Effectiveness Analysis of Oversampling Techniques By The Lens of Item Response Theory

Fabrício E. Corrêa[1], Lucas F. F. Cardoso[1,2], Vitor C A Santos[1,2], Regiane S. Kawasaki Francês[1], Ronnie C. O. Alves[2]

[1]Universidade Federal do Pará (UFPA)
Belém – PA – Brasil

[2]Instituto Tecnológico Vale (ITV)
Belém – PA – Brasil.

fabricio.correa@detran.pa.gov.br, lucas.cardoso@icen.ufpa.br

kawasaki@ufpa.br, (vitor.cirilo.santos, ronnie.alves)@itv.org

Abstract. *It is increasingly common for sectors of society to use Machine Learning (ML) techniques to make decisions and make variations with the data generated. One of the most common problems that a dataset can present is imbalance. Under these conditions, the tendency is to produce biased models, which favor the majority class. To mitigate this problem, data balancing algorithms can be used, one of which is oversampling. However, it is not a simple task to define whether an oversampling technique really helps in the model learning process. The experiments carried out show that IRT is capable of revealing the impact of oversampling even when there is no variation in performance when using classical characteristics. Furthermore, the results found pointed to the existence of an imbalance threshold where oversampling techniques are more effective.*

1. Introduction

The large amount of information generated and stored by different sectors of society increases every day, such as industry and academia. It is increasingly common for these different sectors to use Machine Learning (ML) techniques to make decision-making and predictions with the data they generate. ML is a sub-area of study of Artificial Intelligence and encompasses studies of computational methods for automating knowledge acquisition and for structuring and accessing existing knowledge [dos Santos 2005]. It is divided into three main types: supervised, unsupervised and reinforcement learning.

Supervised learning is trained using a set of labeled data so that it is later able to identify new data according to previously learned labels. These datasets can be obtained from specific platforms focused on ML such as OpenML [Vanschoren et al. 2014] and UCI [Dua et al. 2019].

One of the most common problems that a dataset can have is imbalance, which is defined as the greater incidence of one category in relation to the others within it. Under these conditions, the tendency is to produce classification models (or rules) that favor these majority classes. Such biased models can generate incorrect predictions. To generate more reliable models, the data set used must be balanced, and there are two main methods to perform balancing: one of them is undersampling, which is balancing

by “cutting” instances of the majority class. The second method is oversampling, which is the creation of synthetic data in the minority class. Both methods aim to equalize the number of instances of the classes. In this work the focus of study is oversampling.

Good performance in results returned by classic ML metrics, such as Accuracy and F1 score and MCC (Matthews Correlation Coefficient), but metrics like these can only reveal information about the quality of the model's final performance across the entire set of tests. Therefore, when applying oversampling techniques to balance the dataset, there is no way to identify whether the artificially generated instances are of good quality using only classic ML metrics.

Recent studies attempt to resolve this issue by applying techniques from other areas of knowledge in ML. For example, [Cardoso et al. 2022] uses IRT (Item Response Theory), used in psychometric tests to try to explain the model and quantify its reliability when analyzing the relationship between classifier and instance using IRT concepts.

Widely used in evaluating students in exams such as the ENEM (National High School Exam), the TRI does not only count the total number of correct answers in a test, as is done in classic evaluation metrics. The item is the basic unit of analysis and performance on a test can be explained by the ability of the person evaluated and the characteristics of the questions (items) [MEC 2012].

The TRI qualifies the item according to three parameters: Discrimination, which is the ability of an item to distinguish, in this case, students who have the required proficiency from those who do not; Degree of difficulty; Possibility of a random guess (guessing) [MEC 2012].

In this way, making a parallel with the use of IRT in Machine Learning, the items are the instances of the test set while the classifiers are the respondents who are performing the test.

In this work, IRT is used to evaluate the ability of classifiers to recognize data created synthetically by oversampling techniques, with the aim of trying to identify which technique can generate higher quality data and under what conditions this is possible. The remainder of this work is organized into the following sections: Section 2 presents the theoretical foundation on Machine Learning, data imbalance and IRT; Section 3 presents the methodology used to evaluate oversampling techniques with IRT; Section 4 presents the results obtained; Section 5 brings the final considerations of the work.

2. Theoretical Background

2.1. Machine Learning (ML)

As a subarea of Artificial Intelligence, Machine Learning is the science of programming computers so that they can learn from data [Barchilon and Escovedo 2021], encompasses the studies of computational methods for automating knowledge acquisition and for structuring and access to existing knowledge [dos Santos 2005].

