# Parameter Estimation in IRT Models using Matrix Completion with Applications to Intelligent Tutoring Systems

Jack Finkel
Student Collaborator: Haonan Sun
Industry Collaborator: Vivek Varshney
Mentors: Laura Niss, Prof. Ambuj Tewari

April 23, 2020

**Abstract**

We explore the feasibility of estimating question difficulty and student ability parameters in item response theory (IRT) models. We have real data from an educational technology company that provides online tutoring to high school students for university entrance examinations in India. Since the real data is sparse, we also study the usefulness of matrix completion methods to create a filled matrix that can then be used for parameter estimation. Our research is motivated by the desire to provide adaptive question recommendation to improve student learning in a real world setting where there are large amounts of missing data.

Moreover, we attempt to illuminate two practical requirements for accurate estimation of IRT model parameters from real world data: the allowable size of the data and amount of missing elements in the data. We also put the findings from our data analysis in the context of our collaboration with industry partners who want to use these analyses to inform the future design of their question recommendation engine.

**Keywords:** Item Response Theory; Matrix Completion; Intelligent Tutoring System

## 1 Introduction

The motivation for this paper is to ultimately provide a question recommendation algorithm that selects questions to optimize concept mastery by students. We collaborated with SpeedLabs who provided data from their online tutoring system. Students, who are expected to have learned the material in class, use SpeedLabs' platform to gain further practice in order to do better on their tests scores, often in preparation for a university entrance exam. SpeedLabs' current platform structure is to give students the opportunity to log in to sessions, which they can log out of anytime, and then randomly give students a series of questions related to a selected topic. Using the data available from SpeedLabs, we want to develop question selection methods to replace randomly selecting questions, that if implemented into SpeedLabs' platform, would reliably improve the level of concept mastery among its users.

To accomplish this, we wish to develop an algorithm that will select questions based on question parameters and a student's ability that optimizes on student ability improvement. However, a vital first step to creating this algorithm is to model each students' ability level and questions' difficulty parameters. Our initial goal as requested by our industry partners, was to estimate a zone of proximal learning. This zone in theory is where questions are not too easy or too difficult and where a student is capable of learning. Once we model student ability and question parameters, it may be possible to find the zone of proximal learning. We choose the 3-parameter Item Response Theory (IRT) paradigm to model the probability a student will answer a question correctly given a students ability and the item parameters of difficulty, discrimination, and guessing. The IRT model is widely used in computer adaptive testing, typically to estimate a student's ability. We chose this model over others, such as cognitive diagnostic modeling because the questions in our dataset are labeled generally, such as related to a section or a chapter in a textbook. Therefore we only consider a general student ability and cannot test for more fine grained skills.

However, unlike other situations that the IRT model is used for, SpeedLabs' real world data is quite sparse. There are significantly more questions than students. Therefore, many students do not answer most questions. While there are implementations that allow us to use the IRT model with missing data, they seem to fail to handle missing values (Johnson, 2007). To address this problem, we apply the Soft-Impute matrix completion algorithm to impute the data (Mazumder, Hastie, and Tibshirani, 2010) before estimating the parameters. We choose the soft impute matrix completion algorithm (Soft-Impute) due to its convenience and ease-of-use. We use simulated data to analyze the efficacy of this method before applying it to real data.

Given our methods for estimating item parameters, our next question is whether the estimated IRT parameters of the questions used in the industry data indicate that the questions can be used in question recommendation. We see that when using both both a complete subset and a half-complete subset of the real data, there are negative estimates for the discriminating parameter of many of the questions. Questions with an estimated negative discriminatory value should not be used in the question recommendation, since the associated item characteristic curve of that question would not be positively monotonic. This would mean that a student with higher ability level is less likely to answer the question correctly.

The scope of this paper is to analyze the industry data we have by estimating the parameters in simple IRT models with missing data imputed using matrix completion methods. We hope that future research can use our preliminary results to best select questions for students. Specifically, we analyze how the IRT estimates differ with complete data versus half-complete data. In order to further contextualize our estimates of model parameters obtained from industry data, we test how accurate our estimates are with simulated data generated with item and ability parameters to match what we observe in the industry data. We conclude that using a matrix completion method for imputing half-complete testing data before estimating the IRT parameters is indeed a reasonable method for estimating IRT parameters with sparse testing data. We also conclude that having more of both students and questions in the data is useful for more accurate estimates of the item and ability parameters, and that a large student to question ratio is optimal.

# 2 Data

This section will give an overview of the industry data, including where it is from, how it was collected and compiled, and what subsets of the industry data we use in our model.

## 2.1 Industry Data Overview

The industry data we collected is the result of giving a pool of students a varying number of questions to answer from a question bank. This data was collected and given to us through a collaboration with the company SpeedLabs, an online learning and practice platform (*SpeedLabs - Online Learning Platform for Exam Prep* 2020). A user starts a session by picking a topic to be tested on, and are given a series of randomly selected questions without replacement. Students are allowed a second chance, they can skip questions, review related textbook material, or ask a tutor. The general topics include math, physics, chemistry, or general science. The session ends when the student chooses to leave. No student has restrictions or caps on how many questions he or she should or could answer.

We have received and analyzed testing data from three main batches: the first collected on October 12th, 2018, the second on December 16th, 2018, and the third on May 30th, 2019. All testing data was concatenated together before doing any analysis. We are able to combine the testing data, since it uses the same user ids and question ids to reference the same users and questions. Furthermore, students were given questions using random sampling throughout all three sets of testing data, which is important for maintaining independence. Lastly, since the IRT estimates remain invariant across populations and the ability parameters remain invariant across tests (Zhang and Chang, 2016), we are able to add more data, when estimating the parameters of our model. For each question answered by each student, a record was made containing the id of the student, the id of the question, and whether the question is answered correctly. There is other information, such as the amount of time users spend on question they answer, but it is not used in this paper.

## 2.2 Subsets of Industry Data Used for Parameter Estimation

This paper focuses on estimating the IRT parameters with testing data that is either complete or half-complete, where the questions are in a single subject. However, the industry data as a whole is too sparse for these methods. Even though we have 169,642 total records of 223 users answering variable sample sizes from a set of 19,365 questions, the average number of times a question was answered was just 8.7 times. Furthermore, as indicated, the user is allowed to log out or quit the session at any time. This results in large disparities in amount of participation between users, with some answering very few questions compared to others. Even though the industry data is sparse, we find two subsets of testing data that meet our requirement for complete or half-complete data: a complete subset of the data in the math category with 11 students and 98 questions, and a half-complete subset of the data in the math category with 65 students and 241 questions. Both these subsets of the industry data were the largest subsets we could find that met the specification of being complete or half-complete.

