}$, however, we prefer the following interpretation. Suppose that we have an estimate of student scores $\mathbf{s}$. 'Good' questions are those where high score students answer correctly, and low score students do not, whereas 'bad' questions are the opposite. The aim is therefore to assign each question j a score that is based on how the responses $\mathbf{C}_j$ are correlated with the scores $\mathbf{s}$. We elect to use the *Pearson correlation coefficient* and refer to it simply as correlation.

Let $\bar{x} = \mathbf{1}^\top\mathbf{x}/m$ be the mean of $\mathbf{x} \in \mathbb{R}^m$. The correlation of two vectors $\mathbf{x}, \mathbf{y}$ can be written as

$$\text{Corr}(\mathbf{x}, \mathbf{y}) = \frac{(\mathbf{x} - \bar{x}\mathbf{1})^\top(\mathbf{y} - \bar{y}\mathbf{1})}{\|\mathbf{x} - \bar{x}\mathbf{1}\|\,\|\mathbf{y} - \bar{y}\mathbf{1}\|}.$$

It is also known as the *point biserial correlation* if one of the vectors is binary-valued. We know from the Cauchy-Schwarz inequality that the correlation is bounded between -1 and 1.

Our third update rule called BSRL ("biserial") is thus defined by

$$g^{\text{BSRL}}(\mathbf{c}, \mathbf{s}) := \text{Corr}(\mathbf{c}, \mathbf{s}). \tag{BSRL}$$

Comparing this to (7), we have an extra factor $1/\|\mathbf{s}^{(t-1)} - \bar{s}^{(t-1)}\mathbf{1}_m\|$, which affects all question scores similarly, hence does not change the resulting ranking of students.

Like cHITS, BSRL is robust to misleading questions. We first show that g^{BSRL} is antisymmetric (5).

Lemma 8 (Antisymmetry of BSRL). *Let* $\mathbf{s} \in \mathbb{R}^m$ *and* $\mathbf{c} \in \{0,1\}^m$ *be arbitrary. Then*

$$g^{\text{BSRL}}(1 - \mathbf{c}, \mathbf{s}) = -g^{\text{BSRL}}(\mathbf{c}, \mathbf{s}).$$

Proof. Let $\bar{c}$ be the mean of $\mathbf{c}$. Then $1 - \bar{c}$ is the mean of the complement vector $\mathbf{1} - \mathbf{c}$. It follows that

$$\begin{aligned} g^{\text{BSRL}}(\mathbf{1} - \mathbf{c}, \mathbf{s}) &= \frac{\langle \mathbf{s} - \bar{s}\mathbf{1}, \mathbf{1} - \mathbf{c} - (1 - \bar{c})\mathbf{1}\rangle}{\|\mathbf{s} - \bar{s}\mathbf{1}\|\|\mathbf{1} - \mathbf{c} - (1 - \bar{c})\mathbf{1}\|} \\ &= \frac{\langle \mathbf{s} - \bar{s}\mathbf{1}, -\mathbf{c} + \bar{c}\mathbf{1}\rangle}{\|\mathbf{s} - \bar{s}\mathbf{1}\|\|-\mathbf{c} + \bar{c}\mathbf{1}\|}, \end{aligned}$$

which equals $-g^{\text{BSRL}}(\mathbf{c}, \mathbf{s})$. □

Furthermore, since BSRL is a correlation, it trivially satisfies translation invariance (4). It then follows directly from Theorem 4 and lemma 8 that, if the biserial update rule is initialized with a student vector $\mathbf{s}^{(0)}$, then it is robust to misleading questions:

Corollary 9 (Robustness of BSRL). *Let* $\mathbf{s}^{(t)}$ *and* $\boldsymbol{\sigma}^{(t)}$ *denote the student scores of the biserial algorithm on* $\mathbf{C}$ *and* $\mathbf{C}'$*, respectively, where* $\mathbf{C}'$ *is obtained by flipping a subset of columns of* $\mathbf{C}$*. If* $\mathbf{s}^{(0)} = \boldsymbol{\sigma}^{(0)}$*, then for every iteration* t*,* $\mathbf{s}^{(t)}$ *and* $\boldsymbol{\sigma}^{(t)}$ *give the same ranking.*

Surprisingly, it is possible to show that BSRL is equivalent to an eigenvector calculation for a symmetric positive definite matrix and therefore convergent. Denote $\mathbf{W} = \mathbf{I}_m - \mathbf{1}_m\mathbf{1}_m^\top/m$ and let $\mathbf{D}_C$ to be the $n \times n$ diagonal matrix with entries $1/\|\mathbf{C}_j - \mathbf{1}_m f_j\|$. In matrix form, the updates are

$$\mathbf{q}^{(t-1)} = \mathbf{D}_C\mathbf{C}^\top\mathbf{W}\mathbf{s}^{(t-1)}, \text{ and } \mathbf{s}^{(t)} = \mathbf{C}\mathbf{q}^{(t-1)}.$$

The update rule written just in terms of the student scores $\mathbf{s}^{(t)}$ is

$$\mathbf{s}^{(t)} = \mathbf{C}\mathbf{D}_C\mathbf{C}^\top\mathbf{W}\mathbf{s}^{(t-1)} = (\mathbf{C}\mathbf{D}_C\mathbf{C}^\top\mathbf{W})^t\mathbf{s}_0, \tag{8}$$

where the second equality follows by recursively applying the first. As with HITS and cHITS, these are un-normalised iterations of the

power method, except that $\mathbf{C}\mathbf{D}_C\mathbf{C}^\intercal\mathbf{W}$ is not symmetric. However, we can still show that the final ranking of BSRL is equivalent to computing an eigenvector. The full proof appears in the appendix.

Theorem 10. *Compute* $\mathbf{s}^{(t)}$ *according to* (8). *Then* $\mathbf{s}^{(t)}/\|\mathbf{s}^{(t)}\| \to \mathbf{C}\mathbf{D}_C\mathbf{C}^\intercal\mathbf{e}$, *where* $\mathbf{e}$ *is the eigenvector of the largest eigenvalue of the symmetric positive definite matrix* $\mathbf{W}^\intercal\mathbf{C}\mathbf{D}_C\mathbf{C}^\intercal\mathbf{W}$ *that is not orthogonal to* $\mathbf{s}_0$.

