University of Brasília
Institute of Exact Sciences
Department of Statistics

Master's Dissertation

Deep IRT: an application of deep learning methods to Item Response Theory

by

Lucas de Moraes Bastos

Brasília, June of 2024

Deep IRT: an application of deep learning methods to Item Response Theory

by

Lucas de Moraes Bastos

Dissertation submitted to the Departament of Statistics at the University of Brasília, as part of the requirements required to obtain the Master Degree in Statistics.

Advisor: Prof. Dr. Guilherme S. Rodrigues

Brasília, June of 2024

Dissertation submitted to the Graduate Program in Statistics of the Departament of Statistics at the University of Brasília, as part of the requirements required to obtain the Master Degree in Statistics.

Approved by:

Prof. Guilherme S. Rodrigues

Advisor, EST/UnB

Prof. Dalton Franscisco de Andrade

Professor, UFSC

Dr. Antonio Eduardo Gomes

EST/UnB

Dr. José Augusto Fiorucci

EST/UnB

The only thing that interferes with my learning is my education.

(Albert Einstein)

Acknowledgments

Firstly, I acknowledge and thank what made this work possible. Some may call that chance or randmoness, but I recognize as God. In any case, it is undeniable the existence of an adequate and precise concatenation of events out of our control in order for anything to come to life.

In this case, this fortunate sequence of events begun with my parents, who always encouraged me to study and taught me the value of knowledge. I specially thank my dear mother, who gave me happy memories in the first steps of the student life, by showing me how to study, going even to the trouble of preparing review tests before each exam I would take until the fifth year of middle school.

Next, I had the grace and privilege of being introduced to my lucid advisor, Professor Guilherme S. Rodrigues, by dear Professor Cibele Queiroz da Silva and Professor Antonio Eduardo Gomes. I complete the fourth work under his apt and thoughtful orientation, not only concerning to the technique, but to the human aspect involved, him being the touchstone that allowed the birth of whatever good there is in this work. What perhaps have not reached a satisfactory level must be attributed only to myself.

Finally, nonetheless most fundamentally, I thank my beloved friend, lifemate, Ana Júlia, who since we met has been my luminous beacon, my breath, my sunlight, my peace...
This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

Resumo Expandido

TRI Profundo: uma aplicação de métodos de redes neurais profundas à Teoria de Resposta ao Item

Este trabalho buscou um novo método para a estimação do nível de conhecimento de alunos, no contexto de TRI para estimação da habilidade latente, o que foi feito ao criar um modelo de redes neurais capaz de utilizar algoritmos de aprendizado de máquina a fim de estimar os parâmetros do ML3. Essa rede foi aplicada ao Exame Nacional do Ensino Médio (Enem), resultando em estimativas mais precisas do que alcançado por métodos tradicionais. Também faz parte do escopo deste trabalho a replicação dos métodos oficiais de correção e atribuição de notas aos alunos em uma linguagem aberta.

O trabalho é introduzido por uma contextualização do que é o Exame Nacional do Ensino Médio, destacando sua relevância para a população e para a educação do país. Os métodos utilizados na correção desse exame são indicados, com observações acerca de suas limitações intrínsecas. Então, métodos de inteligência artificial são ilustrados, demonstrando sua abrangência e grande utilidade, e, em sequência, a utilização desses métodos em Teoria de Resposta ao Item, na área que está sendo denominada TRI Profunda, sendo este trabalho uma expansão dos modelos atualmente disponíveis e sua aplicação aos dados do Enem.

Em seguida, a revisão de literatura busca cadenciar o surgimento da TRI Profunda, tratando

inicialmente da própria TRI e de redes neurais profundas, para mostrar como o modelo tradicional de 3 parâmetros (ML3) pode ser escrito como uma rede neural. Então, é mostrado sucintamente o modelo de TRI Profunda de Tsutsumi et. al (2021) e o que seriam *positional encoders*, método utilizado mais à frente. Ainda como parte da revisão, são resumidas informações sobre o método de estimação utilizado pelo Inep e sobre as bases de dados disponibilizadas publicamente e que foram o escopo de análise deste trabalho.

O Capítulo seguinte apresenta os dois métodos propostos para tratar dados binários originados de respostas de alunos a um teste: ML3 Raso (*Shallow 3PL*) e ML3 Profundo (*Deep 3PL*), buscando extender os modelos previamente revisados, especialmente no caso deste último, que se deriva mais diretamente do modelo de Tsutsumi et al. (2021). Os modelos utilizam estratégias diferentes para lidar com o grande volume de dados de entrada, um com camadas de *embedding* e o outro com *positional encoders*.

A Aplicação inicialmente descreve os passos percorridos neste trabalho, desde o tratamento da base de dados, indicando os *softwares* e métodos utilizados. Então são apresentados os resultados, primeiramente comparando o alcançado pelo Inep utilizando o *software* proprietário BILOG e sua replicação no *mirt*, pacote de TRI disponível no R. Estes resultados são então comparados aos modelos propostos, destacando-se a performance do modelo ML3 Raso, mais preciso (com o menor EQM e perda) e maior verossimilhança das probabilidades de acerto estimadas às respostas empíricas. Os resultados alcançados em lingaguem aberta também indicam performance superior, apesar da falta de informações divulgadas sobre o método oficial de estimação.

Por fim, esses resultados são sintetizados na Conclusão, com sugestões para trabalhos futuros, destacando-se a fixação dos parâmetros dos itens na estimação das habilidades dos alunos. Assim a estimação das notas poderá ser feita pelo algoritmo Adam, muito mais parcimonioso, em vez das aproximações e integrações (Quadratura Gaussiana) do algoritmo EM utilizado. Dessa forma, apresenta-se uma alternativa que poderia contribuir com o trabalho atualmente feito pelo Inep, tanto na questão da transparência do cálculo das notas dos alunos, como na

simplicidade e maior confiabilidade da estimação conduzida pela rede neural.

Palavras-chave: Teoria de Resposta ao Item (TRI); redes neurais profundas; aprendizado de máquina; inteligência artificial; TRI Profundo

Abstract

This work aimed to provide a new method for the estimation of examinees' abilities in the IRT context, which was done by designing a novel neural network capable of using machine learning algorithms to estimate the 3PL IRT model parameters. This network was applied to Brasil's National High School Exam (Enem), yielding more accurate estimates than the traditional methods. It is also in the scope of this work to replicate the official methods used in the evaluation of students' grades in an open software.

Keywords: Item Response Theory (IRT); deep neural networks; machine learning; artificial intelligence; Deep IRT

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The Enem (National High School Exam) is a multidisciplinary test administered annually, since 1998, by Inep, the Anísio Teixeira National Institute for Educational Studies and Research. While primarily aimed at high school students, participation is open to anyone with a valid identity document, making it the largest educational assessment in Brazil. Its significance arises from three main factors. Firstly, since 2009, it serves as the primary pathway for admission to most public universities in Brazil, including prestigious institutions and several universities in Portugal. It also facilitates access to full scholarships and educational loans in private institutions. Secondly, it provides an assessment of the quality and effectiveness of education provided in the country. Lastly, it serves as a historical benchmark for monitoring educational standards in Brazil. Given these reasons, it is crucial to ensure the credibility of the exam's measurement of students' knowledge to maintain fairness among all candidates. To achieve this, Inep employs Item Response Theory (IRT) alongside traditional estimation methods. However, it is concerning that Inep's code is not open access and that it utilizes expensive and obscure software. The difficulty in replicating one's Enem grade has been a long-standing criticism voiced by students and experts alike.

The current model used by Inep to estimate student's grades in Enem is the traditional 3PL model (Birnbaum, 1968), which presents intrinsic assumptions, such as local independence and

the assumption of an underlying data generating distribution. When not satisfied, it can lead to suboptimal results, compromising the estimates accuracy, most importantly, the values for the latent skill θ, which directly correspond for the students' grades.

In the last decade, deep neural networks arose again with a boom of new applications in the most diverse fields of knowledge, going from language models (ChatGPT from OpenAI) to computational vision (Khan, Salman et al., 2018), helping in quantile regression (Kumar, Liang, and Ma, 2019), as well as other classes of probabilistic models. Artificial intelligencies have been surprising users all over the world and are being implemented in many businesses and government departments, automating varied working processes or supporting many other tasks, such as contract review (How AI Is Changing Contracts by Beverly Rich, Harvard Business Review, 2018), review and support in judicial systems (K. Ashley, Artificial Intelligence and Legal Analytics, Cambridge University Press, 2017).

Many authors sought methods that combine the IRT interpretability advantage with the predictive capability of deep neural networks, firstly to improve Knowledge Tracing (KT) - see Cheng and Liu (2019), Gan et al. (2020) and of Ghosh et al. (2020). There are even some models that seek to evaluate discursive responses - Amur, Z. H., Hooi, Y. K., Soomro, G. M. (2022, December): Automatic Short Answer Grading (ASAG) using Attention-Based Deep Learning Model.

For the task of exam grading, a new model was developed by Tsutsumi et al. (2021), introducing an architecture able to combine two neural networks - one for the abilities and another for the items - to describe an alternative to the 2PL IRT model. The authors show that joint estimation of the parameters have the benefits of raising the predictive capability while naturally dealing with heterogeneous populations, since it does not assume any underlying distribution.

Despite these advancements, the current Deep-IRT model does not include the "guessing" parameter, and it is not known how well it will perform when applied to Enem data. In this dissertation, we will seek to apply machine learning estimation methods to the IRT 3PL model and to introduce a new model, an extended version of Deep-IRT by Tsutsumi et al. (2021). At

last, the available methods will be compared in the context of a national educational assessment.

In short, this work will provide three main results: an investigation of how Enem's grades and item parameters are calculated and its replication in an open software; calculating Enem's grades using deep learning estimation; and, finally, introducing a new model that combines the 3PL IRT model with deep neural networks: Deep IRT.

Chapter 2

Literature Review

This chapter aims to present the advancements made in the field of study, defining the research's scope and connecting the content in a way that leads to the subsequent evolution of methods.

To begin, a summary of Item Response Theory (IRT) will be provided, highlighting its primary estimation processes. This will be followed by a concise overview of deep neural networks and their ability to incorporate less restrictive assumptions compared to traditional statistical methods. Finally, we will explore the latest developments that merge these two domains through Deep IRT.

2.1 Item Response Theory

How can we assess a person's intellectual capacity? How do we measure their ability to select and apply information, often relying on memorization and problem-solving skills? Exams, which have evolved and taken various forms throughout history, serve as tools to measure what is now referred to as latent traits - characteristics that cannot be directly measured, such as mathematical intelligence, reading proficiency, and logical thinking.

The Chinese Bureaucracy was the earliest example of widespread use of written tests to select the most capable individuals for government service, dating back to at least 1300 years

ago (Wang, Rui, 2013). This practice took eighteen centuries to become common in the Western world. In medieval Europe and many other cultures, candidates for public offices, priestly positions, or universities admissions relied heavily on personal connections. In some cases, they were evaluated through oral examinations, such as debates (*disputatio*) or by giving lectures on the subject. In the seventeenth century, Jesuits, having contact with the highly developed Chinese civilization, introduced the written evaluation that had been in use there for millennia.

