Understanding when SMOTE works

Eduardo Silva up201603135@fe.up.pt
Francisco Ferreira up201605660@fe.up.pt
Joana Silva up201208979@fe.up.pt
Joana Ramos up201605017@fe.up.pt
João Fidalgo up201605237@fe.up.pt

***Abstract*—Oversampling techniques such as SMOTE are commonly used to treat imbalanced datasets. However, it is still not very clear in which conditions does SMOTE actually improve the performance of the classifiers. The main purpose of this work is to understand when this technique is useful, and, to achieve this, a total of 53 imbalanced datasets and its meta-features were analysed, both before and after applying SMOTE, with 35 variants of the algorithm also being tested. In the end, it was possible to observe that SMOTE wasn't very helpful in less complex datasets, not being able to improve the original AUC in ∼64% of the datasets.**

I. Introduction

Class Imbalanced Data (ID) is one of the most common problems in Machine Learning, deeply affecting the data quality. It occurs when there is a substantial difference in class distribution, being the example distribution among the classes skewed and biased. In binary classification problems it is noticeable when the have a class that contains the **majority of the samples** and another the **minority**, being the latter underrepresented in contrast to the class containing the majority of the examples. This is a problem as some classifiers can be biased towards the most represented class (majority class), which results in the classifier not being able to perform a good generalization, thus affecting the quality of the results due to the misclassification in the minority class.

Throughout the years, several techniques have been researched and developed to fix this issue, namely three different approaches: **data-level**, **algorithm-level** and **hybrid-level** [1]. Data-level approaches are the main applied techniques, as they have been proven to be considerably effective in handling the class imbalance problem. These approaches can be divided into two categories, **oversampling** and **undersampling**, being oversampling comprised of generating new samples by replicating some instances of the minority class, whereas undersampling consists of removing samples from the majority class. Algorithm-level techniques can modify existing algorithms or infer certain algorithms to handle the imbalanced data. The third approach, hybrid-level, combines the data-level and algorithm-level, where methods such as ensemble and cost-sensitive learning are used.

This work focuses on the data-level approaches, more specifically in the **SMOTE (Synthetic Minority Oversampling Technique)** [2] oversampling technique. In SMOTE, samples that are close in the feature space are chosen, being drawn a line between the samples in the feature space and a new example at a point within that line. More precisely, a instance from the minority class instance is first selected and its k nearest neighbors are found. Then, a new instance is created by choosing a certain neighbor randomly and connecting it to the first selected minority class instance, forming a line segment and later generating the synthetic instances by making a convex combination of the two previous instances.

The **main objective** of this work is to understand **how the SMOTE technique works and when is indeed useful to use**. To achieve this, an analysis of the classification of multiple imbalanced datasets was performed by observing the variation of meta-features before and after SMOTE, so as to verify if using SMOTE is valuable for each dataset. **Meta-features**, which are also referred to as characterization measures, are one of the most detailed ways of characterizing the complexity of a dataset. In order to comprehend how using SMOTE helps the imbalance problem, various meta-features were extracted for each dataset, as it will be further detailed in following sections.

II. Related Work

In this section, we present a summary and overview of research articles that are related with the theme of this project: studies that deal with strategies to fix the imbalance problem, using SMOTE and other sampling techniques, as well as some using meta-features.

The first relevant study is by Costa et al. [3], that in 2020 analysed the behavior of oversampling imbalance strategies with a meta-learning approach. By using 163 real-world imbalanced binary datasets, the research was divided into three main phases: partitioning and resampling of the datasets, extraction of the meta-features and performance evaluation, and Exceptional Preferences Mining (EPM) [4]. To be able to obtain interesting results, in total 9 imbalance handling techniques were used, such as SMOTE [2], SafeLevel-SMOTE [5], Borderline-SMOTE [6], ADASYN (adaptive synthetic sampling method for imbalanced data) [7], AHC (Agglomerative Hierarchical Clustering) [8], ADOMS (Adjusting the Direction of the synthetic Minority clasS examples) [9], ROS (Random Over-Sampling), SMOTE-TL [10], and SMOTE-ENN [11].

The authors were able to conclude that in datasets where the classification tasks are more simple, preprocessing may not be needed, being better to use the original dataset. With this, they found that the application of preprocessing methods is better when the overall complexity of the dataset is high, as well as the number of borderline instances. Regarding some of the imbalance algorithms, it was possible to observe the following:

- AHC - Works better in less complex problems that present a reduced dimensionality;
- ADOMS - Better for problems where subgroup elements have low variance and a small first principal component of the covariance matrix;
- SMOTE-TL - Good for complex datasets with high dimensionality;
- ROS - Works better when the entropy of the attributes is high.

In a article from 2018, Santos et al. [12] conducted a research where the goal was to observe the implementation of Cross-Validation (CV) in imbalanced datasets, comparing the differences between not oversampled and oversampled approaches, distinguishing overfitting and overoptimistic methods in imbalanced data, and determining what are the most relevant oversampling approaches. To achieve this, the studied was divided into 3 approaches:

- Baseline - The datasets were divided into 5 stratified folds and the classifiers applied without any oversampling techniques;
- Approach 1 - Oversampling techniques are applied to the datasets, being CV and performance evaluation done afterwards, as well as the retrieving of data complexity measures;
- Approach 2 - Oversampling techniques are applied to the training partitions during the Cross-Validation phase, being the datasets first divided into 5 folds. Data complexity measures are then observed for the oversampled training sets.