4.5 Logistic update (LogR)

The motivation for the biserial update rule was to measure correlation between estimated student scores $\mathbf{s}$ and the binary question response vectors $\mathbf{C}_j$. For modelling binary responses, however, *logistic regression* is the most widely used method and appears as more appropriate method. In logistic regression, we aim to predict each response C_{ij} with the estimated student score s_i by assuming the following model:

$$\mathbf{P}(C_{ij} = 1 \mid s_i) = \frac{1}{1 + \exp(-a_j s_i - b_j)},$$

where a_j, b_j are parameters to be learned. With this model, the measure of correlation is a_j, since positive a_j implies that as s_i increases, then $\mathbb{P}[C_{ij} = 1 \mid s_i] \to 1$, i.e., higher scores s_i means we are more likely to see $C_{ij} = 1$. Thus, the responses $\mathbf{C}_j$ are positively correlated with $\mathbf{s}$, and vice versa for negative a_j.

The logistic regression based update rule is therefore

$$g^{\text{LogR}}(\mathbf{c}, \mathbf{s}) \tag{LogR}$$
$$:= \arg_a \max_{a,b} \sum_{i=1}^{m} \left[c_i(as_i + b) - \ln(1 + \exp(as_i + b))\right],$$

which is simply the Maximum Likelihood Estimation (MLE) for the logistic regression model specified above.

As before, let $\mathbf{C}$ be an arbitrary response matrix and let $\mathbf{C}' = \mathbf{C}_J$ for some $J \subseteq [n]$. We find that, just like BSRL and cHITS, LogR is robust to the introduction of misleading questions.

Theorem 11 (Robustness of LogR). *Let* $\mathbf{s}$ *and* $\boldsymbol{\sigma}$ *denote the student scores of the logistic algorithm on* $\mathbf{C}$ *and* $\mathbf{C}'$, *respectively. If* $\mathbf{s}^{(0)} = \boldsymbol{\sigma}^{(0)}$, *then for every iteration* t, $\mathbf{s}^{(t)}$ *and* $\boldsymbol{\sigma}^{(t)}$ *gives the same ranking.*

The proof of this is again a direct application of Theorem 4, after showing that the g^{LogR} is translation invariant and antisymmetric. We defer the details to the appendix.

Theorem 12 (Convergence of LogR). *LogR converges.*

Proof. First notice that there is an isomorphic translation between the 2PL IRT model and logistic regression. To see this, let $\Psi(x)$ stand for the *logistic cumulative distribution* function $\Psi(x) = (1 + \exp(-x))^{-1}$. The 2PL IRT model then assumes item characteristic curves have the form of a logistic cumulative distribution function: $\mathbb{P}_j(\theta_i) = \Psi(L_j^{\text{IRT}}(\theta_i))$ with $L_j^{\text{IRT}}(\theta_i) = \alpha_j(\theta_i - \beta_j)$. In contrast logistic regression uses $\mathbb{P}_j(s_i) = \Psi(L_j^{\text{LR}}(s_i))$ with $L_j^{\text{LR}}(s_i) = a_j s_i + b_j$. The translation is then

$$\begin{aligned} \alpha_i &= a_i \\ \beta_i &= -b_i/a_i \end{aligned} \tag{9}$$

We next state an old result from IRT [**?** , Sec. 18]:

Lemma 13 (Sufficient statistic). $\mathbf{C}_{i:}\boldsymbol{\alpha}$ *is sufficient statistic for student abilities* θ_i *in the 2PL IRT model.*

In other words, all the information about a student i's ability θ_i available in a response pattern $\mathbf{C}_{i:}$ is given by above formula which crucially *does not depend on the question difficulty parameters* $\boldsymbol{\beta}_j$. From the translation, it also follows that $\mathbf{C}_{i:}\mathbf{a}$ is sufficient statistic for student score $\mathbf{s}_i$ in the logistic regression and thus unique (up to isomorphism).

To illustrate the implication, consider the function that logistic regression (and thus our logistic update rule) maximizes at each iteration:

$$\mathbf{a} \leftarrow \arg_{\mathbf{a}} \max_{\mathbf{a},\mathbf{b}} L(\mathbf{a}, \mathbf{b}, \mathbf{s})$$

with

$$L(\mathbf{a}, \mathbf{b}, \mathbf{s}) = \sum_{j=1}^{n}\sum_{i=1}^{m} C_{ij}(a_j s_i + b_j) - \sum_{j=1}^{n}\sum_{i=1}^{m} \ln\left(1 + \exp(a_j s_i + b_j)\right) - \frac{\mathbf{a}^2}{2\lambda}$$

From the translation eq. (9), it follows that our logistic update rule finds the $\boldsymbol{\alpha}$ that maximizes $L(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\theta})$ (irrespective of $\boldsymbol{\beta}$). And from lemma 13 follows that $\mathbf{C}_{i:}\mathbf{a}$ is a sufficient statistic for $\mathbf{s} \leftarrow \arg\max_{\mathbf{s}} L(\mathbf{a}, \mathbf{b}, \mathbf{s})$.

In other words, at each iteration, the two steps of our update rule solve the following two subproblems: $\mathbf{a} \leftarrow \arg_{\mathbf{a}} \max_{\mathbf{a},\mathbf{b}} L(\mathbf{a}, \mathbf{b}, \mathbf{s})$ by keeping $\mathbf{s}$ fixed, and $\mathbf{s} \leftarrow \arg\max_{\mathbf{s}} L(\mathbf{a}, \mathbf{b}, \mathbf{s})$ by keeping $\mathbf{a}, \mathbf{b}$ fixed (recall that the value of $\mathbf{b}$ does not impact the solution for $\mathbf{s}$). In other words, our update rule is isomorphic to an application of coordinate gradient ascent [**?**]. Since the parameters are bounded in the updates (due to the regularization), such a local optimum always exists, which proves convergence of LogR, starting from any initial values. □

4.6 Runtime Discussion

We now discuss the cost per iteration of our methods. Computing the student score vector $\mathbf{s}$ for number of correct responses takes time $O(mn)$ each iteration, and terminates after one iteration. HITS essentially computes two matrix-vector multiplications with $\mathbf{C}$, and hence the cost is $O(mn)$ each iteration. Centered HITS requires a additional pass through the vector of student scores, however, it retains the $O(mn)$ asymptotic complexity. BSRL consists of one matrix-vector product, which is $O(mn)$, and n correlation computations, which is $O(m)$ each. Thus the total cost per iteration is $O(mn)$. For LogR, each iteration we solve n different two-dimensional convex programs, each of which takes $O(\log(1/\epsilon))$ iterations to solve up to accuracy ϵ when using interior point solvers. However, note that since the objective is a sum of m terms, computing gradients for this sum takes $O(m)$ time. Thus, the total cost per iteration is $O(mn \log(1/\epsilon))$.