Today, exams are the most common, useful, and efficient method of assessing knowledge, despite not being able to assess other aspects equally or more important, such as discipline and commitment. Regardless, beyond the exam itself, there is a need to determine how to measure and translate the data from examinees' answers into meaningful information. The Classical Test Theory (CTT) approach, the most common one, sums up the correct answers to obtain a general test score. Another approach is Item Response Theory (IRT), which focuses on the response to each item, aiming to extract more information beyond a simple right or wrong. IRT also allows for the comparison of individuals who have participated in different tests, provided there are common items between them.

IRT is able to provide a more detailed interpretation of the cognitive process which led to a specific answer by attempting to describe the probability of a examinee j with a certain skill level to give the correct response to an item i. In turn, each item is described by parameters, such as their difficulty level and discrimination capacity (the ability of the item to differentiate between high-skilled and low-skilled examinees).

The simplest models only have one item parameter, its difficulty, and one parameter for the examinee's ability. By adding more parameters, we can attempt to describe more items' characteristics and even different skill sets, with more examinees' parameters in the multidimensional models. Nevertheless, we shall focus on the model at hand, the three-parameter logistic model.

2.1.1 Three-Parameter Logistic Model (3PL)

Presented by Birnbaum (1968), it consists in the use of the logistic function to associate skill θ_j - the latent trait - of a examinee j with three item parameters, a_i, b_i and c_i, from a certain item i, and then to compute the probability of the correct answer to item i by examinee j, p_{ij}:

$$U_{ij} \sim \text{Bernoulli}(p_{ij}),$$

$$p_{ij} = c_i + \frac{1 - c_i}{1 + \exp\left[-a_i(\theta_j - b_i)\right]},$$

$$U_{ij} = \begin{cases} 1, & \text{if examinee } j \text{ correctly answers item } i; \\ 0, & \text{if examinee } j \text{ incorrectly answers item } i; \end{cases}$$

where a_i is the discrimination parameter of item *i*; b_i represents the difficulty (position) of item *i*; c_i reflects the probability of a examinee with very low skill to correctly answer item i (lower asymptote parameter, also called “guessing parameter"); and θ_j represents the skill or ability (or proficiency) of the j-th examinee.

This probability is visualized as a function of the ability θ, forming the Item Characteristic Function or Item Characteristic Curve (ICC) - as shown in Figure 1(a). The parameter a is proportional to the derivative of the tangent of the ICC at its inflection point. It denotes the item’s capacity in discerning the proficient examinee by discriminating if the examinee possess the necessary ability to solve the item. It is not expected to have negative values, which would mean that examinees with lower skill levels have greater probability of correctly answering the item. If this happens, the item should be reviewed or discarded. The parameter b is measured on the same scale as the ability θ, while the parameter c varies between 0 and 1, being a probability. The ability scale is arbitrarily defined by any transformation of the values ranging from $-\infty$ to ∞, by choosing an origin (reference value) and a unit of measure to the variability.

The simpler models 2PL (Two-Parameter Logistic Model) and 1PL (Rasch Model) are triv-

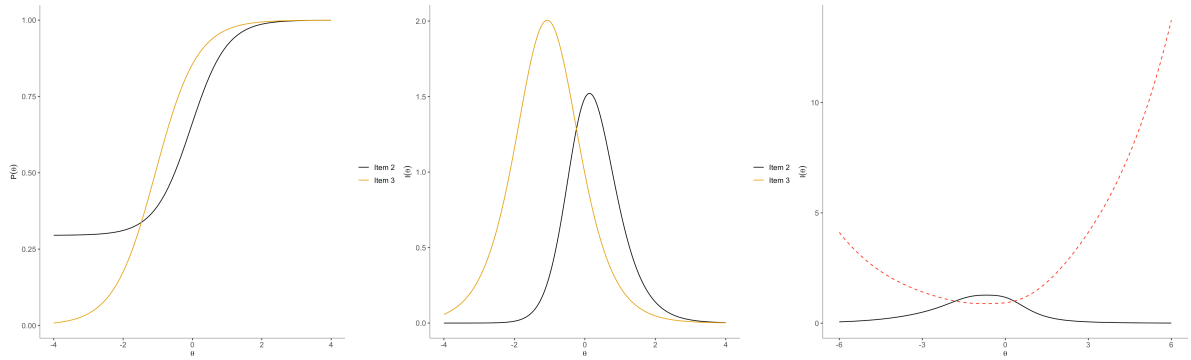
ially obtained from 3PL, as shown by Birnbaum (1968) based on the models suggested by Lord (1952). One can easily get 2PL from 3PL making $c = 0$, and from that get 1PL making a constant for all the items.

Besides the ICC, another useful visualization is the Item Information Function or Curve (IIC), which indicates the amount of information given by an item around different skill levels - Figure 1(b) - that is, how much the item contributes to the estimation of the ability of a certain examinee: $I_i(\theta) = \frac{\left[\frac{d}{d\theta}P_i(\theta)\right]^2}{P_i(\theta)[1-P_i(\theta)]}$, where $I_i(\theta)$ is the information provided by item *i* in a θ point, and $P_i(\theta)$ is the answer function of item i, $P(U_{ij} = 1|\theta_j = \theta)$. For the ML3, one can write:

$$I_i(\theta) = D^2 a_i^2 \frac{1 - P_i(\theta)}{P_i(\theta)} \left[\frac{P_i(\theta) - c_i}{1 - c_i}\right]^2,$$

where D is a constant scale factor ($D = 1,7$ when an approximation to the Normal is desired). In the form (2.1.3) of the IIC, it is clear that information is greater as b_i gets closer to θ, a_i grows and c_i approaches zero.

A measure that is very useful for an overall assessment of a test is the Test Information Curve (TIC) - Figure 1(c), which is the sum of all items' informations regarding a certain θ: $I(\theta) = \sum_{i=1}^{I} I_i(\theta)$. By this equation, we can see that each item is independent from the others when contributing to the test information, since each IIC can be obtained without knowing the other items, which is not possible in Classical Test Theory. Inversely proportional to the test information, the precision of the estimation of the ability at a point θ is given by the standard estimation error: $SE(\hat{\theta}) = \frac{1}{\sqrt{I(\theta)}}$.



(a) Item Characteristic Curves. (b) Item Information Curves. (c) Test Information Curve.

Figure 2.1: (a) ICC's of items 2 and 3 from R database LSAT7 (Bock Lieberman, 1970). The second item shows adequate discrimination power (sharp slope) and difficulty close to the mean (0), but has approximately 0.3 probability of randomly choosing the correct item. The third item is somewhat ideal, with almost no chance of lucky answers, but it's an easy question. (b) IIC's from the same items 2 and 3. One is able to see that the items provide most information around the difficulty level (when $\theta \approx b$), with the third item providing overall more information, showing that the parameters a and c really influence the item information. (c) TIC providing the overall information given by the five items in the test database. The red dashed line shows the $SE(\hat{\theta})$.

2.1.2 Estimation Methods

Inep uses marginal bayesian estimation, that, when assumed an uniform prior distribution, is equivalent to the maximum marginal likelihood for the estimation of the item parameters. It will be presented here shortly; the interested reader may see the method in more detail in the article "Marginal Maximum Likelihood Estimation of Item Parameters: Application of an EM Algorithm" (Bock and Aitkin, 1981, p. 443). Other methods will be pointed out by the end of this Section.

Let $\xi_i^\top = (a_i, b_i, c_i)$ be the vector parameter of the i-th item, and $\xi^\top = (\xi_1^\top, \ldots, \xi_I^\top)$ the vector with the parameters of each item. Let, still, $Y^\top = (Y_1, \ldots, Y_I)$, $Y_i \in \{0, 1\}$, for $i = 1, \ldots, I$, a vector of responses from a examinee with skill θ. The probability of these answers is:

$$P(Y|\theta, \xi) = \prod_{i=1}^{I} P(Y_i = 1)^{y_i}[1 - P(Y_i = 1)]^{1-y_i}.$$

Assuming that the examinees come from a population where the ability θ is continuously distributed, with density $g(\theta)$ whose average and variance are finite, the marginal probability is given by:

$$P(y|\xi) = \int P(y|\theta, \xi)g(\theta)d\theta \tag{2.1}$$

and the marginal likelihood function,

$$L = P(y_1, \ldots, y_N|\xi) = \prod_{j=1}^{N} P(y_j|\xi),$$

where N is the amount of examinees that answered the test.

The integral in (2.1.5) is estimated by Gaussian quadrature (Stroud and Secrest, 1966). Maximizing L is the same as maximizing $\log L$, which is easier. Finally, the EM Algorithm is used to obtain the solution of the system $\frac{\partial \log L}{\partial \xi_{il}} = 0$, assuming the normal distribution for the ability, with average 0 and variance 1.

In the case of multiple groups, as in Enem, the densities $g_k(\theta)$ from each group must be incorporated, by estimating those densities' parameters together with the item parameters by choosing a reference group that will give fixed density parameters also with average 0 and variance 1, forming an $\eta_k = \{\mu_k = 0, \sigma = 1\}$ vector of parameters from the data generating distribution of this group k. The marginal likelihood is then given by:

$$L = \prod_{k=1}^{K}\prod_{j=1}^{N_k} P(y_{kj}|\xi, \eta_k) = \prod_{k=1}^{K}\prod_{j=1}^{N_k} \int P(y_{kj}|\theta, \xi)g_k(\theta)d\theta, \tag{2.2}$$

and the EM algorithm can be used in this equation for item parameters' estimation.

The likelihood can incorporate the prior distribution of item parameters $g(\xi)$, so the bayesian estimator is the value that maximizes the posterior distribution

$$g(\xi|y_1, \ldots, y_N) = \frac{P(y_1, \ldots, y_N|\xi)g(\xi)}{P(y_1, \ldots, y_N)},$$

so the optimization problem becomes maximizing

$$\log(L) + \log g(\xi), \tag{2.3}$$

which also can be done by the EM algorithm. Notice that the optimization problem becomes a Maximum a Posteriori (MAP) bayesian inference with a Gaussian prior that acts as a regularization term in the field of machine learning.

Ultimately, Expected A Posteriori (EAP) is used for the estimation of the abilities, considering the answer vector of each examinee and the previously calculated item parameters. For the examinees in the reference group, this function also has average 0 and variance 1. For the rest of the examinees in the other groups, the ability (that is, their grade) is given by:

$$\hat{\theta}_j \approx \frac{\sum_{q=1}^{Q} X_q P(y_j|X_q,\xi)A(X_q)}{\sum_{q=1}^{Q} P(y_j|X_q,\xi)A(X_q)}, \tag{2.4}$$

whose precision is measured by the posteriori standard deviation (PSD):

$$PSD(\hat{\theta}_j) \approx \frac{\sum_{q=1}^{Q} (X_q - \hat{\theta}_j)^2 P(y_j|X_q,\xi)A(X_q)}{\sum_{q=1}^{Q} P(y_j|X_q,\xi)A(X_q)},$$

where X_q is a quadrature point, and $A(X_q)$ is a positive weight corresponding to the density $g(\theta)$ at the point X_q.

EAP basically consists of a degenerate prior for the ability estimation, assuming that the estimated item parameters are the real values. The estimation of θ can be approximated by several methods other than Gaussian-Hermite quadrature, such as Markov Chain Monte Carlo (MCMC).

There are even more alternatives when one chooses to estimate items and ability parameters together, instead of integrating over the ability distribution in order to obtain a marginal likelihood. For more information, see You, H. (2022).