Please see Section 8.1 for the initial visualizations done before implementing the IRT model.

# 3 Related Works

The focus of this paper is how to define question difficulty and student ability, using real data. There are many different psychometric paradigms and variations to choose from to model question difficulty and student ability, but we chose the 3-parameter Item Response Theory (IRT) model. In the IRT model, each question, $q_i$, is modeled with three 3 parameters, $(a_i, b_i, c_i)$, and each student is modeled with a unidimensional ability parameter, $\theta$, for a particular topic. The question parameters, or item parameters, are defined as follows:

- $a_i$ - discrimination: maximum change in probability that the student gets the answer correct, given an increase in $\theta$.
- $b_i$ - difficulty: the value of $\theta$, where the change in probability that the student gets the answer correct, given an increase in $\theta$ is maximized
- $c_i$ - chance: the chance the student gets the question right, if the student randomly guesses. This is set to 0.25 for the real data set to mimic the four choice answer set.

For each question $i$, with corresponding item parameters $(a_i, b_i, c_i)$, the probability that a student with the ability parameter $\theta$ gives a correct response, $p_i(\theta)$, is given by

$$p_i(\theta) = c_i + \frac{1 - c_i}{1 + e^{-a_i(\theta - b_i)}}$$

The item characteristic curve (ICC), the relationship between $\theta$ and $p_i(\theta)$, is shown in Figure 1. In this figure, the item parameters are set to (1,0,0.25).

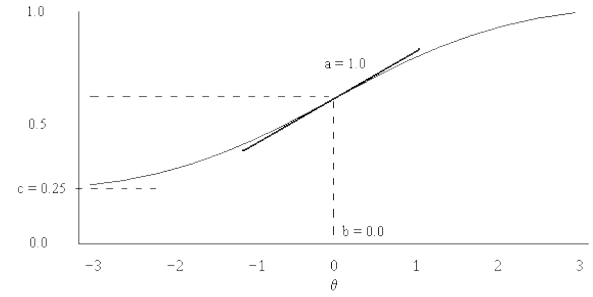


Figure 1: Item Characteristic Curve (ICC) with item parameters set to (1,0,0.25) (Iulus Ascanius, 2008)

An important property of the IRT model is that the item parameters in the IRT model remain invariant across populations and the ability parameters remain invariant across tests. This enables us to put the item parameters on the same scale, regardless of which students answered the corresponding questions. Similarly, it also allows us to put the ability parameters on the same scale, regardless of which questions the students answered (Zhang and Chang, 2016). Therefore, we could add more data to be added to the model, without affecting the scale of the parameters.

The IRT has several benefits over other models. The first reason is that the IRT model is simpler compared to other models since it uses an unidimensional ability parameter. Other models, such as the cognitive diagnostic models (CDMs), do not provide an unilateral ability parameter, but rather an number of proficiency parameters that each measures a particular student's ability to perform in different types of problems or subtopics (Sinharay and Almond, 2007). Furthermore, the IRT model uses an unidimensional ability parameter, it contains a monotonic ICC, meaning that if a student has a better ability, they are more likely to answer a given question correctly, regardless of the suptopic of the question. Therefore, with the IRT model, we can determine how well students do in a particular subject and select particular questions that challenge the student in the subject. That being said, future research should explore the use of more complex models including CDMs.

We also choose the IRT models since they are widely used in psychometric research and industry. The IRT models are used in such settings as the GRE test (Carlson and Davier, 2013) , and they are considered a standard psychometric paradigm. Furthermore, there are implementations of the IRT model in R which allow us to model our questions easily.

While there are many variations of the IRT paradigm, we chose to work with the 3-parameter IRT model, mainly because of the structure of SpeedLabs questions. There are three main IRT models to consider when working with dichotomous data: the 2-parameter IRT model, the Rasch IRT model, and the 3-parameter IRT model. Unlike the 3-parameter IRT model, the 2-parameter IRT model contains a discrimination and difficulty parameter but no chance parameter. The Rasch IRT model also contains a difficulty parameter but no chance parameter and assumes all questions have the same discrimination level (Johnson, 2007). However, since all questions are multiple choice with four answers, we want to be able to set a minimum probability of every student answering a question correct, or the chance parameter ($c$), with the value 0.25. For some questions in the industry data, 0.25 may be an underestimate, as some answers for those questions may be obviously incorrect to all students, so the true value could be closer to approximately 0.5. However, including the guessing parameter of 0.25 would better represent the properties of SpeedLabs' questions, than the 2-parameter IRT model or the Rasch IRT model, both which do not contain the chance parameter.

# 4 Methods

We use the 3-parameter logistic IRT model to characterize an individual's probability of answering a question correctly with regard to their ability.. The vector $(a_i, b_i, c_i)$ is the item parameter for question $i$. For an individual with ability $\theta$ and the item parameters of question $i$, the IRT model characterizes the probability distribution function of the individual answering a question correctly or incorrectly.

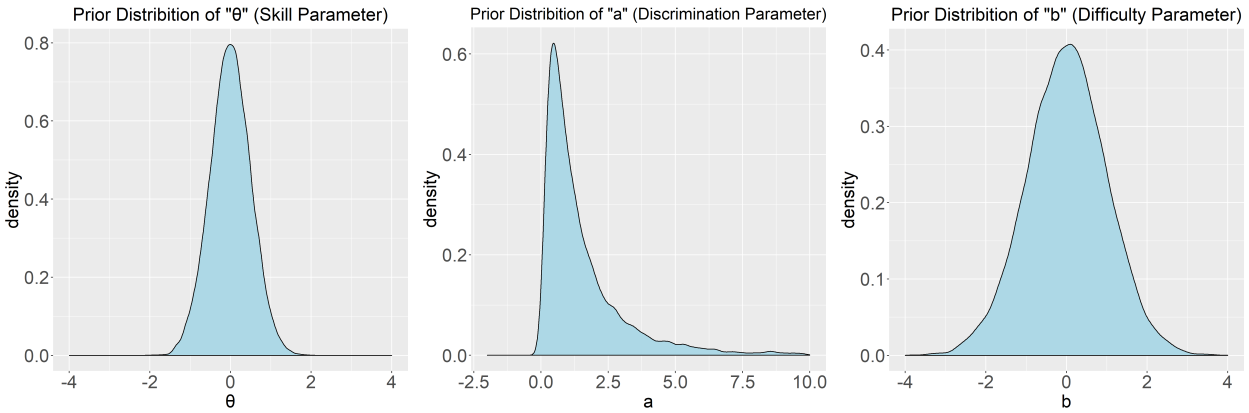
To estimate these parameters, we exclusively use R's Latent Trait Models (ltm) package. To estimate the item parameters, we use ltm's Binbaum's Three Parameter Model (tpm) function, which is the implementation of the three parameter version of the IRT model. The tpm function estimates the item parameters with the maximum likelihood estimates, since they are a standard way to estimate the IRT parameters (Bock and Aitkin, 1981). There are several implementations of different optimization algorithms to find the maximum likelihood, but we use the BFGS algorithm to find the maximum likelihood estimates since it is the default method of the tpm function. To estimate the ability parameter, we use the ltm's Factor Scores (factor.scores) function to estimate the ability parameters. We chose to use factor.scores' default estimates of the ability parameters, the Empirical Bayes estimates.