The use of ML can contribute to organizations, with the purpose of making efficient prediction. It has three main types: supervised learning, unsupervised learning, and reinforcement learning. Supervised learning is a machine learning paradigm that aims to acquire relationship information between the input and output of a system, based on a

set of training samples [Monard and Baranauskas 2003]. The model is trained using a set of labeled data so that it is later capable of identifying new data according to previously learned labels.

In a binary dataset, imbalance is defined as the lower incidence of one category (minority class) compared to the other category (majority class). This means that we have a lot of information from the majority class. Therefore, the tendency is to produce classification models (or rules) that favor this class, resulting in a low recognition rate for the minority class [Castro and Braga 2011], which could result in problems for the model. One of the problems in model evaluation is the accuracy paradox, which can be defined as the contradictory situation in which a high accuracy in your classification model can highlight a failure of your own model to make truly significant predictions.

One way to try to mitigate the bias caused by the difference between classes is to manipulate the data, increasing or removing, so that the classes are balanced. This can be done using oversampling and undersampling techniques. In this work the focus is on oversampling balancing techniques.

Oversampling balancing techniques aim to increase the number of instances of the minority class by generating synthetic data, leaving them with the number of instances equal to that of the majority class. There are different oversampling algorithms, for this work 5 techniques used by the ML community were chosen, they are: SMOTE, ADASYN, SVMSMOTE, SMOTEN and BorderlineSMOTE.

- SMOTE: Generates virtual training records by linear interpolation for the minority class. These synthetic training records are generated by randomly selecting one or more of the k-nearest neighbors for each example in the minority class [Chawla et al. 2002]. To do this, the difference between the feature vector (sample) under consideration and each of the selected neighbors is taken. This difference is multiplied by a random number drawn between 0 and 1 and then added to the previous feature vector. This causes the selection of a random point along the "line segment" between the features. In the case of nominal attributes, one of the two values is randomly selected [Fernández et al. 2018].
- ADASYN: The main idea of the algorithm is to use the density distribution as a criterion to automatically decide the number of synthetic data that needs to be generated for each example of the minority class [He et al. 2008]. It creates different number of synthetic data based on data distribution. The algorithm ADASYN can decide the number of synthetic examples that need to be generated for each minority example by the number of its closest majority neighbor, i.e., the closer majority neighbor, the more synthetic examples will be created. An important disadvantage of this approach is that synthetic data is only generated close to the boundary [Majumder et al. 2020]. .
- SVMSMOTE: Generates synthetic samples close to the optimal decision region. It focuses on generating samples close to the edge region of the class, using only samples that make up the optimal decision region, and it is not necessary to use all samples from the minority class [Nguyen et al. 2011]. The optimal region is obtained by training the SVMs on the original database. SVMSMOTE generates synthetic data through interpolation and extrapolation. The first is carried out if the number of samples belonging to the class majority is greater than or equal

to half of the total number of neighbors. In this way, the edge of the minority class will be consolidated. The second occurs when the number of samples belonging to the majority class is less than half the total number of neighbors. Thus goes the extrapolation and consequent expansion in the area of the minority class [LIMA et al. 2020].

- SMOTEN: It works in such a way that the closest neighbors of the majority class are estimated and excluded, which excludes the most extensive data before oversampling the minority class [Alabrah 2023]. It expects that the data to resample are only made of categorical features[Lemaître et al. 2017].
- Borderline SMOTE: It is a variant of the SMOTE technique that focuses on generating synthetic instances of the response variable in regions closer to the decision boundary between the minority and majority classes [Han et al. 2005].

2.2. Item Response Theory (IRT)

Widely used in evaluating students in tests such as the ENEM (National High School Exam), the IRT emerged as a way of considering each item of an object studied, without revealing only the total score. Therefore, in the case of tests or questionnaires, the conclusions depend on each item that composes them [Araujo et al. 2009]. That is, IRT does not only account for the total number of correct answers in a test, the item is the basic unit of analysis. Performance on a test can be explained by the ability of the respondent and the characteristics of the questions (items) [MEC 2012].

These characteristics make it possible to estimate the ability of an evaluated candidate and to ensure that these skills, measured from one set of items, are compared with another set on the same scale, even if they are not the same and there are different amounts of items used for the calculation [MEC 2012].