2.2 Deep Neural Networks

Although new optimization methods and adaptations that allow choosing from a larger set of assumed population distributions, the traditional IRT models present intrinsic limitations: local independence (McDonald, 1982) - that is, all the manifest variables (y_{ij}) are independent random variables if the latent variables ($\theta, \xi = \{a, b, c\}$) are controlled (fixed), so the conditional probability of observing a response pattern given a particular latent trait value - $P(y|\theta, \xi)$ - equals the product of the itemsâ conditional probabilities; and assumption of a data generating distribution.

Deep Neural Networks do not require any of those assumptions, and can, in fact, encompass these other methods in its own structure, as we will see later on.

They are called networks for representing a composition of many functions in a chain structure. For example, suppose three functions f_1, f_2 e f_3 connected as $f(\mathbf{x}) = f_3(f_2(f_1(\mathbf{x})))$. This feature allows them to map a path between input variables $\mathbf{x}$ and output variables $\mathbf{y}$ that best approximates the true function $\mathbf{y} = f^*(\mathbf{x})$. Of course, the neural network model is not merely approximating a mathematical function. It estimates the parameters of a probabilistic model.

Each function in this chain is a layer of the network, known as hidden layers, except for the last one, called the output layer. The learning algorithm chooses how to use those hidden layers in such a way to minimize a determined cost function for the model at hand. The term “deep" originates from this layer terminology: the more layers composing a neural network, the deeper it will be.

Lastly, each layer can have varied dimensions, that is, the amount of hidden units composing them. Each of those units compute the data in parallel, acting as functions of vectors to scalars, resembling actual biological neurons in the sense that they receive inputs from many other units (connected to each other as dendrites) and compute a new activation value. In this way, we can understand the name deep neural networks.

Notice that this solution does not require knowledge of a prior distribution nor that inputs be

independent from each other. Besides, as the model allows for the composition of many, varied functions, basically any interaction between parameters can be mapped, not being limited to linearity.

2.2.1 The 3PL Model as a MLP

The feedforward network is the exemplary deep neural network. Also known as multilayer perceptron (MLP), they are one application of the general principle of improving models by learning features. They learn mappings from input to output without feedback connections between the layers. The data flows constantly until the output layer.

To be able to seek the best mapping from input to output, the neural network must have an architecture consisting of the choice or the tuning of how many layers (how deep one wants the network to be), how many units in each layer (dimensionality), how these layers will be connected, and also of the optimizer, the cost function, the form of the output units, and the activation functions to compute each hidden layer values.

The 3PL IRT model can be written as a feedforward network, letting N be the number of examinees and I the number of items in a test.

Consider the following neural network, with just one hidden layer and a customized activation function, whose structure can be visualized in Figure 2. The ability of the second examinee corresponds exactly to the w_{21} weight, $w_{21} = \theta_2$, since w_{11} and w_{31} are multiplied by zero. The same happens to the item parameters in the multiplication of $\mathbf{W}^\top\mathbf{X}$, since there is only one element in q_3 that is not zero; so $w_{62} = a$, $w_{63} = b$, and $w_{64} = c$.

With the inclusion of another hidden layer, as in Tstsumi et al. (2021) - see Figure 3, s_j would be projected to θ_j, and q_i projected to β_i separately, eliminating the need of fixing some weights to be zero.

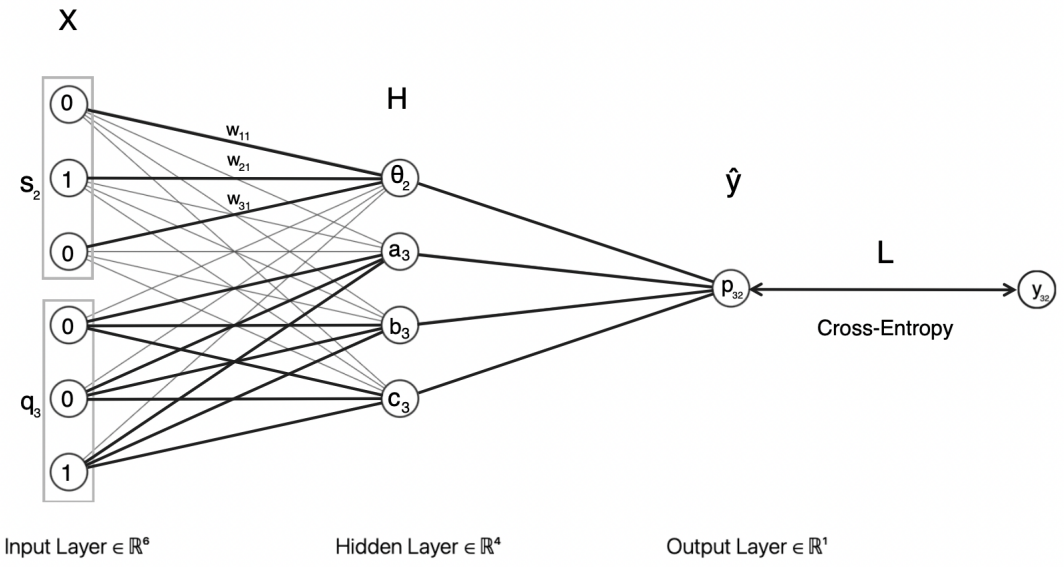


Figure 2.2: 3PL IRT model as a neural network. This visualization shows the path in the model for the second examinee ($s_2 = (0, 1, 0)$) and the third item ($q_3 = (0, 0, 1)$) in a hypothetical test with only 3 examinees and 3 items. The light grey lines represent the elements in the weight matrix which are forced to equal zero.

The input is simply concatenating two one-hot vectors q_i, $i = 1, \ldots, I$, and s_j, $j = 1, \ldots, N$, wherein the i-th and the j-th elements represent the i-th item and the j-th examinee, which are equal to 1 while the rest of the elements are just zeroes. Suppose $N = I = 3$, meaning a test with only three examinees and three items, and let us write the network for the second examinee and the third item, $j = 2$ and $i = 3$.

$$\mathbf{X'_{23}} = (\underbrace{0, 1, 0}_{s_2}, \underbrace{0, 0, 1}_{q_3})'$$

The weight matrix maps the input layer to the first hidden layer. It has a structure of $(N + I) \times (4)$, in our example, 6×4.

$$\mathbf{W} = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \\ w_{51} & w_{52} & w_{53} & w_{54} \\ w_{61} & w_{62} & w_{63} & w_{64} \end{bmatrix} = \begin{bmatrix} \theta_1 & 0 & 0 & 0 \\ \theta_2 & 0 & 0 & 0 \\ \theta_3 & 0 & 0 & 0 \\ 0 & a_1 & b_1 & c_1 \\ 0 & a_2 & b_2 & c_2 \\ 0 & a_3 & b_3 & c_3 \end{bmatrix},$$

where the circled elements are all zeroes.

So, we can write $H_{23} = W^\top X_{23}$,

$$\mathbf{H_{23}} = \begin{bmatrix} \theta_2 \\ a_3 \\ b_3 \\ c_3 \end{bmatrix}.$$

Finally, the examinee and item parameters in $\mathbf{H}$ translate to the probability of examinee j correctly answering item i through the following customized activation function:

$$p_{ij} = \hat{y}(i,j) = P(Y_{ij} = 1|\mathbf{H}) = \Phi(\mathbf{H}) = \alpha(c) + [1 - \alpha(c)]\frac{1}{1 + \exp[-a(\theta - b)]},$$

where $\alpha(.)$ is the logistic function, in order to guarantee that parameter $\tilde{c} = \alpha(c) = 1/(1 + \exp(-c))$ is within the $[0, 1]$ interval, since it represents a probability.

The cross-entropy loss function is

$$l(\hat{y}_{ij}, y_{ij}) = -[y_{ij} \log \hat{y}_{ij} + (1 - y_{ij}) \log(1 - \hat{y}_{ij})], \tag{2.5}$$

where y is the observed answer, so we can minimize

$$\sum_{i=1}^{I}\sum_{j=1}^{N} l(\hat{y}_{ij}, y_{ij}),$$

by some optimizer, in order to find the weights $\mathbf{W}$.

Notice that equation (2.2.1) has a direct correspondence to the likelihood in equation (2.1.2):

$$\begin{aligned} L = \prod_{j=1}^{N}\prod_{i=1}^{I} P(Y_{ij} = y_{ij}|\mathbf{H}) &= \prod_{j=1}^{N}\prod_{i=1}^{I} P(Y_{ij} = 1|\theta, \xi) \\ &= \prod_{j=1}^{N}\prod_{i=1}^{I} P(Y_{ij} = 1)^{y_{ij}}[1 - P(Y_{ij} = 1)]^{1-y_{ij}} \\ &= \prod_{j=1}^{N}\prod_{i=1}^{I} \Phi(\mathbf{H})^{y_{ij}}[1 - \Phi(\mathbf{H})]^{1-y_{ij}} \\ &= \prod_{i=1}^{I}\prod_{j=1}^{N} \hat{y}_{ij}^{y_{ij}}(1 - \hat{y}_{ij})^{(1-y_{ij})}, \end{aligned}$$

which can be written to equal the cross-entropy:

$$-\log(L) = -\log\left[\prod_{i=1}^{I}\prod_{j=1}^{N} \hat{y}_{ij}^{y_{ij}}(1 - \hat{y}_{ij})^{(1-y_{ij})}\right] = \sum_{i=1}^{I}\sum_{j=1}^{N} l(\hat{y}_{ij}, y_{ij}). \tag{2.6}$$

So we can see that the task of minimizing the cross-entropy function is the same as the task of maximizing the likelihood function performed in the MAP method described in section 2.1.2, more specifically, equation 2.1.3, that provides the loss function with weight decay regularization (see Section 2.2.2), except that, there, the ability parameter is estimated by EAP in another step, after the MAP estimation of item parameters.

For the task of optimization, one can choose from a multitude of methods, such as stochastic gradient descent, AdaGrad, RMSProp and Adam - see Chapter 7 and 8 of GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. Deep Learning; [S.l.]: MIT Press, 2016.

2.2.2 Regularization and Optimization

Already introduced in Section 2.1.2, when describing equation 2.1.3, regularization is an attempt of avoiding overfitting and making the algorithm perform well on new inputs. It does so by any alteration in the network design that reduces its generalization error, the error on the test set, even if it means increasing training error.

In the field of machine learning, finding the best model is often done by applying a more general, large model that has been appropriately regularized instead of a very complex model that has a right number of parameters that perfectly describes the data.

On the other hand, optimization, in the context of machine learning, involves minimizing a cost function $L(\theta)$ that can also minimize some performance measure P that we are actually interested in. And most of the time, for the problem at hand, reaching a value that is smaller than some criterion is enough, so one does not have to be concerned about reaching the global minimum. Another advantage is that optimization in machine learning typically compute each update to the parameters based on an expected value of the cost function estimated using only a subset of the terms of the full cost function, because it can be decomposed as a sum over the training examples (see equation 2.2.2). These are called batch and minibatch algorithms, which can be optimized in parallel.

Stochastic gradient descent (SGD) may be the most popular optimization algorithm, not just for deep learning, but for machine learning in general. It is an elegant and simple solution to optimize the parameters one is interested in. It works by “following" the gradient of randomly selected minibatches of the training set until a convergence criterion is satisfied. Many algorithms are modifications to the SGD in the crucial parameter that is the learning rate, which can have momentum to accelerate learning (e.g., Nesterov momentum) and even by changing learning rates for each parameter in the adaptative moments techniques (AdaGrad, RMSProp, Adam). Other optimization algorithms involve approximate second-order methods and many strategies can be applied to these algorithms to try and make them more efficient, such as batch

normalization and averaging.