In the ITS setting, the response matrix can be sparse when there is a large question bank relative to the number of users, or new questions are continually added. However, current implementations of estimating item and ability parameters are unable to handle missing data (Johnson, 2007). Furthermore, since our industry data is quite sparse, we would only be able to use a very small subset of the data, in which every student answers every question to estimate our IRT parameters. However, if we impute the missing data by matrix completion methods before estimating ability and item parameters, we can use larger subsets of the industry data in our IRT model.

While there are many other matrix completion algorithms, we use one of the proximal gradient methods, the Soft-Impute algorithm, for our problem of sparse data. Since latent variable models can be shown to have an intrinsic low-rank structure (Udell and Townsend, 2018), it is reasonable to use low-rank matrix completion methods, like the Soft-Impute method. We specifically choose Soft-Impute as our proximal gradient method, since it has an implemented R-package (softImpute) and has good training errors, test errors, and quicker run-times, when compared to other matrix completion algorithms (Mazumder, Hastie, and Tibshirani, 2010).

# 5 Analysis

We want to find the IRT estimates of the item and ability parameters, using collected industry data. As mentioned in Section 2.2, we have half-complete industry testing data and complete industry testing data, and we use the methods mentioned in Section 4 to estimate the IRT parameters, using the industry data. However, in order to better analyze these IRT estimates, we also estimate and evaluate the accuracy of the IRT parameters, which were estimated using generated complete and half-complete testing data from selected prior distributions for each ability and item parameter.



(a) $\theta$

(b) a

(c) b

Figure 2: Density plots of 10,000 of each parameter in the IRT paradigm, each generated with the respective chosen prior distribution for that paradigm

## 5.1 Generating Data

The following are the prior distributions used to generate our data:

- $\theta \sim$ truncnorm(min = -4, max = 4, mean = 0, var = 0.5). The distribution for $\theta$ is plotted in Figure 2a.
- a $\sim$ lognormal(0, 1)

  The parameters and choice to use a lognormal distribution is based off Natesan's similar choice for prior distribution (Natesan et al., 2016). The distribution for parameter a is plotted in Figure 2b.
- b $\sim$ truncnorm(min = -4, max = 4, mean = 0, var = 1)

  The normal distribution was used since the normal distribution can serve as prior distribution for difficulty (Harwell and Baker, 1991). Variance was chosen to be large enough, and range chosen so 4 standard deviations can be seen. The distribution for parameter b is plotted in Figure 2c.
- c $\sim .5 - 0.25\cdot$ bern(.95)

  Most multiple choice questions from SpeedLabs are likely to have a guessing parameter around 0.25. However, occasionally, some questions may have answers that are obviously not correct, so we account for that by giving each question a 5% chance of having a 50% chance of getting right. The distribution for parameter c is plotted in Figure 3.
- d $\sim$ 1.0

  Parameter d would be the chance a question could have of being answered correctly. We chose 1.0 for parameter d, as we are using the 3 parameter logistic model (3-PL), instead of the 4-PL.

## 5.2 Parameter Estimation with Complete Data

In this section, we analyze the IRT estimates, which were estimated using complete data, to further understand how the estimates change under different conditions, such as testing data size and whether testing data is generated versus collected. For the sets of complete testing data throughout this section, the methods referenced in Section 4 are used to estimate the item and ability parameters. Since testing data is complete, we do not use any matrix completion algorithms on the data before estimating the item and ability parameters. In Section 5.3, we compare how the results in this section compare to IRT estimates, given half-complete data, instead of complete data, to see how they differ and remain the same.

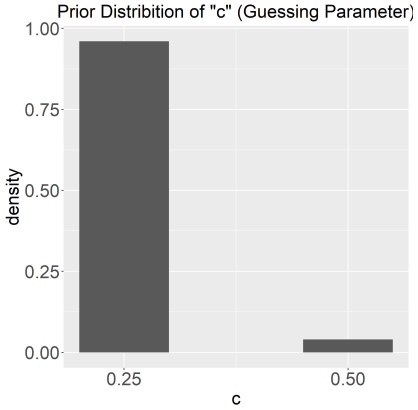


Figure 3: Density plot of 10,000 ”c” parameters, each generated from the prior distribution of ”c”

### 5.2.1 Parameters Estimation with Complete Generated Data

Here, we focus on the IRT estimates, given complete testing data generated with the chosen prior item and ability distributions. The accuracy and variability of the estimates is examined, so in later sections, we can compare the difference and similarities of parameter estimation with complete generated data versus parameter estimation with industry data or half-complete data.

In Table 1, we look at how the dimensions of a complete data set affect the accuracy of our model in estimating student ability. Given the number of students and number of items, algorithm 1 was used to create Table 1. In choosing which sets of dimensions to analyze, we never choose anything with less than 20 items, as that often leads to several outlying local maxima (Magis, Béland, and Raîche, 2011). We also choose sets of dimensions to match the dimensions of the available industry data mentioned in Section 2.2, so that we can compare the results. We would have chosen more sets of dimensions or larger dimension sizes, but were limited by time constraints of the large computation time.