Adopting the dichotomous model or cumulative multiple-choice items (those that are corrected as right or wrong), we have the logistic models of 1, 2 and 3 parameters. The logistic model adopted in this work is the 3-parameter model, which is given by the Equation 1.

$$P(U_{ij} = 1|\theta_j) = c_i + (1 - c_i)\frac{1}{1 + e^{-a_i(\theta_j - b_i)}} \tag{1}$$

Where, U_{ij} is a dichotomous variable that takes on the value 1, when the respondent j answers correctly, agrees or satisfies the conditions of item i, or 0 otherwise; θ_j represents the latent trait (ability) of the respondent j; b is the difficulty parameter of the item i, measured on the same scale as the latent trait, refers to the level of skill needed to answer the item correctly, the higher the value of b, the greater the difficulty of the item; a is the discrimination (or slope) parameter of the item i, with a value proportional to the slope of the characteristic curve of the item at the point b. It refers to the item's ability to discriminate between individuals with different abilities. Items with higher a values provide better breakdowns; c is the parameter of chance hit or guesswork. The higher the value of c, the greater the chance of the item being answered correctly by chance.

In the interpretation of the 3-parameter logistic model, the $P(U_{ij} = 1|\theta_j)$ is the probability of getting it right ($U_{ij} = 1$) given the parameters of the item i and the estimated ability of the respondent j [Araujo et al. 2009]. The relationship between these

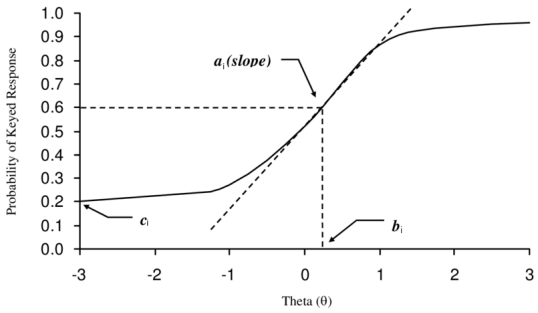

Figure 1. ICC for 3-parameter logistic model.

mathematical functions is estimated and is called Item Characteristic Curves (ICC), as can be seen in Figure 1.

It is noted that the discrimination parameter a is very determining for the behavior of the ICC. Normally this parameter has positive values, but it is not impossible for items with negative discrimination values to appear. Such values are not expected by the IRT, and when they appear they usually indicate that there is some inconsistency with the item, as in these cases the ICC curve inverts so that respondents with less ability have a greater chance of getting it right than respondents with greater ability [Cardoso et al. 2022].

In addition to evaluating the students themselves, IRT is used in ENEM to evaluate the quality of student learning. Likewise, this work aims to use IRT to evaluate the stage that the model is studying for testing, i.e., the training stage in which there will be artificial instances created by oversampling techniques and at the end the IRT results are observed to evaluate whether the synthetic data generated by the different oversampling techniques actually improved model learning.

3. Methodology

To carry out the experiments, the CONDA 23.3.1 programs were used, with the Python 3.10.4 language, and the Jupyter notebook. Five oversampling balancing techniques were selected from the Imbalanced-learn library: ADASYN, SMOTE, SMOTEN, SVMSMOTE and BorderlineSMOTE.

When carrying out the proposed study, the analyzed dataset has its split stratified and is divided into training and testing, in a 70/30 proportion. The training dataset goes through the oversampling process, which is done using several techniques separately, one at a time, where the models will be generated. The result is a response matrix that contains the result of each model's prediction for each test instance, 1 for when the model gets the prediction right and 0 for when it gets it wrong. From the response matrix, the IRT item parameters are estimated, as well as the classic evaluation metrics Accuracy, F1 score and MCC.

To carry out this process automatically, decodIRT[Cardoso et al. 2020] was used, which is a tool composed of a set of Python scripts, which automates the model generation process and estimates item parameters, with their results. it is possible to calculate classical metrics and ICC. By definition, the tool uses different families of algorithms to generate the results contained in the response matrix in order to generate a diversity of