4.7 Methods for missing responses

Suppose each student is assigned only a subset $k < n$ of questions. Even if two students can solve all of their assigned k questions, it is quite possible that they had a different number of points available to them. We thus need to make sure that the point assignment is both *transparent* and *fair*.

Transparent aggregation. It is natural and transparent, for any fixed question weights, to let a student's score be the fraction of

the available points which they received. Formally, suppose that student i attempts the set of questions indexed by N_i. The best score achievable on the subset of questions N_i is $\sum_{j\in N_i} \max\{0, q_j\}$. Observe that on the questions with negative weight (the ill-coded questions) we expect a perfect student to answer "incorrectly" and score 0 instead of $q_j < 0$. If students answer different subsets of questions we let a student score be the fraction of this maximum achievable score attained and replace (3) with

$$s_i = \frac{\sum_{j\in N_i} C_{ij} q_j}{\sum_{j\in N_i} \max\{0, q_j\}}. \tag{10}$$

In addition to this aggregation method, it is necessary to modify the calibration functions of some of our methods to take missing responses into consideration. We next describe these modifications for each of our 5 methods:

1) AvgSc: When using AvgSc, all questions have weight 1 and no modification beyond dividing the number of correct responses by the number of questions attempted is required. Thus,

$$g^{\text{AvgSc-M}}(\mathbf{C}_j, \mathbf{s}) = g^{\text{AvgSc}}(\mathbf{C}_j, \mathbf{s}) = 1.$$

2) HITS: For HITS, we modify g^{HITS} analogously to (10). Let M_j be the set of students who attempted question j. Now

$$g^{\text{HITS-M}}(\mathbf{C}_j, \mathbf{s}) = \frac{\sum_{i\in M_j} C_{ij} s_i}{\sum_{i\in M_j} \max\{0, s_i\}}.$$

3) Centered HITS: is structurally identical but uses the centered vector of student scores instead, specifically,

$$g^{\text{cHITS-M}}(\mathbf{C}_j, \mathbf{s}) = g^{\text{HITS-M}}(\mathbf{C}_j, \mathbf{s} - \bar{s}\mathbf{1}_m).$$

4) Biserial: In the biserial method, $\text{Corr}(\mathbf{C}_j, \mathbf{s}^{(t)})$ can be expressed as a sum over students j, here we simply limit the expression to the terms for students who attempted the question. Let $M_j \subseteq [n]$ be the students who attempted question j and denote with $\mathbf{C}_j^j$ and $\mathbf{s}^j$ the vectors $(C_{ij})_{i\in M_j}$ and $(s_i)_{i\in M_j}$, respectively. Now

$$g^{\text{BSRL-M}}(\mathbf{C}_j, \mathbf{s}) = \text{Corr}(\mathbf{C}_j^j, \mathbf{s}^j) = g^{\text{BSRL}}(\mathbf{C}_j^j, \mathbf{s}^j)$$

5) LogR: Similarly, the logistic update fits a logistic curve through the set of points $(C_{ij}, s_i)_{i\in[m]}$ to determine the weight of question j. When only a subset of the students attempt a question, we fit the curve through $(C_{ij}, s_i)_{i\in M_j}$ instead with the calibration rule

$$g^{\text{LogR-M}}(\mathbf{C}_j, \mathbf{s}) = g^{\text{LogR}}(\mathbf{C}_j^j, \mathbf{s}^j).$$

5 EXPERIMENTAL EVALUATION

Experimental setup. We evaluate the methods on both synthetic data and real-world data. Our synthetic data set is simulated using long-standing models from Item Response Theory; this setup allows us to rigorously test our algorithms under a large variety of conditions, and to compare them to existing IRT maximum likelihood estimators. The real-world data is collected during two classes in which students created and answered multiple-choice questions as part of their regular homeworks. Our algorithms are implemented in Python. The widely used baseline method MIRT [9] is implemented in R. The experiments are run on a server with 2.3GHz CPU and 64GB RAM.

Questions. Our experiments focus on following five questions: (1) How accurate are the algorithms in identifying the true student ranking in the presence of misleading questions? (2) How accurate are the algorithms in the presence of missing responses? (3) How well can the algorithms identify misleading questions? (4) How scalable are the methods? (5) How well do the various methods perform on real data?

Summary of findings. Our key findings are:

(1) *Accuracy & robustness*: When students attempt all questions, both LogR and BSRL are robust to the presence of at least 40% misleading questions, and are at least as accurate as MIRT across all parameter regimes (Figure 5).
(2) *Resilience*: When students answer only a subset of questions, LogR is consistently the most accurate method when questions are easy, and the robust methods are comparable to MIRT when questions are difficult (Figure 6).
(3) *Identification* When questions are easy, LogR and BSRL identify ill-coded questions much more accurately than MIRT; when questions are harder all methods are able to identify misleading questions very accurately (Figure 7).
(4) *Scalability* We empirically confirm that all our methods scale linearly in the number of students whereas MIRT scales quadratically (Figure 8).
(5) On 2 real datasets with approximately 60% missing responses, LogR is the most consistent method (Table 2).

Our experimental results confirm our theoretic analysis and show that LogR and, to a lesser extent, BSRL satisfy all the desiderata set out in Section 1; provide compelling evidence for their use for *transparent and fair* ranking of students in *very large* online classroom settings.

5.1 Experiments on synthetic data

2PL model. Our simulated data is based on the 2PL model from Section 2.1, where each student $i \in [m]$ has an *ability* θ_i, and each question $j \in [n]$ is described by a *difficulty* parameter β_j and a *discrimination* factor α_j. The probability of student i answering question j correctly is given by eq. (1), and the ground truth (GT) rank of the student is the ranking according to the student abilities θ_i. Notice that this model is the one for which maximum likelihood estimation tools like MIRT are designed.

Parameterized experiments in (ω, ν). We have three vectors of problem parameters $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\theta})$ and assume that all parameters are uniformly and independently distributed: $\theta_i \sim U(\theta_{\min}, \theta_{\max})$, $\alpha_j \sim U(0, \alpha_{\max})$, $\beta_j \sim U(\beta_{\min}, \beta_{\max})$. To simplify the comparisons, we parameterize those as follows: First, notice that WLOG we can normalize the parameter space to $\theta_i \sim U(0, 1)$ and thus have 4 free parameters. Then recall from Section 2.1 that the information that a question provides about a student is greatest when the student's ability is close to the question difficulty. We assume that the range of question difficulties are similar to the one of student abilities: $\beta_{\max} = \beta_{\min} + 1$ and then parameterize the relationship between student abilities and question difficulties by $\omega := \beta_{\min}$. In order to analyze the impact of *misleading questions*, we parameterize the fraction of misleading questions by flipping the sign $\alpha_j \leftarrow -\alpha_j$ for a fraction of ν questions. We choose $\alpha_{\max} = 4$ for our experiments.