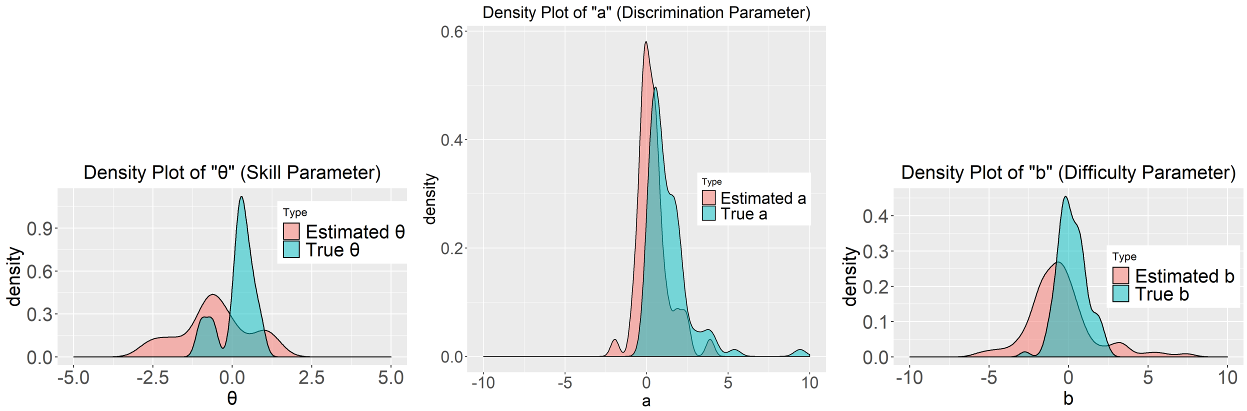
**Algorithm 1:** Algorithm for Creating Table 1

```
Student Number $S$ , Item Number $I$
array $RMSE[50, 3]$
for $i$ in $1 : 50$ do
    $A_{true}, B_{true}, C_{true} = GenerateItemParameters(I)$
    $\theta_{train} = GenerateAbilityParameters(S)$
    $Data_{Train} = GenerateTestingData(A_{true}, B_{true}, C_{true}, \theta_{train})$
    $model = ltm.tpm(TestingData)$
    for $j$ in $1 : 3$ do
        $\theta_{true} = GenerateAbilityParameters(S)$
        $Data_{Test} = GenerateTestingData(A_{true}, B_{true}, C_{true}, \theta_{true})$
        $\theta_{estimate} = factor.scores(model, Data_{Test})$
        $RMSE[i, j] = \sqrt{mean((\theta_{true} - \theta_{estimate})^2)}$
    end
end
Observe: the mean (Mean RMSE), 95% confidence interval (C.I. of RMSE), and standard deviation (S.D. of
  RMSE) of RMSE
```

From Table 1, there are few patterns in how dimension size affects the RMSE of the ability estimates. While normally more data results in either a constant or lower RMSE, solely adding more questions sometimes increases, or worsens, the RMSE of the ability, as seen in going from 10 student by 100 questions (1.187) to 10 students by 1000 questions (1.556) and from going from 50 students by 50 questions (0.529) to 50 students by 250 questions (0.812). We are uncertain why this happens, since the accuracy of the parameters should not suffer with an increase in data, as shown in Section 3. However, it is possible that with more questions, the optimization methods chosen have a harder

| Student # | Question # | Mean RMSE | C.I. of RMSE | S.D. of RMSE |
|---|---|---|---|---|
| 10 | 100 | 1.187 | ( 1.114 , 1.261 ) | 0.407 |
| 10 | 1000 | 1.556 | ( 1.461 , 1.650 ) | 0.522 |
| 100 | 100 | 0.530 | ( 0.488 , 0.573 ) | 0.235 |
| 50 | 50 | 0.529 | ( 0.496 , 0.561 ) | 0.181 |
| 50 | 250 | 0.812 | ( 0.752 , 0.872 ) | 0.332 |
| 250 | 50 | 0.540 | ( 0.503 , 0.577 ) | 0.206 |

Table 1: The IRT model is used to estimate the ability parameter of 50 sets, given complete generated testing data, with dimensions # Students by # Questions. Then, given the root mean square error (RMSE) of estimated ability parameter versus the true ability parameter, the mean, confidence interval (C.I.), and standard deviations (S.D) of all the RMSE measurements are recorded



(a) $\theta$

(b) a

(c) b

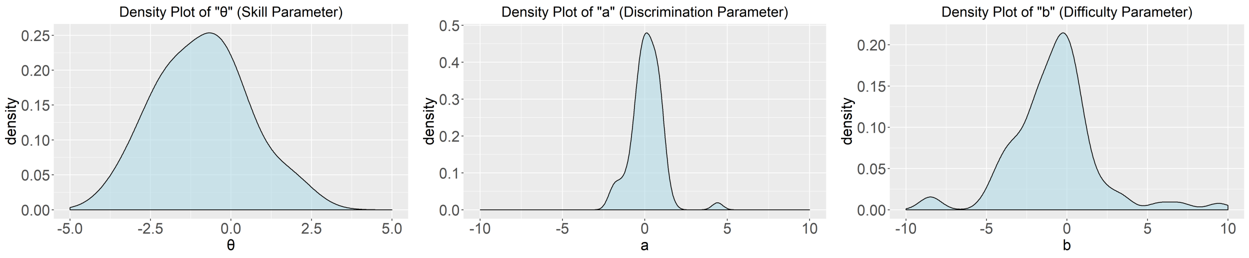
Figure 4: The distributions of the true parameter priors and the IRT estimates of the parameter priors, using generated complete testing data with 10 students and 100 questions

time estimating the item parameters. Therefore, since we use the item parameters to estimate the ability parameters, the estimation of the ability parameter will be adversely affected by the errors made in question parameter estimation. Therefore, the RMSE could go up when increasing the number of questions, while holding the number of student constant.

The dimension sizes with the lowest, or best, RMSE and the smallest variation in RMSE are the sets of testing data with 50 students by 50 questions, 100 students by 100 questions, and 250 students by 50 questions. This indicates that having dimensions that are equivalent to each other or having more students than questions may result in a higher accuracy of the model. This could be related to having larger variation in student ability. Furthermore, it is worth noting that the testing data with the best average RMSE has an RMSE around 0.5, which is the variance of the chosen prior distribution for the ability parameter.

The distributions of the estimated item and ability parameters versus the distributions of the true item and ability parameters for a complete set of 10 students by 100 questions testing data are plotted in Figures 4a to 4c. We choose 10 students and 100 questions as our dimensions to generate the data, since they are similar to the dimensions of the complete industry data set found in Section 2.2.