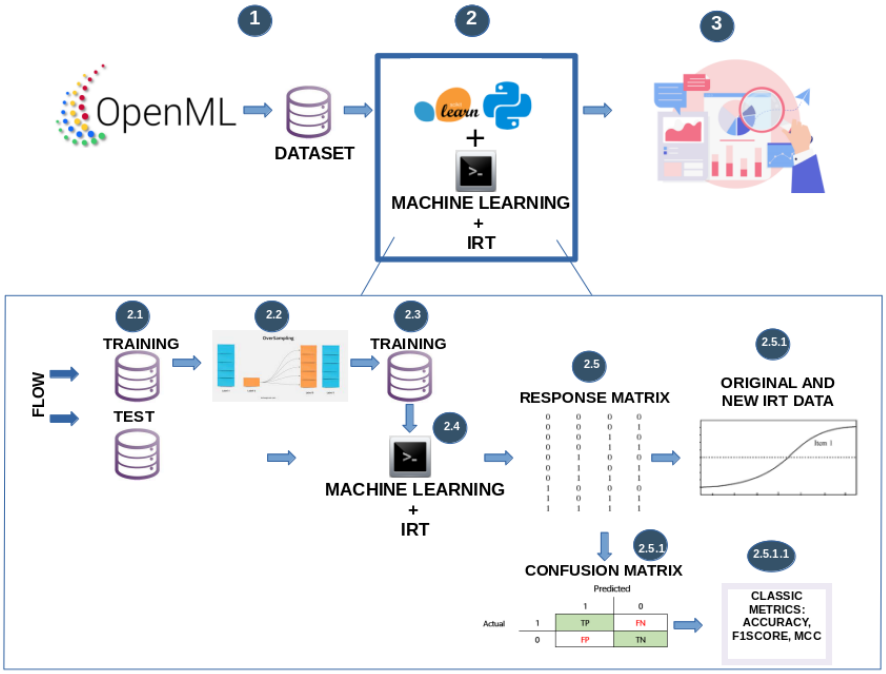

Figure 2. Methodology flowchart.

responses: Naive Bayes, MLP, KNN, SVM, Decision Trees and Random Forests. The step-by-step process is described below and is also illustrated in Figure 2:

1. Obtaining the datasets: the pre-selected datasets were acquired from the OpenML platform, through the platform's API made available for Python;

2. Dataset processing:

   2.1 Division of data sets: then decodIRT divides the dataset into training and test, where the training dataset is subjected to data processing;

   2.2 Data processing: datasets are balanced using oversampling techniques;

   2.3 Training dataset processing: then the training dataset is processed so that results are returned, both for the original training dataset and for the training dataset that was balanced by oversampling techniques;

   2.4 Machine Learning + IRT: the decodIRT tool processes training datasets with and without data processing, that is, with and without balancing;

   2.5 Response Matrix: the result of the processing is the response matrix, from which the IRT item parameters (2.5.1) are calculated and a confusion matrix (2.5.1) is formed, from which the classic metrics (Accuracy, F1score and MCC) were generated, all from both the original dataset and the balanced dataset.

3. Results: the end of the process is the result of the classic metrics and item parameters, both from the original dataset, with unbalance, and from the datasets that were subjected to balancing through oversampling techniques. The results obtained are then organized and analyzed.

Table 1. Metadata of the selected datasets.

Level	Dataset	Major %	Minor %	Nº Feat.	Nº Inst.	Dimen.	C. Entropy	Auto Corr.	Feat. N.	Feat. C.
1	Credit apvl	55,51	44,49	16	690	0,023	0,991	0,978	6	10
	Cylinder	57,77	42,23	40	540	0,074	0,982	0,811	18	22
	dresses-sales	57,99	42,01	13	500	0,026	0,981	0,4729	1	12
2	wdbc	62,74	37,26	31	569	0,054	0,952	0,625	30	1
	tic-tac-toe	65,34	34,65	10	958	0,010	0,931	0,999	0	10
	qsarbiodeg	66,00	34,00	42	1055	0,040	0,922	0,997	41	1
3	credit-g	70,00	30,00	21	1000	0,021	0,0881	0,570	7	14
	ilpd	71,35	28,65	11	583	0,019	0,864	0,613	9	2
	haberman	73,529	26,47	4	306	0,130	0,833	0,786	2	2
4	jm1	80,66	19,34	22	10885	0,002	0,709	1,000	21	1
	backache	86,11	13,89	32	180	0,177	0,581	0,731	5	27
	pc4	87,79	12,21	38	1458	0,026	0,535	0,885	37	1
5	sim. crashes	91,48	8,52	21	540	0,039	0,420	0,839	20	1
	pc1	93,06	6,94	22	1109	0,198	0,363	0,990	21	1
	oil_spill	95,63	4,37	50	937	0,053	0,259	0,928	49	1

3.1. Dataset Selection

Since our goal is to balance data by oversampling techniques, it is necessary to select datasets with different levels of imbalances. To this end, the rule of gradual growth of these levels was defined:

- Level 1: Imbalance greater than 50% and less than 60%;
- Level 2: Imbalance greater than or equal to 60% and less than 70%;
- Level 3: Imbalance greater than or equal to 70% and less than 80%;
- Level 4: Imbalance greater than or equal to 80% and less than 90%;
- Level 5: Imbalance greater than or equal to 90% and less than 100%.