There are some challenges in neural network optimization. Ill-conditioning is one of them, characterized by a very slow learning process despite the presence of a strong gradient because the learning rate must be shrunk to compensate for an even stronger curvature. Other challenges involve plateaus, saddle points and other flat regions in a cost function. When a network has many layers, there may be cliffs and exploding gradients. But we do not expect to encounter such complex challenges in the optimization of Deep IRT.

2.3 Deep-IRT

Though innovative in its conception, Deep IRT research still did not achieve interpretable parameters for the examinee's skill and item difficulty, crucial factor in the field of Test Theory. The proposed models of deep knowledge tracing were aimed at the estimation of time series changes in the skill of a group of examinees. With the model proposed by Tsutsumi et al. (2021), Deep IRT is presented as a new test theory, seeking the estimation of item parameters and examinees' skills independently, in order to achieve the interpretability needed to actually grade the test.

The new model showed that its precision when estimating examinees' skills is greater than traditional IRT when those skills do not come from a same data generating distribution and when there are no common items between the tests that were applied. In other words, Deep IRT overcomes the assumptions that the parameter be identically distributed and locally independent.

The Deep IRT model proposed by Tsutsumi et al. (2021) is composed by two neural networks, one for the item difficulty parameter and the other for the examinee ability parameter. This way, the parameters will still have interpretability with the possibility of higher accuracy in their estimation.

The examinee network, described in Figure 3 as Examinee Layer, has two hidden layers. To

represent the j-th examinee from a total of J examinees, the input is a one-hot vector $\boldsymbol{s}_i$, where the element j is equal to one and the other $J-1$ elements are zero. The layers are given by

$$\begin{aligned}
\boldsymbol{\theta}_{\mathbf{1}}^{(i)} &= \tanh\left(\boldsymbol{W}_{\boldsymbol{\theta_1}}\boldsymbol{s}_j + \boldsymbol{\tau}_{\boldsymbol{\theta_1}}\right); \\
\boldsymbol{\theta}_{\mathbf{2}}^{(i)} &= \tanh\left(\boldsymbol{W}_{\boldsymbol{\theta_2}}\boldsymbol{\theta}_{\mathbf{1}}^{(i)} + \boldsymbol{\tau}_{\boldsymbol{\theta_2}}\right); \\
\theta_3^{(i)} &= \boldsymbol{W}_{\boldsymbol{\theta_3}}\boldsymbol{\theta}_{\mathbf{2}}^{(i)} + \tau_{\theta_3},
\end{aligned}$$

where the activation function is the hyperbolic tangent

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)}.$$

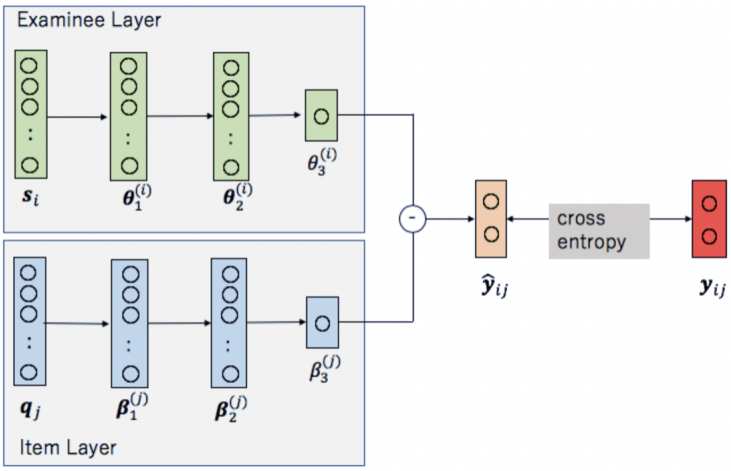


Figure 2.3: Deep-IRT.

The weight matrices in the hidden layers are

$$\mathbf{W}_{\theta_1} = \begin{bmatrix} w_{\theta_1}^{(11)} & w_{\theta_1}^{(12)} & \dots & w_{\theta_1}^{(1I)} \\ w_{\theta_1}^{(21)} & w_{\theta_1}^{(22)} & \dots & w_{\theta_1}^{(2I)} \\ \vdots & \vdots & \ddots & \vdots \\ w_{\theta_1}^{(|\theta_1|1)} & w_{\theta_1}^{(|\theta_1|2)} & \dots & w_{\theta_1}^{(|\theta_1|I)} \end{bmatrix},$$

where $|\theta_k|$ denotes the length of the θ_k vector,

$$\mathbf{W}_{\boldsymbol{\theta_2}} = \begin{bmatrix} w_{\theta_2}^{(11)} & w_{\theta_2}^{(12)} & \dots & w_{\theta_2}^{(1|\theta_1|)} \\ w_{\theta_2}^{(21)} & w_{\theta_2}^{(22)} & \dots & w_{\theta_2}^{(2|\theta_1|)} \\ \vdots & \vdots & \ddots & \vdots \\ w_{\theta_2}^{(|\theta_2|1)} & w_{\theta_1}^{(|\theta_2|2)} & \dots & w_{\theta_2}^{(|\theta_2||\theta_1|)} \end{bmatrix},$$

and the weight vector in the output layer is

$$\mathbf{W}_{\boldsymbol{\theta_3}} = \left(w_{\theta_3}^{(1)}, w_{\theta_3}^{(2)}, \dots, w_{\theta_3}^{(|\theta_2|)} \right).$$

Also, $\boldsymbol{\tau_{\theta_1}}$ and $\boldsymbol{\tau}_{\theta_2}$ are the bias parameter vectors, and τ_{θ_3} is the bias parameter.

The item layer is analogous, with the $\mathbf{q}_j$ one-hot vector input and layers

$$\boldsymbol{\beta}_{\mathbf{1}}^{(j)} = tanh\left(\boldsymbol{W}_{\boldsymbol{\beta_1}} \boldsymbol{q}_i + \boldsymbol{\tau}_{\boldsymbol{\beta_1}}\right); \tag{2.7}$$

$$\boldsymbol{\beta}_{\mathbf{2}}^{(j)} = tanh\left(\boldsymbol{W}_{\boldsymbol{\beta_2}} \boldsymbol{\beta}_{\mathbf{1}}^{(j)} + \boldsymbol{\tau}_{\boldsymbol{\beta_2}}\right); \tag{2.8}$$

$$\beta_3^{(j)} = \boldsymbol{W}_{\boldsymbol{\beta_3}} \boldsymbol{\beta}_{\mathbf{2}}^{(j)} + \tau_{\beta_3}. \tag{2.9}$$

Then, the output from both networks is the input for the hidden layer $\hat{y}_{ij}$, describing the probability of examinee j correctly answering item i:

$$\begin{aligned} \mathbf{h}_{ij} &= \mathbf{W}_y^\top (\theta_3^{(i)} - \beta_3^{(j)}), \\ \hat{y}_{ij} &= \text{softmax}(\mathbf{h}_{ij}) \\ &= \frac{1}{1 + \exp(\mathbf{h}_{ij})}. \end{aligned}$$

Here, $\mathbf{W}_y = (w_{y_1}, w_{y_2})$, which can be related to the function of the discrimination parameter a in the 2PL IRT model, with the restriction that, here, it is not related to each item, so it is a general parameter a for all the items.

This model, called Deep-IRT, does not assume any distribution for examinees' abilities neither for items' parameters. It estimates the relation of one examinee's ability to all the others', which is done in the second to the third layer, by maximizing the prediction accuracy of the responses. Consequently, this method does not require linkage to different tests by having common items between them.

The cross-entropy function is the same of equation (2.2.2) but with a cost sensitive approach that weighs some small number of data over majority, because Deep-IRT would not be able to make accurate predictions when the data has small numbers of correct or incorrect answers. So, the loss function for Deep-IRT is given by

$$\begin{aligned}
\text{Loss} &= \sum_i \sum_j l(\hat{y}_{ij}, y_{ij}) \\
&= \gamma_1 \sum_{i \in L_e} \sum_{j \in u_{ij}=1} l(\hat{y}_{ij}, y_{ij}) \\
&= \gamma_2 \sum_{i \in H_e} \sum_{j \in u_{ij}=0} l(\hat{y}_{ij}, y_{ij}) \\
&= \gamma_3 \sum_{i \in L_i} \sum_{j \in u_{ij}=1} l(\hat{y}_{ij}, y_{ij}) \\
&= \gamma_4 \sum_{i \in H_i} \sum_{j \in u_{ij}=0} l(\hat{y}_{ij}, y_{ij}),
\end{aligned}$$

where L_e stands for a group of examinees whose correct answer rates are less than α_{L_e}, H_e denotes a group of examinees whose correct answer rates are more than α_{H_e}, L_i is a group of items of which correct answer rates are less than α_{L_i}, and H_i represents a group of items with correct answer rates that are more than α_{H_i}. Here, γ_1, γ_2, γ_3, γ_4 and α_{L_e}, α_{H_e}, α_{L_i}, α_{H_i} are tuning parameters. Because of the fact that Enem has many respondents to each item, this cost sensitive approach may not be necesssary, so the loss function is simply Loss $= \sum_i \sum_j l(\hat{y}_{ij}, y_{ij})$. Adaptative moment estimation (Adam) is in charge of learning all parameters simultaneously through this loss function.

2.4 Positional Encoders

Positional encoding describes the location or position of an object within a sequence in order to assign an unique representation to each position. There is a reason why a single number, such as the index value, is not used to represent an itemâs position in transformer models. For long sequences, the indices can grow large in magnitude, making it impractical or computationally expensive.

Transformers, or positional encoders, use a clever positional encoding scheme, where each position/index is mapped to a vector. Therefore, the output of the positional encoding layer is a matrix, where each row of the matrix represents an encoded object of the sequence summarised with its positional information (VASWANI, A. et al. Attention is all you need. In: Advances in neural information processing systems. [s.n.], 2017. p. 5998â6008).

The positional encoders are the result of the following:

$$P(k, 2i) = \sin\left(\frac{k}{n^{2i/d}}\right), \tag{2.10}$$

$$P(k, 2i+1) = \cos\left(\frac{k}{n^{2i/d}}\right). \tag{2.11}$$

Here,

k: position of an object in the input sequence,

d: dimension of the output embedding space,

$P(k, j)$: position function for mapping a position in the input sequence to index of the positional matrix,

n: user-defined scalar, set to 10,000 - VASWANI, A. et al. (2017),

i: mapping of column to indices, with a single value to both sine and cosine functions.

In the above expression, one can see that even positions correspond to a sine function and odd positions correspond to cosine functions.

2.5 Enem

This section aims to present and gather the information about the tests used in Enem - their structure, scale, etc. - and the estimation process conducted by Inep. It basically follows the description in Section 2.2, where the integral in (2.1.5) is estimated by Gaussian quadrature (Stroud and Secrest, 1966) - Inep uses 40 quadrature points. The normal distribution is the prior for all the parameters. Also, it is not divulged which would be the “reference group" that had its parameters made to be fixed in the Institute’s estimation, regarding equations 2.1.2 and 2.1.3. All of these procedures are conducted by Anísio Teixeira Institute in the paid software BILOG, which has a license costing US$ 10,475.00 annually.