The distribution of the estimated discrimination parameter has a similar variance and skews only slightly to the left of the distribution of the true discrimination parameter in Figure 4b. However, in Figure 4a, the distribution of the estimated $\theta$ has a similar average but larger variance than the distribution of the true $\theta$. Similarly, the distribution of the estimated difficulty parameter has the same mean but larger variance than the distribution of the true difficulty parameter. This is probably because the chosen prior variances may need to be larger, since very small prior variances for small samples is discouraged (Yanyan, 2010). While it is out of the scope and time available to complete this paper,



(a) $\theta$

(b) a

(c) b

Figure 5: Distribution of the IRT estimates of each parameter for collected complete industry testing data with 11 students and 98 questions

the variance chosen for the priors could be adjusted to see how the model estimates improve as a function of the prior variance.

More visualizations of parameter estimates with complete generated data can be found in Section 8.2.

### 5.2.2 Parameter Estimation with Complete SpeedLabs Data

This section focuses on the IRT estimates of the complete industry testing data mentioned in Section 2.2. We use our model to estimate the IRT parameters, using SpeedLabs' complete testing data with 11 students and 98 questions. We then plot the distributions of the parameters in Figure 5a, Figure 5b, and Figure 5c, using the previous section to contextualize the analysis of the plots.

We can see that the distributions of $a$, $b$, and $\theta$ are all estimated to be between -5 and 5. However, the distribution of the estimated ability parameter, with only 11 students, has less data points than the distribution of the estimated item parameters, which has 98 questions, and therefore, it probably contains more error. The variations and means of Figure 5a, Figure 5b, and Figure 5c are similar to the distributions of the estimated parameters but different from the distributions of the true parameters in Figure 4a, Figure 4b, and Figure 4c, respectively. Since the distributions of the estimated parameters are similar for both generated and SpeedLabs' data, the parameter estimation methods, given small sized data, may distribute the estimated parameters in a similar way, regardless of the distribution of the true parameters.

In Figure 5b, we see that the estimate for $a$ is sometimes negative. This should not be the case, since a negative discrimination implies that a student with a higher level of ability has a smaller probability of getting the question correct, which could map to a poorly written question or incorrect labeling of answers. This contradicts the assumption that the ICC is monotonic, as mentioned in Section 3. This trend for our model to estimate a negative $a$ is seen in Figure 4b, as well. This could potentially occur if the question poorly discriminates between those with high and low ability (Yang and Kao, 2014), but it could also be a result of problems in optimization.

## 5.3 Parameter Estimation with Incomplete Data

This section contains similar tables and graphs as Section 5.2, except we now use only generated and real data that is half complete. We use half-complete data here because, as mentioned in Section 2.2, the industry data is very sparse. We show that the model estimates are still comparably accurate when the half the data is imputed versus having full data, which allows us to reliably use more of the industry data than we would have been otherwise when estimating the IRT parameters. As referenced in Section 4, to deal with the problem of having incomplete testing data, the Soft-Impute algorithm is used to complete the data. Then, using the data with imputation, we estimate the IRT parameters.

### 5.3.1 Parameter Estimation with Incomplete Generated Data

Before we estimate the IRT parameters with half-complete industry data, we first estimate the IRT parameters with half-complete generated data, so we can analyze the accuracy and variance of the estimates. With Table 2, we can see the average RMSE (Mean RMSE (H)) of the estimated ability parameter versus the true ability parameter, which were estimated using half complete generated testing data, with # Students and # Questions (Mean RMSE (H)). Out of convenience, the average RMSE calculated in Table 1 (Mean RMSE(C)) is also included. Mean RMSE (H) is calculated in the same way that Mean RMSE (C) is, except all the generated data has exactly half its data removed at random. Instead of estimating parameters with the half-complete data, we first complete the data with the Soft-Impute algorithm, and then run our model on the fully completed set of data. The accuracy of the Soft-Impute algorithm in imputing the data correctly is given by S.I. Accuracy, which measures how accurate the imputations of the entries themselves are. The percent change between Mean RMSE (C) and Mean RMSE (H) is given in RMSE Change, and the standard deviation of the RMSE for the model run on the half complete data set is given by S.D of RMSE (H).

| Students | Questions | S.I. Accuracy | Mean RMSE (C) | Mean RMSE (H) | RMSE Change (%) | S.D. of RMSE (H) |
|---|---|---|---|---|---|---|
| 10 | 100 | 0.620 | 1.187 | 0.882 | -25.6 | 0.373 |
| 10 | 1000 | 0.634 | 1.556 | 0.913 | -41.2 | 0.351 |
| 100 | 100 | 0.689 | 0.530 | 0.552 | 4.24 | 0.269 |
| 50 | 50 | 0.671 | 0.529 | 0.646 | 22.1 | 0.237 |
| 50 | 250 | 0.685 | 0.812 | 0.433 | -46.5 | 0.287 |
| 250 | 50 | 0.692 | 0.540 | 0.642 | 18.8 | 0.235 |

Table 2: The soft-impute matrix algorithm is used to complete 50 sets, given half-complete generated testing data, with dimensions Student # by Question #. Then, the IRT model is used to estimate the ability parameter of the imputed testing data. Finally, given the root mean square error (RMSE) of estimated ability parameter versus the true ability parameter, the average (Mean RMSE(H)) and standard deviations (S.D of RMSE (H)) of all the RMSE measurements are recorded as well as how the RMSE Change between the Mean RMSE here and the Mean RMSE in Table 1 (Mean RMSE (C))

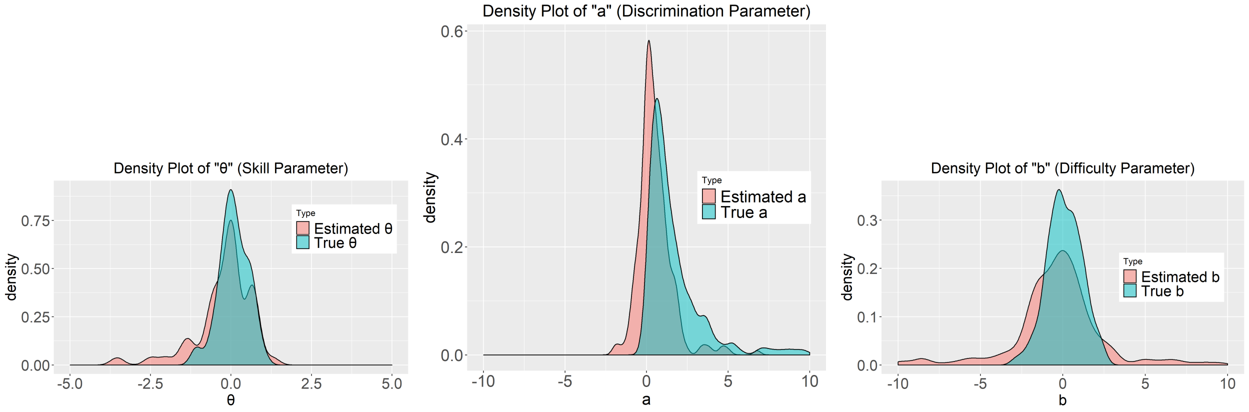
From Table 2, several different observations can be made. The first is that the Soft-Impute's accuracy is higher with more data. Since every entry can only be 0 or 1, the minimum accuracy for Soft-Impute is expected to be 0.5, as that would be equivalent to randomly guessing 0 or 1 for each value to impute. However, Soft-Impute consistently does better than random, and it even has accuracy measures 0.685 or higher for the larger data set sizes.