As a case study, fifteen datasets of the OpenML [Vanschoren et al. 2014] were selected, through the platform's API made available for Python. There are 3 datasets for each level of unbalance, starting from 55.5% of *credit-approval*, considered here as the dataset with the lowest degree of unbalance, going up to the *oil spill* dataset, whose majority class has 95.6%, considered as here, as the dataset with the highest degree of unbalance. In addition, other metadata is provided, which can be seen in the Table 1.

4. Results and Discussion

This section presents the results obtained from fifteen data sets organized from the lowest to the highest level of imbalance, starting from level 1 to level 5. Firstly, the results are analyzed using only classic ML metrics and subsequently the Analysis is performed through IRT.

4.1. Through the lens of classical metrics

For each dataset of each level of imbalance, the 5 Oversampling techniques studied were applied to subsequently calculate the Accuracy, F1 and MCC metrics. In this way, 5 values of each evaluation metric will be obtained, one for each Oversampling technique. The average variation of the values was then plotted in the graph in Figure 3, which presents an overview of the behavior of classical metrics as the level of imbalance in the datasets increases.

It is noted that there is little or almost no impact from the application of oversampling techniques on the performance of the models for level 1 and 2 datasets. This was expected, because for these datasets that already have close numbers of instances per class, oversampling techniques need to generate few instances to balance the datasets and, therefore, will have little impact. The interesting results are for datasets from level 3 onwards. It is noted that there is a gradual drop in accuracy as the imbalance of the datasets

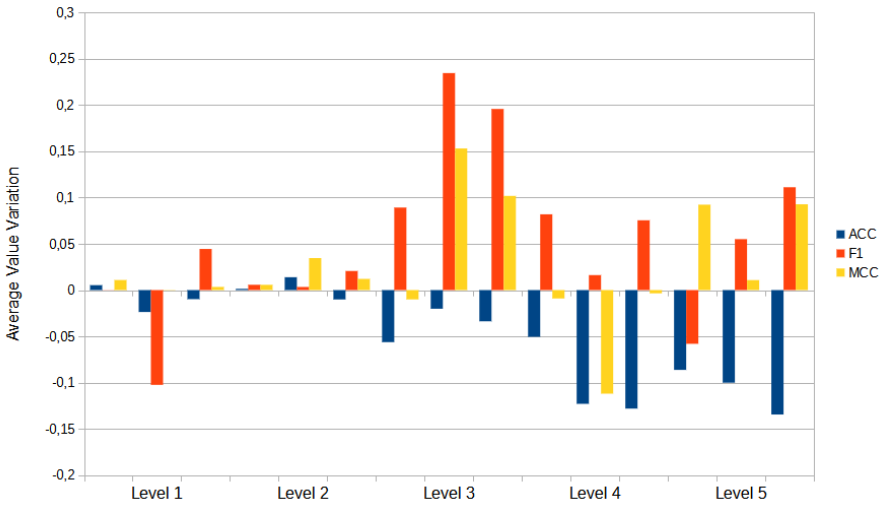

Figure 3. Variation in the average value of classic metrics after oversampling.

increases, this indicates that the application of oversampling techniques may not have the desired effect on very unbalanced datasets.

Only level 3 datasets present the most interesting model performance results after applying oversampling. It is noted that the average drop in accuracy was minimal and showed more significant increases for F1 and MCC, with the *ilpd* (71.35% imbalance) and *haberman* (73% imbalance) datasets that had the greatest improvement in performance. *ilpd* had an average F1 increase from 0.3 (original) to 0.53 (oversampling) and the MCC increased from 0.18 to 0.34, *haberman* in turn, had an average F1 increase of 0.13 to 0.32, while the MCC increased from 0.01 to 0.11. For the level 3 datasets, it was the SMOTE and ADASYN techniques that showed a greater performance improvement than the others, despite it being by little difference.