Regarding the scale of the sutdents’ grade, in Enem, the reference value for the ability is 500, that is, the average skill of a student from that year’s test, with a standard deviation of 100. Being N the number of students, $\theta_1, \ldots, \theta_N$ are shifted in this manner: $a(\theta - b) = (a/100)[(100\theta + 500) - (100b + 500)]$. Therefore, $\theta = 100\theta + 500$, $b = 100b + 500$ and $a = a/100$. In this scale, a student whose grade is 600 is one standard deviation unit from the average skill.

Enem 2022 microdata is available at Inep’s website and it is composed by two files. One is a table of 76 columns and 3,389,832 observations representing each student applying for Enem that year; the variables of interest are the ones containing the students’ answers for each item and the respective answer key for each of the 4 knowledge areas in which the exam is divided - so we actually have 4 different exams - and the rest of the columns are only sociodemographic questions. The second file is also a table, but with a smaller size of 810 observations by 14 columns, containing information about every item used in that year’s Enem, including their position in each version of the test (“caderno de prova") and the item parameters.

Chapter 3

Proposed Methods

Seeking to build up on the previous models reviewed, this work proposes two neural networks to treat binary data originated from a test, attempting to maintain the usefulness of traditional IRT, but making use of the developments reached in the field of deep learning.

The first model is shallow in the sense that it was designed to provide a faster and simpler connection of input data to output data, while the deep model has an intuitive scalable structure, by simply increasing the number of layers and neurons after the input data.

Both models present different ways of receiving input data, one making use of embedding layers, while the other applies positional encoders, as shown in the next Sections.

3.1 Shallow 3PL

Comprising the key aspect of this model, the embedding layers seem to be an ideal solution, because they provide a direct relation of the examinee to its ability, and the item to its three parameters, working as a look-up table. So the interpretability of the IRT paradigm is maintained, extended from the Deep-IRT model of Tsutsumi et al. (2021), that could only reach a general item discrimination parameter (Section 2.3).

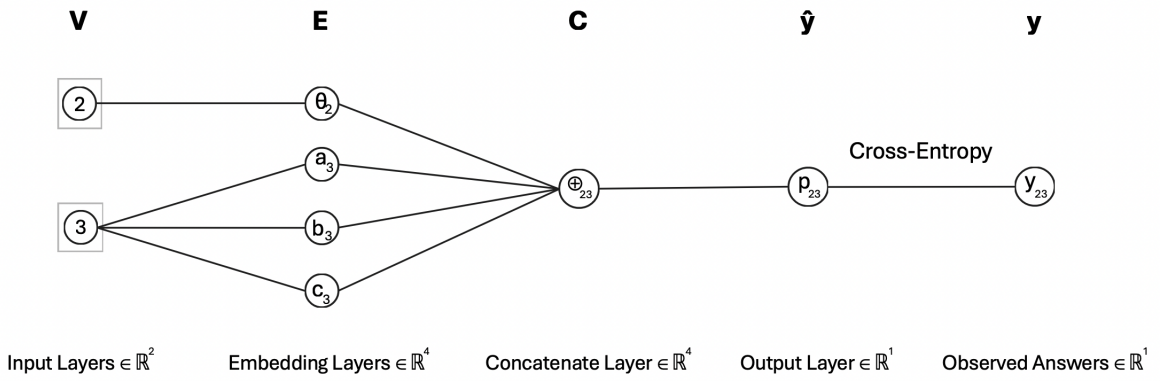


Figure 3.1: Shallow 3PL model. This visualization higlights the path in the model for the second examinee ($v_s = 2$) and the third item ($v_q = 3$) in a hypothetical test with only 3 examinees and 3 items.

With N examinees and I items, the inputs for the examinee and item networks are one-to-one vectors:

$$\mathbf{V} = \{v_s, v_q\}, \tag{3.1}$$

$$v_s = i, i = \{1, 2, \ldots, N\}, \tag{3.2}$$

$$v_q = j, j = \{1, 2, \ldots, I\}, \tag{3.3}$$

embedded to:

$$\mathbf{E} = \{e_s, e_q\}, \tag{3.4}$$

$$e_s = \theta_i, \tag{3.5}$$

$$e_q = \{a_j, b_j, c_j\}, \tag{3.6}$$

The concatenation layer $\mathbf{C} = \{\theta_i, a_j, b_j, c_j\}$ is necessary so we can use the customized

activation function for the output $\hat{\mathbf{y}}$:

$$\hat{y}_{ij} = c_j + (\mathbf{1} - c_j)\text{softmax}(a_j(\theta_i - b_j)) \tag{3.7}$$

$$= c_j + (\mathbf{1} - c_j)\frac{1}{1 + \exp(a_j(\theta_i - b_j))}. \tag{3.8}$$

The loss is the same cross-entropy function explicited in equation 2.2.2, and is the one to be optimized by the Adam algorithm.

3.2 Deep 3PL

The proposed deep network is similar to the Deep-IRT network structure and equations (see Section 2.3), but with changes in the input of the examinee network, in the number of neurons of the output of the item network, and in the activation function.

The model proposed in Tsusumi et. al (2021) does not assume normality, but it was not designed for a large dataset. So instead of one-hot vectors, the input of the examinee network is composed by positional encoders, in such a way that each student is represented by a unique vector that has a practical size, say d. That way, the input is not a million $\times$ million matrix, but a million $\times$ d matrix (VASWANI, A. et al. Attention Is All You Need, 2023).

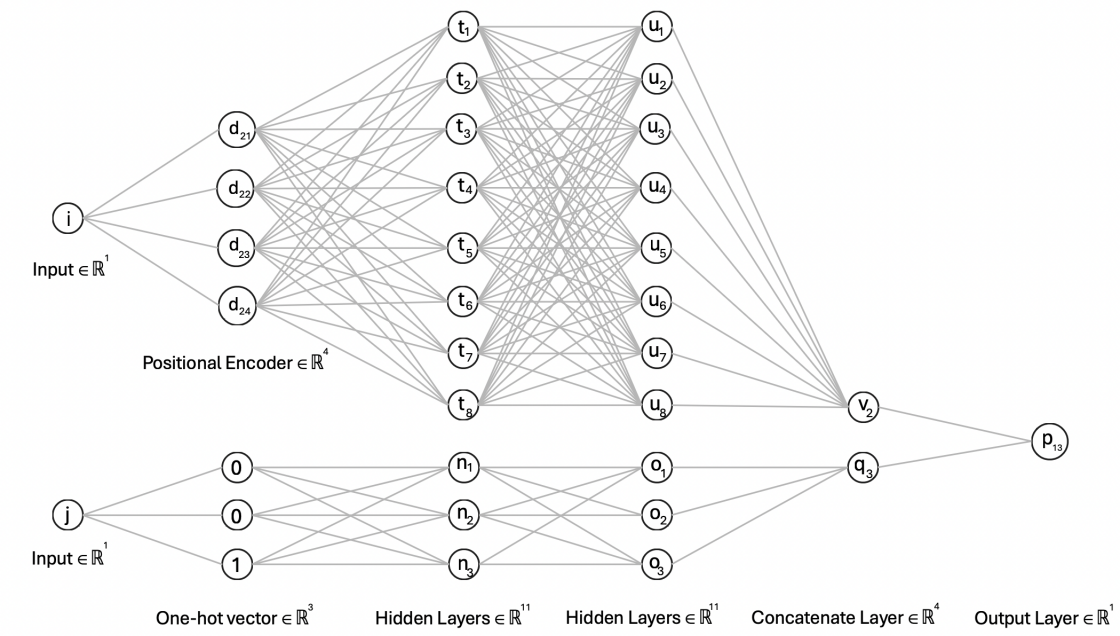


Figure 3.2: Deep 3PL model. This visualization shows the path in the model for the i-th examinee and j-th item in a hypothetical test with only 3 examinees and 3 items. The examinee network has dimension $d = 4$ for the positional encoder, 8 neurons in both densely connected layers.

The item network would keep its one-hot vectors inputs, since it leads only to an $I \times I$ matrix, with two fully connected hidden layers and hyperbolic tangent activation functions, as in equations (2.3.1) and (2.3.2). But the final layer does not output only one parameter β_3 to be concatenated with the single output θ_3 of the examinee network. It outputs three parameters, whose weights should work as the item parameters in the 3PL IRT model, so equation (2.3.3) would be: $q^{(j)} = \boldsymbol{W_q}\boldsymbol{\beta_2}^{(j)} + \tau_q$.

And the activation function would be customized:

$$\hat{y}_{ij} = w_{q_3} + (\mathbf{1} - w_{q_3})\text{softmax}(w_{q_1}(\theta_3^{(i)} - w_{q_2})) \tag{3.9}$$

$$= w_{q_3} + (\mathbf{1} - w_{q_3})\frac{1}{1 + \exp(w_{q_1}(\theta_3^{(i)} - w_{q_2}))}. \tag{3.10}$$

The three item weights $(w_{q_1}, w_{q_2}, w_{q_3})$ and the student ability weight $(\theta_3^{(i)})$ are joined in a

concatenation layer and then passed to the final layer $\hat{y}_{ij} = p_{ij}$.

The network has to reach a significant size so that the weights can actually be identified to item parameters and the ability parameter, because there is not a direct correspondence of network weights from the densely connected layers to IRT model parameters as the embedding layers do in Shallow 3PL. In the example of Figure 5, there are $4 \times 8 + 8 \times 8 + 8 \times 1 = 104$ weights in the examinee network. Notice that if there were, say, 1,000 examinees, this structure would not nearly be enough to represent their respective ability parameters. This indicates a difficulty when the model is met by a very large sample size.

Chapter 4

Application

Besides applying a new estimation method for Enem candidates' grades (proportional to their abilities), it is in the scope of this project investigating the calculation performed by Inep. All the information made available by the institute is presented and, from that, we sought to replicate the results of the previous exam in an open software.

The preprocessing of the data was somewhat toilsome, because the items had to be aggregated according to their knowledge area (Nature Sciences, Mathematics, Human Sciences, and Languages), and the answers labeled to 0 or 1 (if correct) according to the answer key, which resulted in 4 tables with about 100 columns representing the items.

Then, Shallow 3PL and Deep 3PL are applied to the same Enem data, and the precision of each method and its impact on the ranking/classification of the examinees are compared.

4.1 Inep's estimation process

Prior to the estimation of Enem candidates' abilities, the estimation of item parameters is not conducted using the national exam itself, but using information from groups of students whose schools were previously selected for the so called pre-tests, gathering it in equation (2.1.2) and using the EM algorithm.

Having obtained items' difficulty, discrimination, and guessing parameters, Inep, by some undisclosed internal set of criteria, selects some items from the pre-tests to be part of the National Bank of Items (BNI).

From this, students abilities are estimated using actual Enem data with fixed item parameters, using equation 2.1.3. At this point, there is another frail step in the estimation, because Inep does not inform which is the reference group, that is, the group of examinees that were used to solve the indetermination of parameters from multiple groups by fixing the mean to 0 and the variance to 1.

4.2 Alternative estimation approaches

In order to present an accessible, public, transparent alternative, it is part of this project estimating Enem applicants' grades given by 3PL IRT model using a free and open code software. In this way, the researchers and organizations could replicate the estimation of the applicant's grades, which today is, sadly, far from reality, where anxious students seeking to enter in prestiged universities must blindly trust the results given by Inep, since reference groups identification and the software program used are not public.