Another observation is that RMSE change is largest when the Mean RMSE (C) values are low. This indicates that removing the data and imputing it does negatively affect a model, if the model's estimated ability parameters had a low RMSE. However, the highest percent change is low, with 22% as the maximum increase in the RMSE. Moreover, the RMSE change is negative in the three places in which the IRT estimates are most inaccurate. This indicates our estimates are better when we randomly remove half our data and then complete it with Soft-Impute. While surprising, this may be because there is a high amount of variance in the data itself and these three places contain higher parameter to data ratios than other places. This is supported by Table 1, which shows that the RMSE of the ability parameter have the highest amount of variance at the same three places. However, Soft-Impute uses a regularization parameter to control for training error (Mazumder, Hastie, and Tibshirani, 2010), which means the completed data would be regularized. This would result in a higher level of bias, but due to the bias-variance trade-off, a lower level of variance. The possible lower variance in the resulting completed data could be the reason for more accurate IRT estimates. Future work can study this phenomenon of regularization as a side-effect of imputation via matrix completion.

A third observation concerns the accuracy of our IRT parameter estimates of our two sets of industry data: the first that is complete with 11 students, and 98 questions and the second that is half-complete with 65 students and 241 questions. The Mean RMSE (C), given testing data with 10 students and 100 questions, is 1.187 but the mean RMSE (H), which were estimated using testing data with 50 students and 250 questions, is .433, which is almost one third the magnitude. Therefore, the results from Table 2 indicate that the IRT estimates, which were estimated using the larger and half-complete industry testing data, are likely to be more accurate.

Figures 6a to 6c are equivalent to Figures 4a to 4c. The only difference between the two sets of plots are that Figures 6a to 6c show the distributions of IRT estimates, which were estimated using half-complete generated testing data with 50 students and 250 questions, instead of complete testing data with 10 students and 100 questions. We chose a different set of dimensions since the largest set of half-complete industry data collected in Section 2.2 contained a similar set of dimensions (65 students and 241 questions).

The means in of the estimated distribution versus true distributions are the same in each of Figures 4a, 4c, 6a and 6c. However the variance of the distribution of the estimated parameter is similar to the variance of the distribution of the true parameter in Figures 6a and 6c but the variance of the distribution of the estimated parameter is much larger than

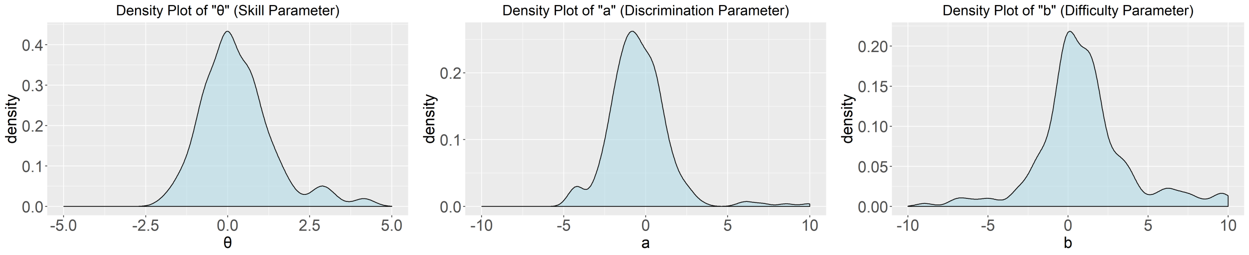


(a) $\theta$

(b) a

(c) b

Figure 6: The distributions of the true priors for each parameter and the IRT estimates of the priors for each parameter, which were estimated using generated half-complete testing data with 50 students and 250 questions



(a) $\theta$

(b) a

(c) b

Figure 7: Distribution of the IRT estimates of each parameter, which were estimated using collected half-complete industry testing data with 65 students and 241 questions

the variance of the distribution of the true parameter in Figures 4a and 4c. Furthermore, the variance of the estimated distribution is larger in Figure 4a than Figure 6a and like wise with Figure 4c than Figure 6c. This indicates that our model does a better job of estimating $\theta$ and $b$ when given the half complete larger set of testing data, than when given the complete smaller set of testing data, which is further supported by Table 2. There seems to be no notable difference in Figure 6a and Figure 4a but it is worth noting the leftward skew of the distribution of the estimated $a$ is prevalent in both graphs.

More visualizations of parameter estimates with half-complete generated data can be found in Section 8.3.

### 5.3.2 Parameter Estimation with Incomplete SpeedLab Data

In this section, we estimate the testing data from SpeedLabs of 65 students and 241 questions found in Section 2.2. Similarly to Figure 5a, Figure 5b, and Figure 5c, we plot the distributions of the estimated IRT parameters, which were estimated using the collected half-complete industry data, in Figure 7a, Figure 7b, and Figure 7c. Then, we analyze the similarities and differences between the distributions of the estimated parameters, which were estimated using half-complete industry data, and the estimated parameters, which were estimated using complete industry data, using previous sections to support our analysis.

The distribution of $\theta$ in Figure 7a has a smaller variance than the distribution of $\theta$ in Figure 5a. This may be due to a higher level of accuracy in the estimate of the ability parameter in testing data with higher dimensions, as indicated by Table 2. Also, a similar relationship between data dimension size and the variance of the estimated distribution is noticed with Figure 4a and Figure 6a.

The variance of $a$ is higher in Figure 7b than Figure 5b, but they have similar means. While the distributions in Figure 5c and Figure 7c are different, there does not appear to be any difference of practical significance based off these graphs.

In Figure 5b, we see that the estimate for $a$ is sometimes negative and this trend continues in Figure 7b. This further contradicts the assumption that the ICC is monotonic, as mentioned in Section 3. While the distribution of the estimated $a$ in Figure 6b appears to be skewed left over the true distribution, the skew is not great enough to account for the large leftward skew seen in Figure 7b. Even though this trend continues to be worth exploring, it is out of the scope of this paper.