These results indicate that there may be imbalance thresholds, at which oversampling techniques may or may not have a positive effect. In the experiments carried out, level 3 of unbalance is the condition that is most worth the effort of applying an oversampling technique, while for levels 1 and 2 the unbalance is too small for the techniques to have any impact. For example, for the level 2 *wdbc* dataset (62.74% imbalance), the difference between the performance metrics values was very low, being 0.001, 0.005 and 0.005 for Accuracy, F1 and MCC, respectively.

For levels 4 and 5, the results suggest that the imbalance is too high to apply oversampling, so that the performance of the models may even worsen. For example, the dataset *climate-model-simulation-crashes* (91.48% imbalance) at level 5 presented variations very close to zero in model performance, with -0.08 for Accuracy, -0.05 for F1 and 0.09 for the MCC. It is noted in this case that there was no advantage in applying oversampling techniques.

4.2. Through the lens of IRT

Based on the results obtained by the classical metrics, there was no improvement in the final performance of the models for the *wdbc* dataset, however when subjected to IRT it is possible to notice some positive changes after oversampling. Figure 4 shows the

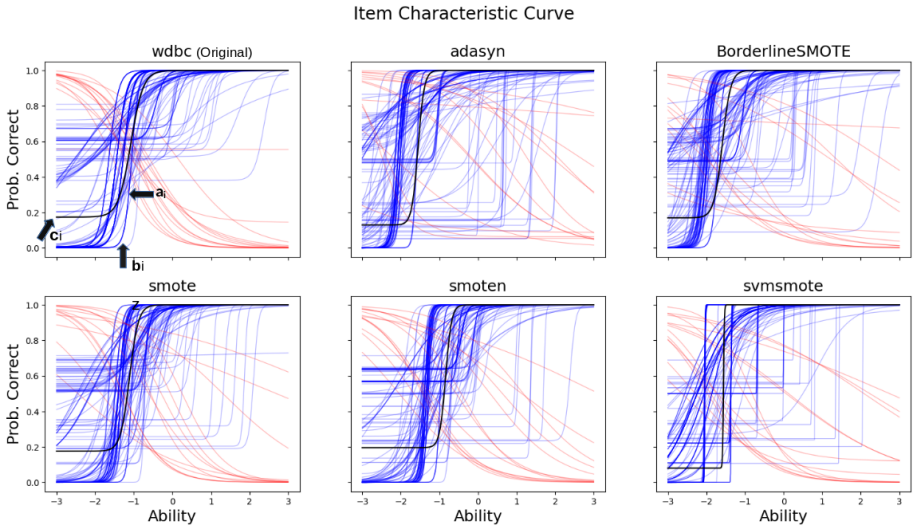

Figure 4. CCI of the techniques applied to the wdbc dataset.

behavior of the instances with the ICCs calculated with the original dataset, i.e., without applying balancing techniques and for each oversampling technique that was applied. The blue lines represent instances with positive discrimination, while the red lines represent instances with negative discrimination. In section 2.2, it was explained that negative discrimination means that the item has some inconsistency that makes it more difficult for high-ability respondents than low-ability respondents.

Even in the original condition, the *wdbc* dataset already presents some instances with negative discrimination, although it is clear that there are many more positive instances (blue lines). Even though there is no improvement in performance with classical metrics, it can be seen that after applying oversampling techniques, the total number of instances with negative discrimination decreases and has a less pronounced slope, with ADASYN and BorderlineSMOTE being the algorithms that present the best performance for this case. For ML, a poorly observed class distribution during the training stage can result in instances with negative discrimination in the test. Therefore, ADASYN and BorderlineSMOTE may be presenting the best performance as they are techniques focused on generating synthetic data based on lower density and edge instances.

When returning to the analogy of the student taking a test, although the number of total correct answers does not increase, and even so negative discrimination decreases, it means that students have greater conviction in their answers and better master the content of the questions they get right. Therefore, it can be said that in this case, because of oversampling, the models have greater confidence in the instances that classify correctly.

In contrast to the result presented previously, *climate-model-simulation-crashes* represents the worst case scenario. As seen previously, the application of oversampling techniques did not result in an improvement in the model's final performance and may even be harming learning. This becomes clearer when observing the ICC curves, as can be seen in Figure 5.