Table 1 illustrates the expected fraction of correct responses for different values of ω. For $\omega = 0$, a question is answered correctly with 50% probability (in expectation), which is exactly the case when

Table 1: Percentage of correct responses for different values of ω under the 2PL model in the absence of ill-coded questions (averaged over 100 repetitions with $n = m = 100$).

	Parameter regime with $\omega =$				
	-1	-0.5	0	0.5	1
$\mathbb{E}[\mathbb{P}(\text{correct})]$	0.94	0.81	0.49	0.21	0.04

the question difficulty is equal to the student ability. On average, we expect questions to have smaller difficulties than the students' abilities (it is challenging to write difficult and relevant questions), which corresponds to the regimes with $\omega < 0$. This regime also corresponds to the one we have encountered in our classrooms (see) We restrict ourselves to the regimes which most closely mimic the classrooms we have encountered ($\omega \leq 0$) and distinguish between the two extreme regimes in this range: $\omega = 0$, when every student submits a question roughly with difficulty roughly equal to his ability; and $\omega = -2$, when students submit *easy* questions, with difficulty significantly lower than their ability. The results for intermediate values of ω are generally weighted combinations of the results for $\omega \in \{0, 2\}$ and are omitted in the interest of space.

Five methods. We test the five ranking algorithms described in Section 4, logistic regression update rule (LogR), the biserial correlation update rule (BSRL), the hubs and authorities fact-finding algorithm (HITS), the centered version of HITS (cHITS) and as baseline ranking students simply based on the number of questions they answer correctly (AvgSc). For all methods we set $\mathbf{s}_0 = \mathbf{C1}_n$ and add a normalisation step at every iteration so that $||q^{(t)}|| = 1$. This multiplicative scaling prevents numerical difficulties and does not affect the rankings. We compare these methods against the maximum likelihood estimator for the 2PL model computed by the `mirt` [9] package (MIRT). For further details on the maximum likelihood estimation for IRT models and the `mirt` package, see [6, 8, 9, 20].

Accuracy. We measure the accuracy of an output ranking using two metrics: the *normalised Kendall Tau* (KT) distance between two rankings, and the *average normalised displacement* (AD) of a student as compared to the true ranking. The normalised KT distance is the average number of concordant pairs between the two rankings minus the average number of discordant pairs, divided by the number of pairs. The normalised KT score has a value of 1 when the two rankings pefectly coincide, and a value of -1 when one ranking is exactly the reverse of the other. A randomly generated ranking has expected normalised KT distance 0 from the ground truth, since we expect 50% of the pairwise orderings in a random ranking to be correct. The normalised displacement of a student is the distance from his true position, divided by the number of students in the class.

5.1.1 Accuracy & Robustness to misleading questions. The motivation for our work is to develop a method for ranking students that is robust to the presence of misleading questions. We compare the accuracy of our methods while varying the fraction of ill-coded questions $\nu \in \{0, 0.05, 0.1, \ldots, 0.45\}$. Figures 5a and 5b represents the normalized KT distances of our experiments with 100 students and 100 questions, averaged over 500 runs. Figures 5c and 5d shows the average normalized displacement over the same trials.

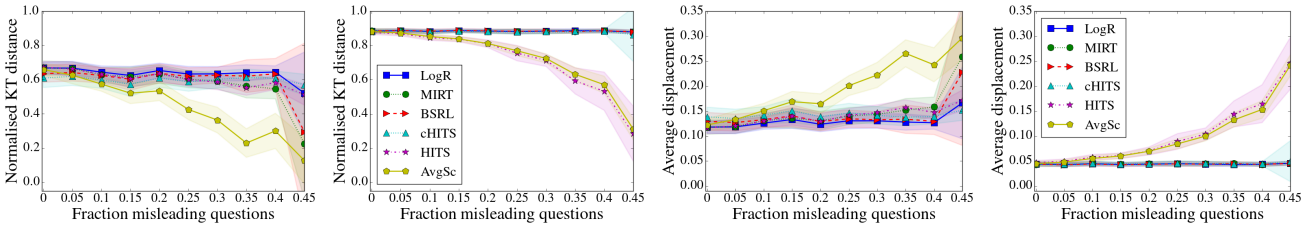
Results. cHITS, BSRL and LogR match MIRT in terms of performance and robustness across both metrics. Interestingly, all three methods exhibit almost identical performance between $\nu = 0$ and $\nu = 0.45$ and show virtually no decrease in performance when the fraction of misleading question increases. This is a stronger observation than implied by our theoretical results: It follows from theorems 6, 9 and 11 that we expect the quality of the ranking to remain unchanged if we use the same starting vector s_0. However, in our experiments $s_0 = \mathbf{C1}$ which depends on ν and does not remain unchanged, yet we still witness remarkable robustness to misleading questions. The baseline methods AvgSc and HITS perform similarly to the others when $\nu = 0$, but degrade significantly as misleading questions are introduced. When questions are easy ($\omega = -2$) and there are many ill-coded questions ($\nu \geq 0.35$) our novel, provably robust methods outperform MIRT.

5.1.2 Resilience to missing responses. As the number of students increases, so does the number of submitted questions. It is therefore unreasonable to expect every student to attempt every question.

First, we fix the number of students and questions to 100 and vary $\omega \in \{0, \pm 1, \pm 2\}$ as before. We vary the fraction of misleading questions $\nu \in \{0, 0.2, 0.4\}$ and the fraction of missing responses between 0 and 90%. Results are averaged over 100 runs of the experiment. Representative results for are visually represented in Figures 6a and 6b. Second, we fix the number of questions answered by each student at $k = 50$ and scale up $m = n$ as would be the case when a student answers 50 questions over the course of a semester independently of the class size. The noise parameter ν is varied as before. We report results for $\nu = 0.2, \omega \in \{-2, 0\}$ in Figures 6c and 6d. We were unable to scale MIRT up to class sizes greater than 800 due to its poor scalability (see fig. 8); for the other methods we report results up to $n = 3\,200$.