Then, two innovative models complete this work, modernizing the application of IRT and striving to seek greater efficiency in the estimation process, by taking advantage of the development of deep learning methods.

4.2.1 3PL in Mirt

First, a data pre-processing relatively toilsome was needed for the Enem data available. Then, R software provides implemented packages for a flexible IRT analysis that allows the choice of the options in the functions' arguments according to the information that is actually disclosed. The *mirt* package was chosen among others tested, *irtoys*, *est* and *ltm*, due to its robustness and flexibility.

4.2.2 Shallow 3PL

As part of this open code estimation bridging the proposed estimation model, R interface with Python machine learning *keras* package was used to build the Shallow 3PL IRT model. Then, the Adam algorithm could be used to estimate the parameters so there is more freedom and simplicity in the estimation process, since Inep uses the MAP method described in Section 2.1.2, using complicated numerical integration with the assumption of normality for marginal distributions.

4.2.3 Deep 3PL

The Deep 3PL model was also built using *keras*. In order to achieve the relation of one studentâs ability to all the othersâ, thus dispensing an assumed underlying data generation distribution (such as the normal distribution that Inep assumes), the formulas 2.4.1 and 2.4.2 were used to obtain the positional encoders for each student. Then, many different architectures, with varying numbers of neurons in the hidden layers, were tested, seeking the maximum prediction accuracy of the students' responses, measured by the loss function (2.2.2) and mean squared error.

4.3 Results

This section presents the performance comparison between the two traditional IRT model estimation approaches (*BILOG*, and *mirt*) and then between them and the proposed Shallow and Deep IRT models, using dispersion plots and Spearman correlation coefficients, as well as a calibration plot designed to present and compare the accuracy of predictions for each method. Rank correlation was also calculated, but showed very similar values, equal to the second decimal of the correlation coefficients.

The test whose results are portrayed in this Section is the *Natural Sciences* test in Enem. The other areas - Mathematics, Languages and Human Sciences - had similar results for the

estimation conducted in *mirt*. It must be pointed out that the differences between the results of the estimation process conducted in *mirt* and the one conducted by Inep in BILOG do not come only from this circumstance. Without access to the so called pre-tests, the item parameters were estimated using data from the actual Enem test, so the estimation was conducted on completely different populations.

The 3PL IRT model in BILOG and its replication in mirt

Here we present the very well known and widely used 3 Parameter Logistic Item Response Theory Model, as reviewed in Section 2.1.1, applied to the analysis of the Enem national exam, comparing the performance obtained from the use of *mirt* and the official estimation conducted by Inep with BILOG.

The item parameters estimates were only somewhat close regarding the difficulty b, as seen in Table 4.1 and in the dispersion plot of Figures 6(a), with a correlation of 89.99% between BILOG and *mirt*. In the case of parameters a and c the correlation observed was much weaker - Figure 4.1(b).

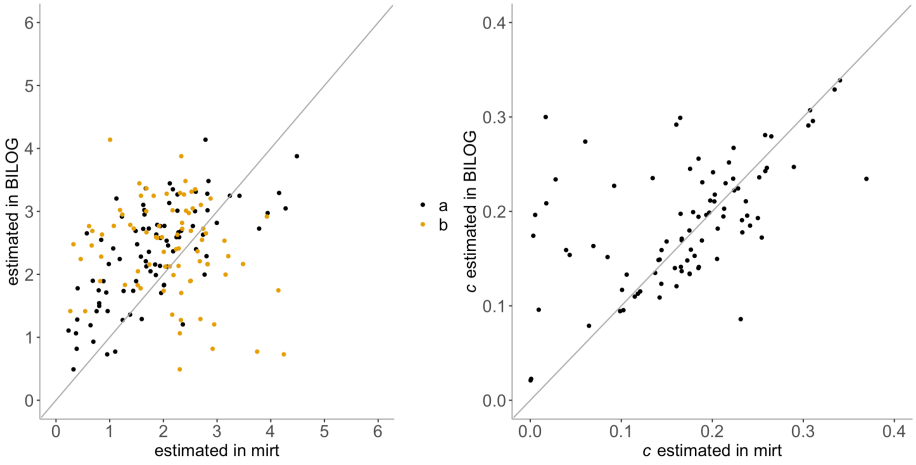
Nonetheless, the estimation of the abilities θ in *mirt* was close to its estimation in BILOG - Figure 4.1(c) - with a correlation of 98.95%.

The dispersion plot of the averaged probabilities p_{ij} - Figure 4.2(d) - indicates calibration issues with *mirt* and BILOG. Their estimated probabilities lines followed a close path to each other, initially overestimating the probability of correct answers in values below 0.7 and then underestimating them. The respective correlations between estimated and empirical probability of a correct answer are given in Table 4.3.

There are noteworthy results when the item parameters are not estimated in R, but fixed using the ones provided by Inep. Given *mirt* could only converge to a certain sample size, with a sample of one million students the correlation between θ estimated in *mirt* and estimated in BILOG achieved 99.93%. This indicates that, if the open code estimation could be conducted in the same pre test populations, perhaps the results would be even more similar, with the

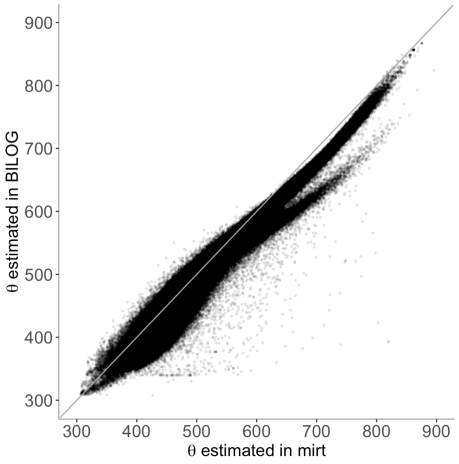
possibility of even greater accuracy by using *mirt*.

Table 4.2 and Table 4.4 shows how students' performance is affected according to the estimation procedure used. Their ranking in *mirt* is only diminished in one position until the sixth best one, but with many alterations in the lower grades, as seen in Figure 4.3(a). The worst classified examinee by Inep would have its ranking improved in 457,396 positions by the estimation conducted in *mirt*.

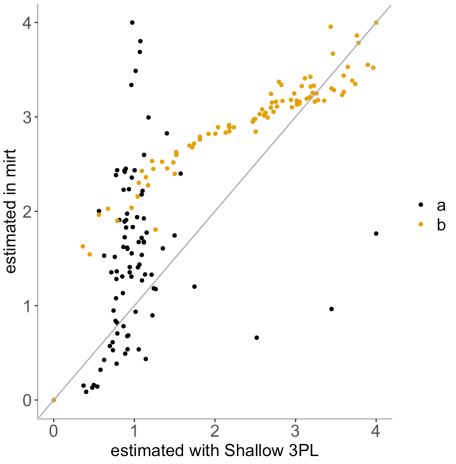


(a) Dispersion plot of a and b.

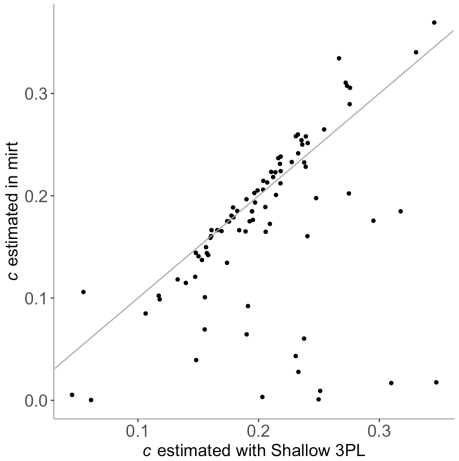
(b) Dispersion plot of c.



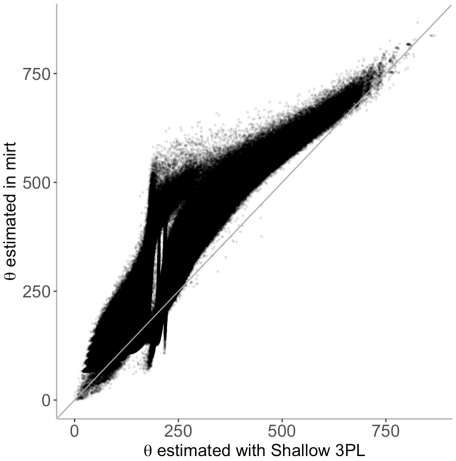
(c) Dispersion plot of θ.



(d) Dispersion plot of a and b.



(e) Dispersion plot of c.



(f) Dispersion plot of θ.

Figure 4.1: Dispersion between Inep's estimation and the estimation conducted in *mirt* of: (a) item parameter discrimination a; and item parameter difficulty b; (b) item "guessing" parameter c; (c) student's grade, obtained by a linear transformation of their estimated abilities θ, given by $100\theta + 500$ (see Section 2.1.1). Dispersion between *mirt* estimation and the Shallow 3PL estimation of: (d) item parameter discrimination a; and item difficulty parameter b; (e) item "guessing" parameter c; (f) student's grade.

Comparison	ρ_a	ρ_b	ρ_c	ρ_θ
BILOG vs. mirt	67.12%	89.99%	53.89%	98.95%
BILOG vs. Shallow 3PL	26.06%	89.92%	62.07%	92.45%
mirt vs. Shallow 3PL	17.94%	91.49%	46.72%	94.19%

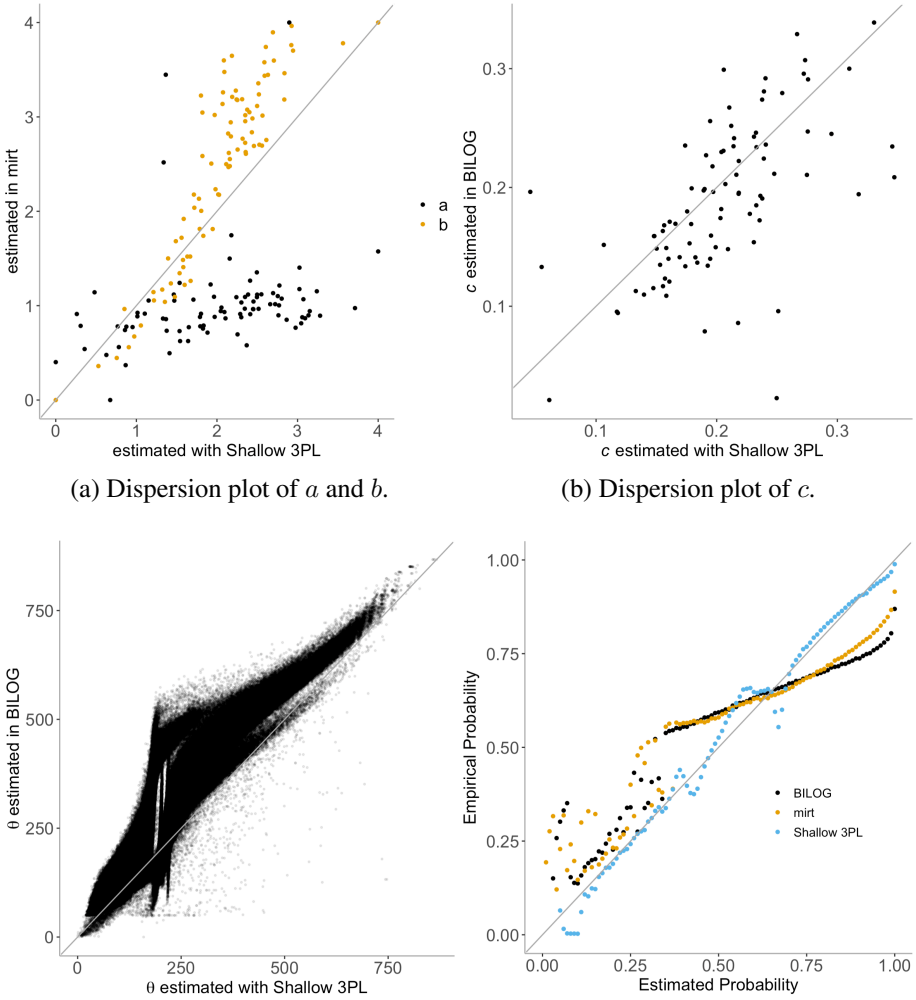
Table 4.1: Comparison of the traditional 3PL methods applied to the estimation of Enem, with the correlation coefficients between BILOG, *mirt* and Shallow 3PL.