It is noted that the original dataset already presented a large number of instances with negative discrimination, in addition to the high difficulty, due to the positioning of

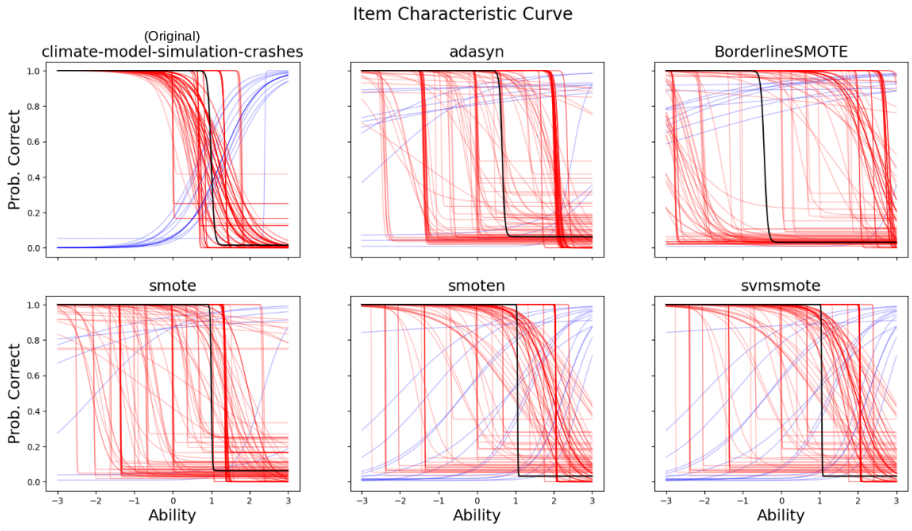

Figure 5. CCI of the techniques applied to the climate-model-simulation-crashes dataset.

the lines to the right of the graphs. After applying each technique, no improvement can be noticed in the general picture, but it can be seen that the total number of good instances (blue lines) decreases and loses slope, which indicates that the discriminative value of these instances is very close to zero.

These results highlight the observations made previously. Depending on the degree of imbalance and the characteristics of the dataset, oversampling techniques may not be of any use. In these cases, the best solution would be to invest in obtaining more data from the minority class.

However, this is not a definitive rule. Among the level 5 datasets, *pc1* (93.56% imbalance) presented very interesting results. As can be seen in Figure 6, all the oversampling techniques that were applied resulted in an improvement in the item parameters of the dataset instances, where the average discrimination increased from -8.79 to 7.93, the difficulty of the dataset decreased from 1.35 to -1.05 and guessing decreased from 0.17 to 0.05.

These results indicate that data that were previously considered bad by IRT, after oversampling techniques, are considered good for evaluation. With the original accuracy (92%) of the models being very close to the percentage of the majority class (93.56%), this could mean that the models were biased towards classifying all instances as being from the majority class without actually having adequately generalized the class distribution. This is highlighted with the guess value decreasing from 0.17 to 0.05 after applying oversampling. But what causes this superior result for *pc1* than *climate-model-simulation-crashes*, both of which have very similar levels of imbalance. When analyzing the metadata in Table 1, it is noted that *pc1* has twice as many instances as *climate-model-simulation-crashes*, the largest number of examples available to the algorithms Oversampling may be the most direct explanation. A deeper investigation into the two datasets would be interesting in future work.

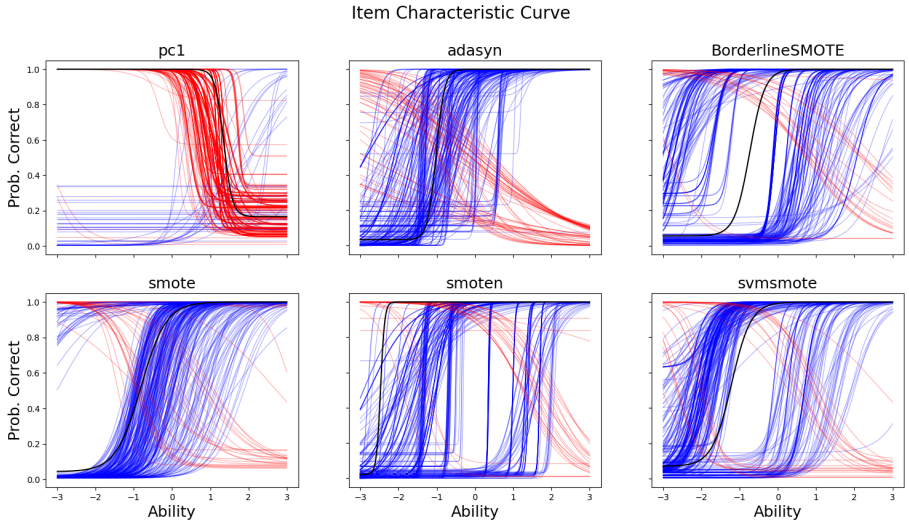

Figure 6. CCI of the techniques applied to the pc1 dataset.