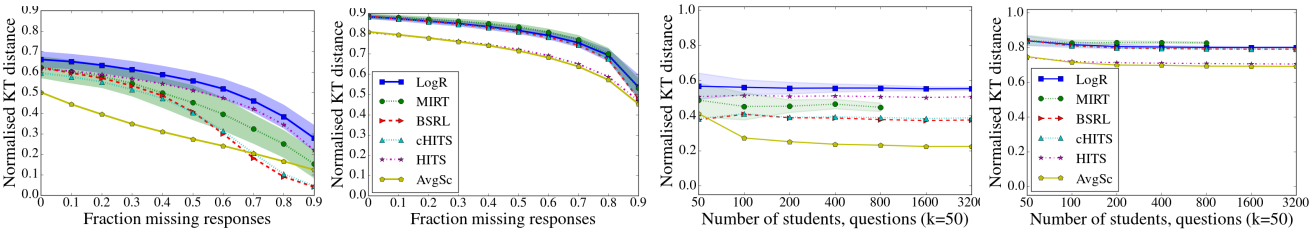
Results. When $\omega = 0$, LogR is largely indistinguishable from MIRT when varying the fraction of missing responses, while MIRT is marginally more accurate when students answer a fixed number of questions. The other two provably robust methods, BSRL and cHITS, match LogR up to roughly 60% missing responses, but lose accuracy thereafter. HITS and AvgSc perform worst. When $\omega = -2$, LogR is significantly better than MIRT in both settings. Interestingly, we also observe clear separation between LogR and the other provably robust methods iterative methods. It is extremely encouraging that when students answer a constant number of questions, accuracy of relative position in class remains fairly constant as the class size increases and the C becomes more sparse. This suggests that ranking students in a very large online class based on their solutions to a small number of potentially misleading questions is a feasible option.

5.1.3 Identification of misleading questions. It is necessary to identifying misleading questions in order to both (*i*) flag and remove such questions from the system, (*ii*) to identify students who may benefit from additional attention, and (*iii*) encourage students to submit high-quality questions. Figure 7 reports the F1-score (harmonic mean between precision and recall) for the classification task of identifying ill-coded questions. The ground truth is taken to be those questions with $a_j < 0$, and a method classifies a question



(a) KT: $\omega = -2$ **(b) KT:** $\omega = 0$ **(c) AD:** $\omega = -2$ **(d) AD:** $\omega = 0$

Figure 5: Section 5.1.1: Robustness as ν changes. (a, b) show the normalized KT score (higher is better), while (c, d) show the normalized average displacement (lower is better). For $\nu \leq 0.35$, LogR, BSRL and MIRT are robust to misleading responses; HITS and AvgSc are not. When there are many easy but misleading questions ($\nu = 0.4$, $\omega = -2$) LogR, BSRL and cHITS are more robust than MIRT according to both metrics.



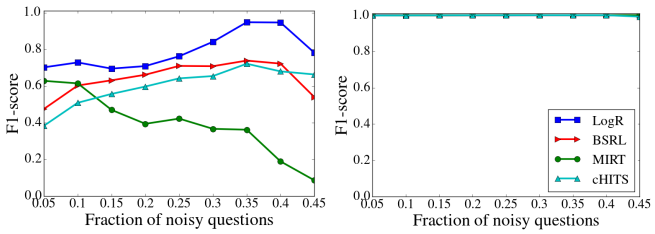
(a) KT: $\nu = 0.2$, $\omega = -2$ **(b) KT:** $\nu = 0.2$, $\omega = 0$ **(c) KT:** $\nu = 0.2$, $\omega = -2$ **(d) KT:** $\nu = 0.2$, $\omega = 0$

Figure 6: Section 5.1.2: Resilience to missing responses. (a,b): The fraction of missing responses increases while $m = n = 100$ stays constant. (c,d): Students answer a constant $k = 50$ questions while $n = m$ increases from 50 to 3 200. The standard deviations of LogR and MIRT are indicated to emphasize the significant improvement in accuracy we obtain with LogR over IRT maximum likelihood methods when questions are easy.

correctly if, for such a question, $q_j < 0$. Recall neither AvgSc nor HITS can flag misleading questions and we thus only show results for LogR, BSRL, cHITS and MIRT.

Results. When $\omega = 0$, all methods identify misleading questions almost perfectly. When $\omega = -2$ the iterative methods, and in particular LogR, have significantly higher f1 scores than MIRT. A closer investigation reveals that while MIRT's precision remains comparable to that of LogR, its recall drops precipitously in the interval $[0.1, 0.3]$.

5.1.4 Scalability. We now evaluate the scalability of our algorithms when increasing the number of students m. We consider two



(a) F1-score: $\omega = -2$ **(b) F1-score:** $\omega = -0$

Figure 7: Section 5.1.3: F1-score for identifying misleading questions as the fraction ν of such questions changes. For moderately difficult questions all methods perform similar (b), however, when questions becomes easier, LogR performs best (a).

variants: (*i*) We fix $n = 100$ questions, $\omega = 0$, $\nu = 0.2$, and increase m between 100 and 50 000.

Results. Figure 8 summarizes the runtimes in a log-log plot. We see that all our algorithms are *linear* in the number of students m. We also see that MIRT is quadratic for fixed number of questions n. Each point is the average over 10 runs of the algorithm, with the exception of MIRT, where due to extreme runtimes (≈2 hour for $m = 12\,800$), we only perform one run for $m \geq 100$ and only increase m up to 25 600.

As expected, the baseline of ranking students based on the number of correct responses is fastest, followed by HITS, cHITS, BSRL, then LogR and finally MIRT. Our novel methods scale linearly in the number of students and are orders of magnitude faster than MIRT. The difference in speed between BSRL and LogR is likely due to the fact that in BSRL there is a simple closed form expression for computing question weights, while in LogR these weights are the result of an optimization program for which a closed form expression is not available.

All our methods are linear and significantly faster than MIRT (LogR needed sec for 100k students whereas MIRT needed h for 20k students). As discussed in [19], computing the marginal maximum likelihood for the IRT generative models involves intractable integrals. When these integrals are evaluated using Gaussian quadrature [7], the number of evaluation points increase dramatically in the number of latent trait parameters to be estimated. Adaptive quadrature methods [36, 37] reduce the number of points to evaluate, however, we do not expect these methods to scale gracefully with the number of students and questions. Our experiments

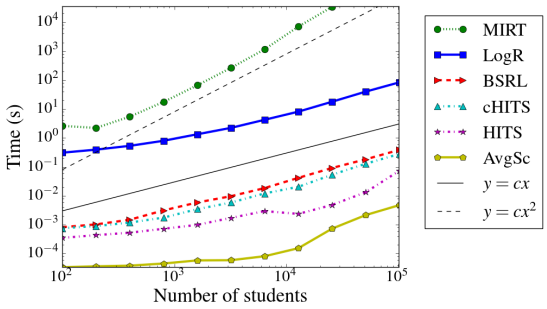


Figure 8: Runtime as the number of students increase while the number of questions $n = 100$ remains constant: The iterative methods scale linearly in the number of students.