student	$\theta_{Shallow3PL}$	θ_{mirt}	θ_{BILOG}	Rank$_{mirt}$	Rank$_{Shallow3PL}$
1	860.87	837.12	867.10	2	3
2	862.23	837.12	867.10	3	2
3	857.50	837.12	867.10	4	5
4	858.53	837.12	867.10	5	4
5	867.10	837.12	867.10	6	1
6	821.26	828.23	854.24	7	7
7	804.48	817.17	850.83	11	26
8	807.91	817.17	850.83	12	16
9	803.54	817.17	850.83	13	27
10	803.12	817.17	850.83	14	29

Table 4.2: Grades and ranking of the ten best students according to Inep's calculation through different estimation methods.

Method	loss	mse	Correlation with empirical probability
BILOG	59.79%	18.74%	94.55%
mirt	57.85%	18.71%	95.37%
Shallow 3PL	$\mathbf{53.25}\%$	$\mathbf{17.62}\%$	99.26%
CTT	58.08%	19.71%	$\mathbf{99.99}\%$

Table 4.3: Performance of all the methods applied to the estimation of Enem, the CTT being the simplest model possible of the Classic Test Theory, that is, merely the sum of correct answers.



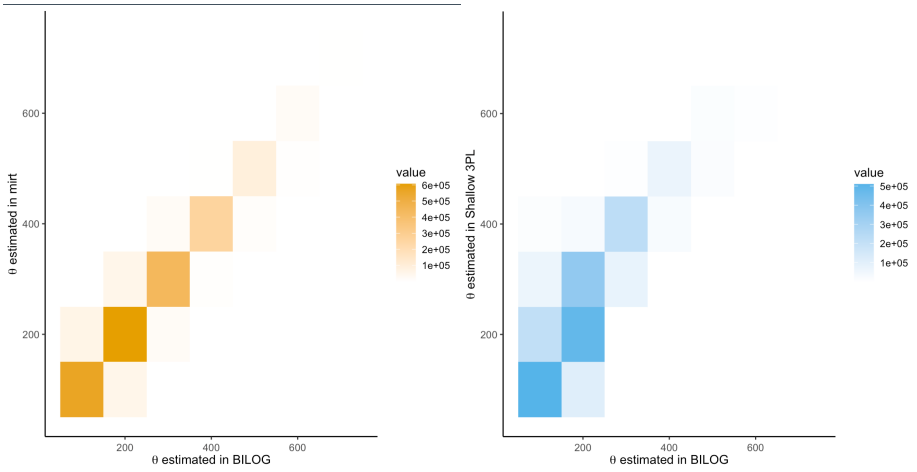
(a) Dispersion plot of a and b.

(b) Dispersion plot of c.

(c) Dispersion plot of θ.

(d) Dispersion plot of $\hat{y}_{ij}$ by y_{ij}.

Figure 4.2: Dispersion between Inep's estimation and the Shallow 3PL estimation of: (a) item parameter discrimination a; and item "guessing" parameter c; (b) item parameter difficulty b; (c) student's grade; and (d) dispersion plot of estimated probabilities by empirical probabilities, by rounding each p_{ij} to the second decimal, then using these rounded values as subsets of the probability dataset in which each of these subsets the average of the binary responses was calculated, thus obtaining the empirical probabilities.



(a) $\theta_{BILOG} \times \theta_{mirt}$. (b) $\theta_{BILOG} \times \theta_{Shallow3PL}$.

Figure 4.3: Confusion matrices of abilities between Inep's estimation and the estimation conducted in (a) *mirt* and (b) Shallow 3PL.

student	$\theta_{Shallow3PL}$	θ_{mirt}	θ_{BILOG}	Rank$_{mirt}$	Rank$_{Shallow3PL}$
1	9.15	13.31	2.01	2245361	2245425
2	10.64	7.67	1.86	2245396	2245415
3	9.16	13.75	1.86	2245359	2245424
4	12.31	3.79	1.70	2245428	2245407
5	10.87	11.01	1.55	2245373	2245413
6	8.75	4.05	1.55	2245426	2245427
7	50.64	63.97	1.39	2244174	2222905
8	9.81	18.18	1.08	2245321	2245417
9	3.44	0.00	0.15	2245445	2245442
10	159.36	177.17	0.00	1788049	1855830

Table 4.4: Grades and ranking of the ten worst evaluated students according to Inep's calculation through different estimation methods.

4.3.1 The proposed models: Shallow 3PL and Deep 3PL

Now, we turn our focus to the estimation provided by the two proposed models, as presented in Section 3.1 and 3.2, applied to Enem data, with the performance comparison to the traditional 3PL. First, the results achieved by the Shallow 3PL model are shown, and afterwards, the results that could be achieved with a limited version of the Deep 3PL model.

For the Shallow model, as in the case between BILOG and *mirt*, the item parameters estimates were close regarding the difficulty b, as seen in Table 4.1 and in the dispersion plot of Figure 4.1(e), reaching a correlation of 91.49% between *mirt* and Shallow 3PL. In the case of parameters a and c the correlation observed was also weaker - Figure 4.1(d).

The comparison of the ability estimation by the Shallow 3PL model and by the 3PL model can be seen in Figures 6(f) and 7(c), respectively showing a 92.45% correlation between Shallow 3PL and *mirt* and a 92.83% correlation between Shallow 3PL and BILOG.

The calibration issue was solved by using the Shallow 3PL model, observed in the dispersion plot of the averaged probabilities p_{ij} - Figure 4.2(d). The estimated probabilities follow the empirical probabilities, meaning that the model is closer to the actual response of the examinees throughout the ability scale, with only some moderate undulations in the average abilities, reaching an outstanding 99.26% correlation to the averaged observed responses - Table 4.3.

Table 4.2, and specially Table 4.4, and Figure 4.3(b) show a greater desynchronization between the examinees' official classification and that given by the Shallow 3PL model. The worst classified examinee by Inep would have its ranking improved by 389,615 positions by the Shallow 3PL model.

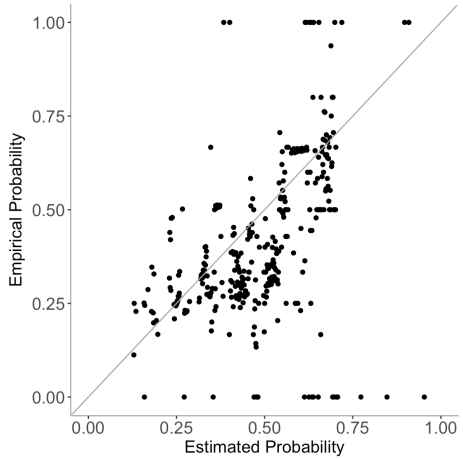
The Deep 3PL model, as explained in Section 3.2, requires a massive amount of weights so it can accurately represent the ability parameter θ for all the examinees. Unfortunately, a model with enough layers and neurons could not be achieved with current computational resources available to this research. It must be pointed out that the model has the potential of greater performance if its structure can be expanded.

Therefore, the proposed Deep 3PL model was tested on a restricted variation of hyperparameters' values: ($d = 10$; $d = 12$; $d = 20$; $d = 28$), for the positional encoders; (128, 264, 512) number of neurons in the second layer; and (64, 128, 264, 512) number of neurons in the third layer. Perhaps the greatest restriction was not being able to work with the whole population of examinees. Various samples were tested, with different sizes ($n = 100000$; $n = 200000$; $n = 400000$; and $n = 500000$), but half a million examinees was the greatest sample that could

be run through the model.

With all these aspects in mind, considering the 59.56% of loss value and 20.32% of MSE (even smaller loss than BILOG's), the proposed Deep 3PL model performance was subpar, showing great discalibration, as seen in Figure 4.4.

The model was run with only half a million students and limited number of neurons: $22, 128, 128, 1$ for the examinee network, so there was no way to identify weights to their respective abilities, with only $22 \times 128 + 128 \times 128 + 128 \times 1 = 19328$ weights to be mathed to the 500000 examinees. Other combinations of hyperparameters showed similar results.



(a) Dispersion plot of $\hat{y}_{ij}$ by y_{ij}

Figure 4.4: Dispersion plot of estimated probabilities by empirical probabilities.

Chapter 5

Conclusions

This work was initiated as an attempt to contribute to the task of evaluating the latent trait of knowledge in a given population that answered a given set of questions in an exam. For such purpose, the National High School Exam (Enem) of Brazil was analyzed.

The process applied by the official government institute responsible for the exam's correction and evaluation (Inep) was studied and replicated in an open language environment. Using *mirt* the 3PL IRT for Enem was estimated as closely as possible to the actual process Inep applies, with the EM algorithm and numerical approximations.

Then, with *keras*, state of the art machine learning algorithms and techniques (such as embedding layers) were used in the optimization of the maximum likelihood estimation. A new IRT calculation approach was proposed: Shallow 3PL.

This proved to be a very successful endeavour, since the precision and calibration of the Shallow 3PL was superior to that of Inep (using proprietary software *BILOG*) and even than the replication performed in *mirt*, as seen in Section 5.3. And this was achieved without the assumption of a normal distribution, in fact, without assuming any underlying data generation distribution neither local independence, thanks to the estimation of the students' abilities interconnected with all the others, characteristic found in Tsutsumi's model (Section 2.3), but applied differently in this case since embedding layers were used to process the input data. So

a more robust and more flexible estimation method was found to suit Enem data.

Despite the incipient practical application, the proposed Deep 3PL model also poses as a new approach to the calculation of IRT, using sophisticated solutions (as the positional encoders) to free ourselves from the traditional assumptions. We hope to achieve better results when the whole population can be analyzed by the model and the number of neurons and layers can be expanded.

Also for future works, simulation studies with diverse tests (with varied numbers of students and items) can be done, as in Tsutsumi et al. (2021), so there is complete knowledge of how the new models will behave in different contexts.

Perhaps, the most relevant next step is fixing the item parameters in the Shallow 3PL estimation, so this new method can be suggested as the new calculation of Enem examinees' grades.

Finally, it was made clear that the process applied to attribute students' grades does not guarantee the best likelihood to their actual knowledge, and how the evaluation of their abilities varies depending on the estimation method, or even by varying the hyperparameters in a given method. We hope this work inspires more transparency in the important social process of examination and testing of individual abilities.