5. Final Considerations

In this article, IRT was explored as a tool to measure the effectiveness of data balancing techniques. For this, 15 separate datasets were evaluated at 5 different levels of imbalance. It was verified, throughout the 5 levels, that there may be unbalance thresholds, in which oversampling techniques may not have a positive effect. Therefore, depending on the degree of imbalance and the characteristics of the dataset, oversampling techniques may not be of any use when analyzing whether there was a performance gain using classical metrics. Although classical metrics indicate that there is no gain, the results obtained by IRT can indicate when a model has gained confidence with the application of oversampling techniques. This can be observed from the relationship between the item parameters and the dataset, revealed by the Item Characteristic Curves of each evaluated instance. For example, for the *wdbc* dataset with level 2 imbalance, the CCI revealed that the number of instances considered inappropriate by the IRT decreased after applying oversampling.

References

Alabrah, A. (2023). An improved ccf detector to handle the problem of class imbalance with outlier normalization using iqr method. *Sensors*, 23(9):4406.

Araujo, E. A. C. d., Andrade, D. F. d., and Bortolotti, S. L. V. (2009). Teoria da resposta ao item. *Revista da Escola de Enfermagem da USP*, 43:1000–1008.

Barchilon, N. and Escovedo, T. (2021). Machine learning applied to the inss benefit request. In *XVII Brazilian Symposium on Information Systems*, pages 1–8.

Cardoso, L. F., de S. Ribeiro, J., Santos, V. C. A., Silva, R. L., Mota, M. P., Prudêncio, R. B., and Alves, R. C. (2022). Explanation-by-example based on item response theory. In *Brazilian Conference on Intelligent Systems*, pages 283–297. Springer.

Cardoso, L. F., Santos, V. C., Francês, R. S. K., Prudêncio, R. B., and Alves, R. C. (2020). Decoding machine learning benchmarks. In *Brazilian Conference on Intelligent Systems*, pages 412–425. Springer.

Castro, C. L. d. and Braga, A. P. (2011). Aprendizado supervisionado com conjuntos de dados desbalanceados. *Sba: Controle & Automação Sociedade Brasileira de Automatica*, 22:441–466.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

dos Santos, C. N. (2005). *Aprendizado de máquina na identificação de sintagmas nominais: o caso do português brasileiro*. PhD thesis, Instituto Militar de Engenharia.

Dua, D., Graff, C., et al. (2019). Uci machine learning repository, 2017. *URL http://archive. ics. uci. edu/ml*, 7(1).

Fernández, A., Garcia, S., Herrera, F., and Chawla, N. V. (2018). Smote for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *Journal of artificial intelligence research*, 61:863–905.

Han, H., Wang, W.-Y., and Mao, B.-H. (2005). Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pages 878–887. Springer.

He, H., Bai, Y., Garcia, E. A., and Li, S. (2008). Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, pages 1322–1328. Ieee.

Lemaître, G., Nogueira, F., and Aridas, C. K. (2017). Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, 18(17):1–5.

LIMA, J. L. P. et al. (2020). Adversarial oversampling: um método para balanceamento baseado em redes neurais adversárias. Master's thesis, Universidade Federal de Pernambuco.

Majumder, A., Dutta, S., Kumar, S., and Behera, L. (2020). A method for handling multiclass imbalanced data by geometry based information sampling and class prioritized synthetic data generation (gicaps). *arXiv preprint arXiv:2010.05155*.

MEC (2012). Teoria de resposta ao item avalia habilidade e minimiza o "chute" de candidatos. *http://portal.mec.gov.br/ultimas-noticias/389-ensino-medio-2092297298/17319-teoria-de-resposta-ao-item-avalia-habilidade-e-minimiza-o-chute*.

Monard, M. C. and Baranauskas, J. A. (2003). Conceitos sobre aprendizado de máquina. *Sistemas inteligentes-Fundamentos e aplicações*, 1(1):32.

Nguyen, H. M., Cooper, E. W., and Kamei, K. (2011). Borderline over-sampling for imbalanced data classification. *International Journal of Knowledge Engineering and Soft Data Paradigms*, 3(1):4–21.

Vanschoren, J., Van Rijn, J. N., Bischl, B., and Torgo, L. (2014). Openml: networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, 15(2):49–60.