Table 2: Results on real data (32 students and 223 question)

	KT	AD
LogR	**0.535**	**0.167**
MIRT	0.424	0.202

suggest a growth rate quadratic in the number of students even when students are described using a single latent variable (their one-dimensional ability).

5.2 Real data

MIRT is currently in use in several real-world settings including at the Khan Academy and for grading the Student Aptitude Tests (SATs). In our experiments with synthetic data, LogR performed as well, and even better than MIRT across a variety of metrics and parameter regimes. We now compare these two methods on two real-world data sets from our classes. Those were pilots on campus and are thus smaller than our intended application scenario.

Setup. We collected a dataset of student-sourced questions and responses during the teaching of a course called ANONYMIZED. As part of regular homeworks, students were asked to submit multiple-choice questions that relate to that week's class material, and also answer such questions created by fellow students (recall Theorem 1). This testset contains 32 students and 223 contributed questions. The average student answered approximately 40% of the questions and the quality of the questions, and whether they are misleading is not known. We use the final class scores of the students to compute their ground truth ranking.

Results. The performance of the different algorithms on this dataset is summarized in table 2. LogR ranks the students much closer to their true ranking than MIRT. The difference of roughly 3.5% in the average normalized displacement between LogR and MIRT means that, on a class size of 1 000, we expect LogR to rank students on average 35 positions closer to their true positions than MIRT.

Though additional experiments with larger datasets should be conducted, the experiments on both the simulated and real data suggest our LogR method has great potential to be an accurate and scalable technique for grading and ranking students which is provably robust to ill-coded questions.

6 CONCLUSIONS

We investigated the problem of *ranking students in the presence of misleading questions*. We identified several properties that make ranking methods robust to the presence of such questions, suggest several algorithms that fulfill these properties, and experimentally confirm their usefulness. In particular, our new iterative update methods based on biserial correlation and on logistic regression scale linearly in the number of students while matching the performance of current state-of-the art parameter estimation methods for IRT models (which are *not transparent* to students and *do not scale* to large class sizes). Our theoretical analysis and experiments thus suggest that these methods have great potential for grading students in very large class settings in a way that is (1) accurate, (2) robust to misleading questions, (3) resilient to missing responses, (4) linearly scalable, (5) transparent and fair.

REFERENCES

[1] ErlingB. Andersen. 1973. A goodness of fit test for the rasch model. *Psychometrika* 38, 1 (1973), 123–140. https://doi.org/10.1007/BF02291180

[2] Erling B Andersen. 1985. Estimating latent correlations between repeated testings. *Psychometrika* 50, 1 (1985), 3–16.

[3] Reid Andersen, Christian Borgs, Jennifer Chayes, Uriel Feige, Abraham Flaxman, Adam Kalai, Vahab Mirrokni, and Moshe Tennenholtz. 2008. Trust-based recommendation systems: an axiomatic approach. In *WWW*. 199–208.

[4] Yoram Bachrach, Thore Graepel, Tom Minka, and John Guiver. 2012. How To Grade a Test Without Knowing the Answers - A Bayesian Graphical Model for Adaptive Crowdsourcing and Aptitude Testing. In *ICML*. 1183–1190.

[5] Frank K. Baker and Seock-Ho Kim. 2004. *Item Response Theory: Parameter Estimation techniques (2nd Ed)*. Marcel Dekker Inc.

[6] R Darrell Bock. 1972. Estimating item parameters and latent ability when responses are scored in two or more nominal categories. *Psychometrika* 37, 1 (1972), 29–51.

[7] R Darrell Bock and Murray Aitkin. 1981. Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. *Psychometrika* 46, 4 (1981), 443–459.

[8] Li Cai and David Thissen. 2014. Modern approaches to parameter estimation in item response theory. *Handbook of Item Response Theory Modeling: Applications to Typical Performance Assessment* (2014), 41–59.

[9] R Philip Chalmers et al. 2012. mirt: A multidimensional item response theory package for the R environment. *Journal of Statistical Software* 48, 6 (2012), 1–29.

[10] M.T.H. Chi, M. W. Lewis, P. Reimann, and R. Glaser. 1989. Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science* 13 (1989), 145–182.

[11] P. Dawid, A. M. Skene, A. P. Dawidt, and A. M. Skene. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics* 1 (1979), 20–28.

[12] Rafael Jaime De Ayala. 2013. *The theory and practice of item response theory*. Guilford Publications.

[13] Jorge Diez, Oscar Luaces, Amparo Alonso-Betanzos, Alicia Troncoso, and Antonio Bahamonde. 2013. Peer Assessment in MOOCs Using Preference Learning via Matrix Factorization. In *NIPS Workshop on Data Driven Education*.

[14] Susan E Embretson. 1991. A multidimensional latent trait model for measuring learning and change. *Psychometrika* 56, 3 (1991), 495–515.

[15] Tim Finin, Will Murnane, Anand Karandikar, Nicholas Keller, Justin Martineau, and Mark Dredze. 2010. Annotating named entities in Twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*. ACL, 80–88.

[16] Michael J Franklin, Donald Kossmann, Tim Kraska, Sukriti Ramesh, and Reynold Xin. 2011. CrowdDB: answering queries with crowdsourcing. In *SIGMOD*. 61–72.

[17] Alban Galland, Serge Abiteboul, Amélie Marian, and Pierre Senellart. 2010. Corroborating information from disagreeing views. In *WSDM*. 131–140.

[18] Manish Gupta and Jiawei Han. 2011. Heterogeneous network-based trust analysis: a survey. *ACM SIGKDD Explorations Newsletter* 13, 1 (2011), 54–71.

[19] Minjeong Jeon and Frank Rijmen. 2014. Recent developments in maximum likelihood estimation of MTMM models for categorical data. *Frontiers in psychology* 5 (2014).

[20] Matthew S Johnson et al. 2007. Marginal maximum likelihood estimation of item response models in R. *Journal of Statistical Software* 20, 10 (2007), 1–24.

[21] Sepandar D Kamvar, Mario T Schlosser, and Hector Garcia-Molina. 2003. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the 12th international conference on World Wide Web*. ACM, 640–651.

[22] Cristobald de Kerchove and Paul Van Dooren. 2008. The pagetrust algorithm: How to rank web pages when negative links are allowed?. In *Proceedings of the 2008 SIAM International Conference on Data Mining*. SIAM, 346–352.

[23] Neal M Kingston and Neil J Dorans. 1982. The feasibility of using item response theory as a psychometric model for the GRE Aptitude Test. *ETS Research Report Series* 1982, 1 (1982).