Appendix A

Description of the marginal maximum likelihood method

Let $\xi_i^{'}$ be the vector of parameters of the *i*-th item, and $\xi^{'} = (\xi_1^{'}, \ldots, \xi_m^{'})$ the vector with all the item parameters.

Let, still, $U^{'} = (U_1, \ldots, U_m)$, $U_k = 0$ or 1, a vector of answers of an examinee with ability θ. The probability of ocurrence of this answer vector is:

$$P(u|\theta, \xi) = \prod_{k=1}^{m} P_{k1}^{u_k}(1 - P_{k1})^{1-u_k}.$$

Assuming that the exmaminees belong to a population whose ability θ is continually distributed, with density $g(\theta)$ and finite mean and variance, the marginal probability is

$$P(u|\xi) = \int P(u|\theta, \xi)g(\theta)d\theta$$

and the marginal likelihood function,

$$L = P(u_1, \ldots, u_N|\xi) = \prod_{k=1}^{N} P(u_k|\xi),$$

where N is the amount of examinees that answered the exam.

The integral above is estimated by the Gaussian quadrature formula:

$$P(u|\xi) \approx \sum_{q=1}^{Q} P(u|X_q, \xi)A(X_q),$$

where X_q is a quadrature point and $A(X_q)$ is the weight corresponding to the density function $g(\theta)$ at X_q (Stroud and Sechrest, 1966). Inep uses 40 quadrature points to Enem.

The task of maximizing L becomes simpler when done in $logL$, which results in equivalent value. This can be done from the following system:

$$\frac{\partial logL}{\partial \xi_{il}} = \sum_{q=1}^{Q} \sum_{k=0}^{1} \tilde{r}_{ikq} \frac{1}{P_{jk(X_q,\xi_i)}} \frac{\partial P_{jk}(X_q, \xi_i)}{\partial \xi_{il}},$$

where $i = 1, \ldots, m$ an $l = 1, 2, 3$, being m the number of items. $\tilde{r}_{ikq}$ is the expected number of answers in category $k = 0$ or 1 of item *i* of examinees with ability in the interval $(X_q - \Delta X_q/2, X_q + \Delta X_q/2)$, which, in turn, can be estimated by:

$$\tilde{r}_{ikq} = \sum_{j=1}^{N} x_{ijk} P(X_q, u_j, \xi) \Delta X_q.$$

Furthermore, $\tilde{N}_{ikq}$ is the expected number of examinees that answered item *i* with proficency in the interval $(X_q - \Delta X_q/2, X_q + \Delta X_q/2)$:

$$\tilde{N}_{ikq} = \sum_{k=0}^{1} \tilde{r}_{ikq}.$$

The probability of the ability to belong to this interval, given the answer vector u_j and the parameter vector ξ is given by:

$$P(X_q|u_j, \xi)\Delta X_q = \frac{P(u_j|X_q, \xi)A(X_q)}{\sum_{q=1}^{Q} P(u_j|X_q, \xi)A(X_q)}.$$

Finally, using the EM algorithm (Expectation-Maximization) one can obtain the solution of

the system $\frac{\partial logL}{\partial \xi_{il}} = 0$. The algorithm is executed in two steps:

- Step E: from the values $\xi_1^{'}, \ldots, \xi_m^{'}$, $\tilde{r}_{ikq}$ and $\tilde{N}_{ikq}$ are calculated;
- Step M: given $\tilde{r}_{ikq}$ e $\tilde{N}_{ikq}$, ξ that solves the system is found.

The cycle is repeated until the estimates of ξ are stable.

Normal distribution with mean 0 and variance 1 is assumed for the proficiency or ability parameter.

Inep uses marginal bayesian estimation, which, however, results in the same process described above.

In this method, a prior continuous distribution is assumed for the item parameters, given by $g(\xi)$. The bayesian estimator is the value of ξ that maximizes the density *a posteriori*

$$g(\xi|u_1, \ldots, u_N) = \frac{P(u_1, \ldots, u_N|\xi)g(\xi)}{P(u_1, \ldots, u_N)},$$

where $P(u_1, \ldots, u_N|\xi) = L$ is the marginal likelihood function of ξ.

Therefore, it is equivalent to maximizer $logL + log[g(\xi)]$, which can be done by the EM algorithm as described in the maximum likelihood method, only adding the prior distributions of the parameters

In the case of multiple groups, as in the Enem, one must incorporate the densities $g_k(\theta)$ of each one of the groups in analysis, estimating these densities' parameters together with the item parameters. For the reference group chosen, the parameters are fixed with mean equal to 0 and variance equal to 1. The marginal likelihood results in:

$$L = \prod_{k=1}^{K} \prod_{i=1}^{n_k} P(x_{ki}|\xi, \eta_k) = \prod_{k=1}^{K} \prod_{i=1}^{n_k} \int P(x_{ki}|\theta, \xi) g_k(\theta) d\theta,$$

and the EM algorithm can be used in this equation for the estimation of the item parameters.

Finally, for the estimation of examinees' abilities the method of the posterior expectation (EAP), considering the answer vector of each examinee and the respective item parameters previously estimated.

The EAP method basically consists in the use of a prior probability function to calculate the ability, that, for the examinees of the reference group, has mean 0 and variance 1. For the rest of the examinees, the formula below expresses the ability, which is rescaled to form their grades:

$$\hat{\theta} \approx \frac{\sum_{q=1}^{Q} X_q P(u_j|X_q,\xi)A(X_q)}{\sum_{q=1}^{Q} P(u_j|X_q,\xi)A(X_q)},$$

whose precision is measured by the posterior standard deviation (PSD):

$$PSD(\hat{\theta}_j) \approx \frac{\sum_{q=1}^{Q} (X_q - \hat{\theta}_j)^2 P(u_j|X_q,\xi)A(X_q)}{\sum_{q=1}^{Q} P(u_j|X_q,\xi)A(X_q)}$$

References

Amur, Zaira Hassan, Hooi, Yew Kwang, and Soomro, Gul Muhammad (2022). "Automatic Short Answer Grading (ASAG) using Attention-Based Deep Learning MODEL". In: *2022 International Conference on Digital Transformation and Intelligence (ICDI)*, pp. 1–7. DOI: 10.1109/ICDI57181.2022.10007187.

Ashley, K. (2017). *Artificial Intelligence and Legal Analytics,* Cambridge University Press.

Birnbaum, A. (1968). "Some Latent Trait Models and Their Use in Inferring an Examinee's Ability". In: *Lord, F.M. and Novick, M.R., Eds., Statistical Theories of Mental Test Scores, Addison-Wesley, Reading, 397-479*.

Bock and Aitkin (1981). "Marginal Maximum Likelihood Estimation of Item Parameters: Application of an EM Algorithm". In: *Psychometrika*.

Casella, George and Berger, Roger L (2010). "Inferência estatística". In: *São Paulo: Cengage Learning*.

Cheng S.; Liu, Q. (2019). "Enhancing Item Response Theory for Cognitive Diagnosis". In: DOI: http://xxx.lanl.gov/abs/1905.10957.

Gan W.; Sun, Y. (2020). "Knowledge Interaction Enhanced Knowledge Tracing for Learner Performance Prediction". In: *Seventh International Conference on Behavioural and Social Computing (BESC), Bournemouth, UK*.

Gao, Lina et al. (2022). "Deep cognitive diagnosis model for predicting studentsâ performance". In: *Future Generation Computer Systems*. DOI: https://doi.org/10.1016/j.future.2021.08.019..

Ghosh A.; Heffernan, N. and Lan, A.S. (2020). "Context-Aware Attentive Knowledge Tracing". In: *26th ACM SIGKDD International Conference on Knowledge Discovery Data Mining, Virtual Event, CA, USA*.

Goodfellow, Ian, Bengio, Yoshua, and Courville, Aaron (2016). *Deep Learning*. `http://www.deeplearningbook.org`. MIT Press.

Inep/MEC (2005). *Exame Nacional do Ensino Médio (Enem): fundamentação teórico-metodológica*. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. Brasília, DF, Brasil.

— (2021a). *Guia do participante*. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. Brasília, DF, Brasil. URL: `https://download.inep.gov.br/publicacoes/institucionais/avaliacoes_e_exames_da_educacao_basica/entenda_a_sua_nota_no_enem_guia_do_participante.pdf`.

— (2021b). *Microdados Enem 2021*. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira. Brasília, DF, Brasil. URL: `https://www.gov.br/inep/pt-br/acesso-a-informacao/dados-abertos/microdados/enem`.

Khan, Salman et al. (Feb. 2018). "A Guide to Convolutional Neural Networks for Computer Vision". In: vol. 8, pp. 1–207. DOI: `10.2200/S00822ED1V01Y201712COV015`.

Kumar, Liang and Ma (2019). *Verified Uncertainty Calibration*.

Linden, W.J. Van der (2016). *Handbook of Item Response Theory, Volume Two: Statistical Tools*. Chapman and Hall/CRC.

OpenAI (2022). *ChatGPT*. https://chat.openai.com. [Online; accessed 18-June-2023].

Park, J.Y. et al. (2022). "Comparing the prediction performance of item response theory and machine learning methods on item responses for educational assessments". In: DOI: `https://doi.org/10.3758/s13428-022-01910-8`.

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: `http://www.R-project.org/`.

Rizopoulos, D. (Nov. 2006). "ltm: an R package for Latent Variable Modelling and Item Response Theory Analyses". In: *Journal of Statistical Software* 17, pp. 1–25.

Rizzo, Maria L (2007). *Statistical computing with R*. Chapman and Hall/CRC.

Stroud, A. H. and Secrest, Don (1966). "Gaussian Quadrature Formulas". In: DOI: https://doi.org/10.1002/zamm.19670470216.

Travitzki, Rodrigo (2013). *ENEM: limites e possibilidades do Exame Nacional do Ensino Médio enquanto indicador de qualidade escolar*.

Tsutsumi E.; Kinoshita, R. and Ueno, M. (2021a). "Deep-IRT with independent student and item networks". In: *14th International Conference on Educational Data Mining, EDM, Paris, France*.

— (2021b). "Deep Item Response Theory as a Novel Test Theory Based on Deep Learning". In: DOI: https://doi.org/10.3390/electronics10091020.

Vaswani, Ashish et al. (2017). "Attention is all you need". In: *Advances in neural information processing systems*, pp. 5998–6008. URL: http://arxiv.org/abs/1706.03762.

Wang, Rui (2013). *The Chinese Imperial Examination System: An Annotated Bibliography*. Rowman Littlefield.

Wang, Zhifeng et al. (2023). "A Unified Interpretable Intelligent Learning Diagnosis Framework for Learning Performance Prediction in Intelligent Tutoring Systems". In: *International Journal of Intelligent Systems*. DOI: https://doi.org/10.1155/2023/4468025.

Yeung, C. (2019). "Deep-IRT: Make Deep Learning Based Knowledge Tracing Explainable Using Item Response Theory". In: *12th International Conference on Educational Data Mining, EDM, Montreal, QC, Canada*.

You, Hyesun (2022). "Bayesian Versus Frequentist Estimation for Item Response Theory Models of Interdisciplinary Science Assessment". In: *Interdisciplinary Journal of Environmental and Science Education*. DOI: https://doi.org/10.21601/ijese/12299.