# 6 Conclusion

As mentioned in Section 5.2 and Section 5.3, we conclude that using a matrix completion method for imputing half-complete testing data before estimating the IRT parameters is a reasonable method when the data is sparse. We also conclude that having either more students and questions or more students in the data is useful for more accurate estimates of the item and ability parameters. However, increasing the number of questions in a disproportionately large amount, compared to the number of students, can decrease the accuracy of the estimates.

In this paper, we analyze how our model performs with half-complete matrices. However, while the testing data may be able to be more sparse, the whole set of industry data is missing 96% of the data, which is much sparser than half-complete matrices. Therefore, there is too much missing data to adequately complete the entirety of the industry testing data for use in our model, even with our current method of estimating parameters.

As mentioned in Sections 5.2 and 5.3, we saw the trend of negative discrimination parameter ($a$) estimates for select questions in the industry data. Since only select questions received negative $a$ estimates while others received positive $a$ values, and the trend only occurred with industry and not generated data, it may be the case that these questions with negative $a$ parameters are not adequately testing the students ability. An educational expert should review some of these questions to help determine if it is an error in the analysis or perhaps something related to question design or answer labeling.

The next step is to define an item selection method based off student ability and difficulty. Then, if the IRT model and matrix completion algorithm are used in an adaptive question recommendation system, more work needs to be done to determine exactly what sort of conditions are optimal. The exact relationship between number of questions, number of students, and sparseness of the data set to the accuracy of the item selection method should be determined so that companies like SpeedLabs may be able to use the item selection method to choose questions in such a way that is reliable way for concept mastery. Furthermore, more work should be done on understanding the inconsistencies highlighted in Section 5.2 and Section 5.3, in case they appear when estimating parameters in the industry data.

# 7 Acknowledgements

I would like to thank Professor Ambuj Tewari for his support. Next, I would like to thank Laura Niss for all her help and guidance. Lastly, I would like to thank Haonan Sun for his contributions to this project.

The data used in this paper is provided by by SpeedLabs.

# References

Bock, R. Darrell and Murray Aitkin (Dec. 1981). “Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm”. en. In: *Psychometrika* 46.4, pp. 443–459.

Carlson, James E. and Matthias von Davier (2013). “Item Response Theory”. en. In: *ETS Research Report Series* 2013.2. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/j.2333-8504.2013.tb02335.x, pp. i–69.

Harwell, Michael R. and Frank B. Baker (Dec. 1991). “The Use of Prior Distributions in Marginalized Bayesian Item Parameter Estimation: A Didactic”. en. In: *Applied Psychological Measurement* 15.4, pp. 375–389.

Iulus Ascanius (Jan. 2008). *English: Graphical demonstration of IRT parameters.*

Johnson, Matthew (May 2007). “Marginal Maximum Likelihood Estimation of Item Response Models in R”. In: *Journal of Statistical Software* 20.

Magis, David, Sébastien Béland, and Gilles Raîche (Mar. 2011). “A Test-Length Correction to the Estimation of Extreme Proficiency Levels”. In: *Applied Psychological Measurement* 35, pp. 91–109.

Mazumder, Rahul, Trevor Hastie, and Robert Tibshirani (Aug. 2010). “Spectral Regularization Algorithms for Learning Large Incomplete Matrices”. en. In: p. 36.

Natesan, Prathiba et al. (2016). “Bayesian Prior Choice in IRT Estimation Using MCMC and Variational Bayes”. English. In: *Frontiers in Psychology* 7. Publisher: Frontiers.

Sinharay, Sandip and Russell G. Almond (Apr. 2007). “Assessing Fit of Cognitive Diagnostic Models A Case Study”. en. In: *Educational and Psychological Measurement* 67.2. Publisher: SAGE Publications Inc, pp. 239–257.

*SpeedLabs - Online Learning Platform for Exam Prep* (2020). en. Library Catalog: www.speedlabs.in.

Udell, Madeleine and Alex Townsend (May 2018). “Why are Big Data Matrices Approximately Low Rank?” In: *arXiv:1705.07474 [cs, stat]*. arXiv: 1705.07474.

Yang, Frances M. and Solon T. Kao (June 2014). “Item response theory for measurement validity”. In: *Shanghai Archives of Psychiatry* 26.3, pp. 171–177.

Yanyan, Sheng (Jan. 2010). “A sensitivity analysis of Gibbs sampling for 3PNO IRT models: Effects of prior specifications on parameter estimates”. In: *Behaviormetrika* 37, pp. 87–110.

Zhang, Susu and Hua-Hua Chang (2016). “From smart testing to smart learning: how testing technology can assist the new generation of education”. In:

# 8 Appendix

## 8.1 More Visualizations of SpeedLabs' Data

Before we chose to use the IRT model to estimate item and ability parameters, several visualizations were done to get a sense of what the data looks like. First, for each each session each user logged, we look at how many questions the users answers in Figure 8.

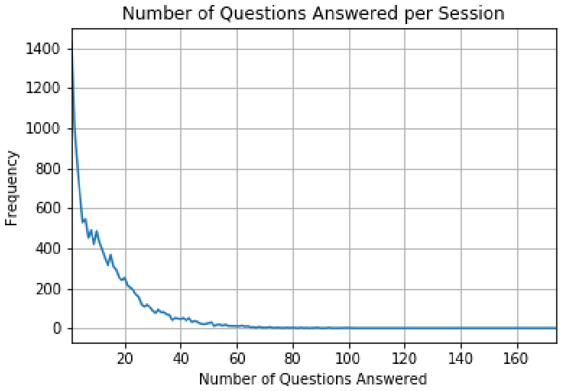


Figure 8: The number of questions answered per logged session in SpeedLabs' testing data.

The graph shows that most sessions last a short amount of questions and that the frequency of sessions decreases as the number of questions per sessions increases. In fact, of the 5,228 logged sessions, the median session length is 12 answered questions.

In Figure 9, we show the distribution of the proportion that questions were answered correctly versus the total times there were answered, as a proxy for question difficulty. We label this proxy for question difficulty for each question the "proportion correct". Only question data where the question was answered at least 6 times, which was the average amount of times a question was seen, is considered in Figure 9 to get a more accurate measure of the distribution of the proportion correct per question.