[24] Jon M Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *JACM* 46, 5 (1999), 604–632.

[25] Igor Labutov, Kelvin Luu, Thorsten Joachims, and Hod Lipson. 2014. Peer Mediated Testing. In *KDD workshop on Data Mining for Educational Assessment and Feedback (ASSESS 2014)*.

[26] Balaji Lakshminarayanan and Yee Whye Teh. 2013. Inferring ground truth from multi-annotator ordinal data: a probabilistic approach. (2013).

[27] Andrew S. Lan, Andrew E. Waters, Christoph Studer, and Richard G. Baraniuk. 2014. Sparse Factor Analysis for Learning and Content Analytics. *JMLR* 15, 1 (2014), 1959–2008. http://dl.acm.org/citation.cfm?id=2627435.2670314

[28] Christopher H. Lin, Mausam, and Daniel S. Weld. 2012. Crowdsourcing Control: Moving Beyond Multiple Choice. In *UAI*. 491–500.

[29] Frederic M Lord, Melvin R Novick, and Allan Birnbaum. 1968. Statistical theories of mental test scores. (1968).

[30] Piotr Mitros, Anant Agarwal, and Vik Paruchuri. 2014. Assessment in Digital At-scale Learning Environments: MOOCs and Technology to Advance Learning and Learning Research (Ubiquity Symposium). *Ubiquity* 2014, April, Article 2 (2014), 9 pages. https://doi.org/10.1145/2591795

[31] Jeff Pasternack and Dan Roth. 2010. Knowing what to Believe (when you already know something). In *COLING*. 877–885.

[32] J. Pasternack and D. Roth. 2011. Generalized Fact-Finding. In *WWW*. 99–101. http://cogcomp.cs.illinois.edu/papers/PasternackRo11.pdf

[33] J. Pasternack and D. Roth. 2013. Latent Credibility Analysis. In *WWW*. 1009–1021. http://cogcomp.cs.illinois.edu/papers/PasternackRo13.pdf

[34] Jeffrey Pasternack, Dan Roth, and V.G. Vinod Vydiswaran. 2013. Information Trustworthiness. AAAI Tutorial. (2013).

[35] Chris Piech, Jonathan Huang, Zhanghao Chen, Chuong Do, Andrew Ng, and Daphne Koller. 2013. Tuned Models of Peer Assessment in MOOCs. In *Proceedings of Sixth International Conference of MIT's Learning International Networks Consortium*.

[36] José C Pinheiro and Douglas M Bates. 1995. Approximations to the log-likelihood function in the nonlinear mixed-effects model. *Journal of computational and Graphical Statistics* 4, 1 (1995), 12–35.

[37] Sophia Rabe-Hesketh, Anders Skrondal, Andrew Pickles, et al. 2002. Reliable estimation of generalized linear mixed models using adaptive quadrature. *The Stata Journal* 2, 1 (2002), 1–21.

[38] Dimitris Rizopoulos. 2006. ltm: An R package for latent variable modeling and item response theory analyses. *Journal of statistical software* 17, 5 (2006), 1–25.

[39] Y Saad. 2003. *Iterative methods for sparse linear systems* (2nd ed ed.). SIAM.

[40] Fumiko Samejima. 1996. Evaluation of mathematical models for ordered polychotomous responses. *Behaviormetrika* 23, 1 (1996), 17–35.

[41] Nihar B Shah, Sivaraman Balakrishnan, and Martin J Wainwright. 2016. A permutation-based model for crowd labeling: Optimal estimation and robustness. *arXiv preprint arXiv:1606.09632* (2016).

[42] Nihar B. Shah, Joseph Bradley, Sivaraman Balakrishnan, Abhay Parekh, Kannan Ramchandran, and Martin J. Wainwright. 2014. Some Scaling Laws for MOOC Assessments. In *ACM KDD Workshop on Data Mining for Educational Assessment and Feedback (ASSESS)*.

[43] Alexey Tarasov, Sarah Jane Delany, and Charlie Cullen. 2010. Using crowdsourcing for labelling emotional speech assets. *W3C EmotionML Workshop* (2010).

[44] Wim J van der Linden. 2016. *Handbook of Item Response Theory, Volume One: Models.* CRC Press.

[45] Wim J van der Linden. 2017. *Handbook of Item Response Theory, Volume Two: Statistical Tools.* CRC Press.

[46] Peter Welinder and Pietro Perona. 2010. Online crowdsourcing: rating annotators and obtaining cost-effective labels. In *CVPRW*. IEEE, 25–32.

[47] Xiaoxin Yin, Jiawei Han, and Philip S Yu. 2008. Truth discovery with multiple conflicting information providers on the web. *TKDE* 20, 6 (2008), 796–808.

[48] Bo Zhao, Benjamin I. P. Rubinstein, Jim Gemmell, and Jiawei Han. 2012. A Bayesian Approach to Discovering Truth from Conflicting Sources for Data Integration. *PVLDB* 5, 6 (2012), 550–561.

[49] Yudian Zheng, Guoliang Li, Yuanbing Li, Caihua Shan, and Reynold Cheng. 2017. Truth Inference in Crowdsourcing: Is the Problem Solved? *PVLDB* 10, 5 (2017), 541–552. https://doi.org/10.14778/3055540.3055547

A NOMENCLATURE

m	number of students
n	number of questions
$\mathbf{C}$	$(m \times n)$ binary matrix of test results, $C_{ij} = 1$ if student i answers question j correctly
$\mathbf{C}_j$	j-th column of $\mathbf{C}$ (short form for $\mathbf{C}_{:j}$)
a_j	discrimination factor of question j
b_j	difficulty of question j
θ_i	ability of student i
$\mathbf{q}$	n-dimensional vector of question weights
$\mathbf{s}$	m-dimensional vector of student scores
$\bar{s}$	average student score
f_j	facility of question$j$: percentage of students who got it right
J	subset of questions that are misleading ($J \subseteq [n]$)
k	number of questions answered by each student (often $k = n$)

B PROOFS FROM §4

We restate and prove the results stated without proof in section 4.

B.1 cHITS is a scaled eigenvalue calculation

Proof. First observe $\mathbf{W}\mathbf{W}^\intercal = \mathbf{W} = \mathbf{W}^\intercal$, thus (6) is equivalent to

$$\mathbf{s}^{(t)} = (\mathbf{C}\mathbf{C}^\intercal\mathbf{W}\mathbf{W}^\intercal)^t\mathbf{s}_0.$$

Now observe that by simply reordering the parentheses, we have

$$\begin{aligned}(\mathbf{C}\mathbf{C}^\intercal\mathbf{W}\mathbf{W}^\intercal)^t &= \mathbf{C}\mathbf{C}^\intercal\mathbf{W}(\mathbf{W}^\intercal\mathbf{C}\mathbf{C}^\intercal\mathbf{W})^{t-1}\mathbf{W}^\intercal \\ &= \mathbf{C}\mathbf{C}^\intercal(\mathbf{W}^\intercal\mathbf{C}\mathbf{C}^\intercal\mathbf{W})^{t-1}.\end{aligned}$$

Let $\mathbf{e}_i$, $i \in [m]$ be an orthogonal basis of eigenvectors of $\mathbf{W}^\intercal\mathbf{C}\mathbf{C}^\intercal\mathbf{W}$, with corresponding eigenvalues λ_i in descending order. Then $\mathbf{s}_0 = \sum_{i=1}^m \mathbf{e}_i^\intercal\mathbf{s}_0\mathbf{e}_i$, hence

$$\mathbf{s}^{(t)} = \mathbf{C}\mathbf{C}^\intercal(\mathbf{W}^\intercal\mathbf{C}\mathbf{C}^\intercal\mathbf{W})^{t-1}\mathbf{s}_0 = \sum_{i=1}^m \lambda_i^{t-1}\mathbf{e}_i^\intercal\mathbf{s}_0\mathbf{C}\mathbf{C}^\intercal\mathbf{e}_i.$$

Therefore, letting λ be the eigenvalue corresponding to the vector $\mathbf{e}$ from the statement of the theorem,

$$\begin{aligned}\frac{\mathbf{s}^{(t)}}{\|\mathbf{s}^{(t)}\|} &= \frac{\sum_{i=1}^m \lambda_i^{t-1}\mathbf{e}_i^\intercal\mathbf{s}_0\mathbf{C}\mathbf{C}^\intercal\mathbf{e}_i}{\left\|\sum_{i=1}^m \lambda_i^{t-1}\mathbf{e}_i^\intercal\mathbf{s}_0\mathbf{C}\mathbf{C}^\intercal\mathbf{e}_i\right\|} \\ &= \frac{\sum_{i=1}^m (\lambda_i/\lambda)^{t-1}\mathbf{e}_i^\intercal\mathbf{s}_0\mathbf{C}\mathbf{C}^\intercal\mathbf{e}_i}{\left\|\sum_{i=1}^m (\lambda_i/\lambda)^{t-1}\mathbf{e}_i^\intercal\mathbf{s}_0\mathbf{C}\mathbf{C}^\intercal\mathbf{e}_i\right\|}.\end{aligned}$$

Considering first the numerator, if $\lambda_i > \lambda$, by assumption we have $\mathbf{e}_i^\intercal\mathbf{s}_0 = 0$, so these terms disappear. For $\lambda_i < \lambda$, $(\lambda_i/\lambda)^{t-1} \to 0$ as $t \to \infty$. Thus the numerator of $\mathbf{s}^{(t)}$ converges to a vector in the eigenspace of λ, multiplied by $\mathbf{C}\mathbf{C}^\top$, and by continuity the denominator simply normalises the resulting vector. This gives us our result. □

B.2 Biserial update rule

Proof. The proof of this is similar to that of Theorem 7 by observing that, since $\mathbf{W}\mathbf{W}^\intercal = \mathbf{W} = \mathbf{W}^\intercal$, we have

$$\mathbf{s}^{(t)} = (\mathbf{C}\mathbf{D}_C\mathbf{C}^\intercal\mathbf{W}\mathbf{W}^\intercal)^t\mathbf{s}_0$$

and

$$(\mathbf{C}\mathbf{D}_C\mathbf{C}^\intercal\mathbf{W}\mathbf{W}^\intercal)^t = \mathbf{C}\mathbf{D}_C\mathbf{C}^\intercal(\mathbf{W}^\intercal\mathbf{C}\mathbf{D}_C\mathbf{C}^\intercal\mathbf{W})^{t-1}.$$

The proof now follows in the exact same fashion as in that of Theorem 7. □

B.3 Logistic update rule

We show that the LogR update satisfies translation invariance (4) and antisymmetry (5).

Lemma 14 (Translation invariance of LogR). *Let* $\mathbf{c} \in \{0,1\}^m$, $\mathbf{s} \in \mathbb{R}^m$ *and* $\alpha \in \mathbb{R}$ *be arbitrary. Then*

$$g^{LogR}(\mathbf{c}, \mathbf{s} + \alpha\mathbf{1}_m) = g^{LogR}(\mathbf{c}, \mathbf{s}).$$

Proof. Let

$$a^+, b^+ = \arg\max_{a,b} \sum_{i=1}^{m} [c_i(a(s_i+\alpha)-b) - \log\left(1+\exp(a(s_i+\alpha)-b)\right)], \text{ and}$$

$$a^*, b^* = \arg\max_{a,b} \sum_{i=1}^{m} \left[c_i(as_i-b) - \ln\left(1+\exp(as_i-b)\right)\right].$$

We will show that $a^+ = a^*$ and $b^+ = b^* + \alpha a^+$.

Substituting $a' = a, b' = b + \alpha a$ into the former objective gives us latter objective, so the former problem upper bounds the latter problem. Substituting $a' = a, b' = b - \alpha a$ into the latter objective gives us the former objective, thus the latter problem upper bounds the former problem. Therefore the two problems are equivalent, and any optimal solution to the former problem can be constructed into a solution for the latter problem with the same a-term. □

Lemma 15 (Antisymmetry of LogR). *Let $\mathbf{c} \in \{0,1\}^m$ be a fixed binary vector, and $\mathbf{s} \in \mathbb{R}^m$ be a fixed real vector. Then*

$$g^{LogR}(\mathbf{c}, \mathbf{s}) = -g^{LogR}(\mathbf{1}-\mathbf{c}, \mathbf{s}).$$

Proof. Observe that

$$\begin{aligned}
&\arg\max_{a,b} \sum_{i=1}^{m} \left[(1-c_i)(as_i-b) - \ln\left(1+\exp(as_i-b)\right)\right] \\
&= \arg\max_{a,b} \sum_{i=1}^{m} \left[c_i(-as_i+b) + \ln\left(\frac{\exp(as_i-b)}{1+\exp(as_i-b)}\right)\right] \\
&= \arg\max_{a,b} \sum_{i=1}^{m} \left[c_i(-as_i+b) - \ln\left(1+\exp(-as_i+b)\right)\right] \\
&= -\arg\max_{a,b} \sum_{i=1}^{m} \left[c_i(as_i-b) - \ln\left(1+\exp(as_i-b)\right)\right].
\end{aligned}$$

□