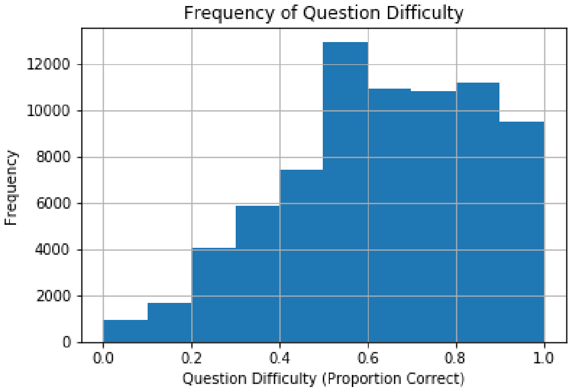


Figure 9: A bar plot of the proportion correct per question in SpeedLabs' data, including only questions answered more than 6 times.

Based off, Figure 9, the proportion correct per question skews towards easy but there is lots of variation.

Finally, we looked at the difference in student ability from the beginning of a session compared to the end of a session in Figure 10. We used the proportion of answers the student got correctly over the total questions the student answered, as a proxy for ability. We then plotted the difference in proportion correct of the first five and last five questions for sessions of length at least 20 questions.

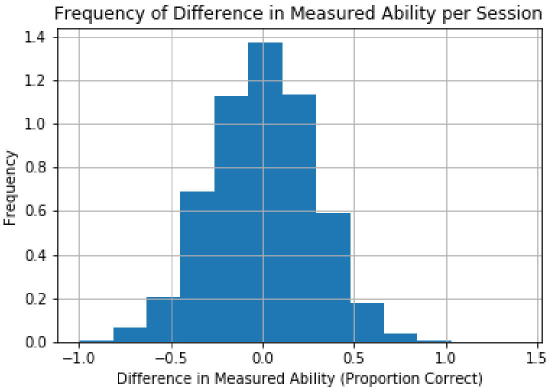
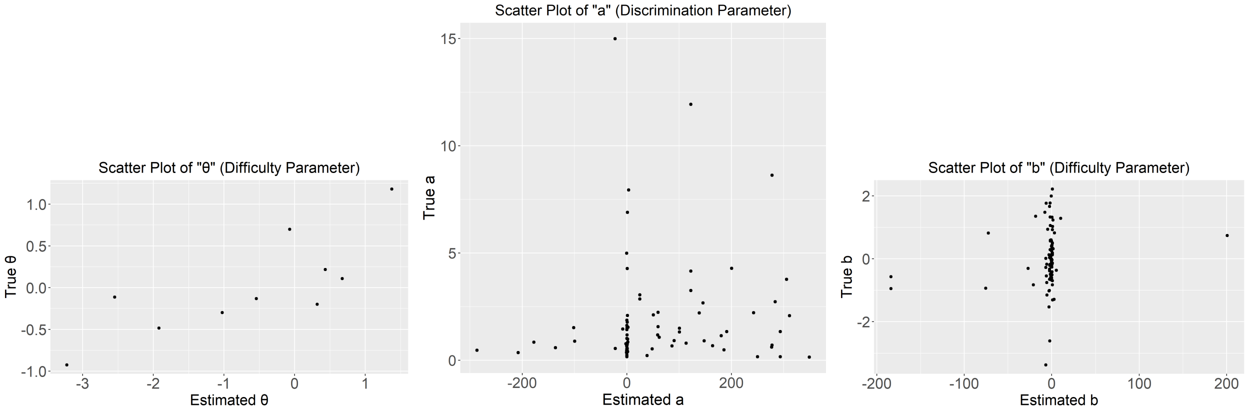


Figure 10: a



(a) $\theta$

(b) a

(c) b

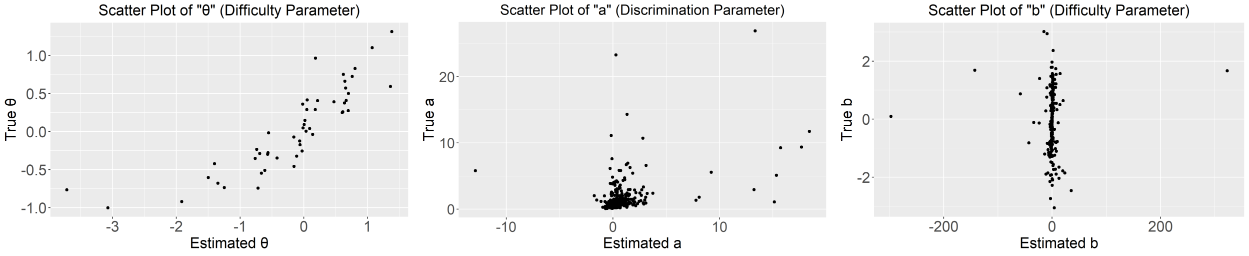
Figure 11: The scatter plots of the true parameter in the y-axis and the estimated parameter in the x-axis for each parameter, which was estimated using generated complete testing data with 10 students and 100 questions

There does not appear to be a strong argument that a student's ability, when measured in this way, has improved over the session. This could be an artifact of student motivation, in that many incorrect answers correlate with the ending of a session.

## 8.2 More Visualizations of IRT Model Estimates with Complete Data Set

The following Figures 11a to 11c are similar to Figures 4a to 4c in that are they show the resulting estimated ability and item parameters from a model, using complete generated 10 student by 100 question testing data. However, Figures 11a to 11c graph scatter plots of each parameter, where the y-axis is the value of the true parameter and the x-axis is the value of the estimated parameter.

The scale of the x-axis of Figures 11a to 11c is much larger than the scale of the y-axis. This indicates that the estimates are often overestimates, in terms of magnitude. However, even considering the scale change, there does not seem to be any discernible relationship between the true parameter and the estimated parameter in either Figure 11b or Figure 11c. Furthermore, Figure 11a has very few data points, so it is difficult to see if there is a clear relationship between the estimated and true $\theta$.



(a) $\theta$

(b) a

(c) b

Figure 12: The scatter plots of the true parameter in the y-axis and the estimated-parameter in the x-axis for each parameter, which was estimated using generated half-complete testing data with 50 students and 250 questions

## 8.3 More Visualizations of IRT Model Estimates with Half-Complete Data Set

The following Figures 12a to 12c are equivalent to Figures 11a to 11c, but consider half-complete generated data with 50 student by 250 question testing instead.

The scale of the estimated parameters tend to be much larger than the true parameters in Figures 12a to 12c, as it did in Figures 11a to 11c. The estimates of the item parameters have no clear relationships with the true values of the item parameters in Figures 12b and 12c. However, Figure 12a has many more data points than Figure 11a, so we can see that the estimated $\theta$ is correlated with the true $\theta$, as there is a linear relationship displayed in the graph.