Concerning the oversampling algorithms used, some of the techniques used in the first article such as ROS, SMOTE, ADASYN, Borderline-SMOTE, SafeLevel-SMOTE, SMOTE-ENN, ADOMS, and AHC were used in this paper, along with other algorithms like CBO (Cluster-Based Oversampling) [13], MWMOTE (Majority Weighted Minority Oversampling Technique) [14] and SPIDER (Selective Pre-Processing of Imbalanced Data) [15].

After a careful analysis using known performance metrics such as AUC, F-1, G-mean, and SENS, there were multiple findings. Overoptimistic results were found for Approach 1, making it not very suitable to handle imbalanced problems. This showed that overoptimism was related to the complexity of the prediction task and not to the imbalance ratio and data's sample size. In what respects the oversampling techniques that were applied to the datasets, MWMOTE and SMOTE-TL provided the best results, having high AUC scores (in average 0.871). Furthermore, the authors also concluded that to have the best classification results, the oversampling methods need to have 3 characteristics: adaptive weight of the minority examples, cluster-based synthetization of examples, and use cleaning procedures.

Zhang et al. [16] in 2019 presented a study which analysed 80 public imbalanced datasets. The main purpose of the research was to present an Instance-Based Learning (IBL) recommendation algorithm that is capable of choosing the most suitable imbalance technique to be used in each case. To achieve this, the authors first built the meta-knowledge database, where all of the extracted meta-features and meta-targets would be stored. Regarding the meta-features extraction, various measures were described:

- Traditional measures - Relative to normal measures such as the number of missing values, number of features, samples, class labels, analysis of variance and correlation, among others...;
- Complexity measures - Measures that are focused on the complexity of the class boundaries, such as separation, cross-overlapping, etc.;
- Landmarking measures - Measures that with disagreements found within classification algorithms, describe the nature of a dataset;
- Model-based measures - Measures that for instance use decision tree-based characterization for meta-learning;
- Structural and statistical information based measures - As the name suggests, measures that name structural and statistical characteristics of the dataset, such as the minimum, maximum, the seven octiles, among others.

After the meta-features being extracted, the meta-objects were recognized using the runtime and AUC to rank each dataset, followed by the recommendation procedure itself, where it was applied the Max-min normalization, the k-nearest neighbors were found and all of the imbalance handling methods were ranked. The imbalance algorithms used were not restricted to oversampling methods but also contained undersampling techniques and other algorithm-level approaches, being implemented multiple algorithms: RUS (Random Under-Sampling), ROS, SMOTE, MetaCost (Cost sensitive analysis), Bagging, AdaBoost, SMOTEBoost (combination of SMOTE and Boosting), EasyEnsemble (RUS + AdaBoost), UnderBagging (combination of Bagging and RUS), EM1vs1 (encode based ensemble learning), and EDBC (dissimilarity-based imbalance algorithm).

Overall, the recommendation algorithm provided very good results, being the hit rate of the top 3 algorithms for any problem higher than 96%, which shows that the instance-based algorithm can provide effective results and that it is indeed possible to choose the best imbalance handling techniques to use.

The survey [17] describes and summarizes various articles that handle imbalance problems.

The first article to be described is by Fernandez et al. [18], where the goal was to understand the inner structure of some of the existing methodologies that handle the imbalance problems in the context of Big Data. The study used the Spark library to implement ROS and RUS variations within the MapReduce model, along with the SMOTE-BigData implementation that is provided in Hadoop, using afterwards the Decision Tree and Random Forest algorithms to classify the models. It was clear that the use of ROS and RUS presented better results than SMOTE-BigData in all of the experiments done, and that

when comparing ROS and RUS, ROS provided better quality results.

In 2015, Rio et al. [19] performed a study that analysed imbalanced data with around 32 milion examples and 631 features from the ECBDL'14 Big Data Competition. Like the previous described article, this paper used the ROS and RUS implementations under MapReduce, along with Random Forest for classification. Similarly to the results obtained by Fernandez et al. [18], ROS proved to be better than RUS for solving imbalance problems, as RUS suffered from the small sample size of the underrepresented data.

Triguero et al. [20], in 2016, tried to answer the problem of having a reduced density in the minority class of extremely imbalanced datasets using Apache Spark and Apache Hadoop. It was revealed that Apache Spark was more efficient than Hadoop, providing a faster runtime. Regarding the imbalanced techniques, the study compared RUS with EUS (Evolutionary Under-Sampling) and found out that while RUS provided a faster runtime, EUS was the best classifier among the two.

The two articles described next are related to a real-world scenario aiming to predict traffic accidents in advance. As the size of the data relating to traffic accidents is very large and the class distribution can be imbalanced, Park and Ha [21] and Park et al. [22] developed studies using the Apache Hadoop framework to determine the best balance between classification accuracy and oversampling (SMOTE). The first study, obtained good results, showing that the optimum balance was around 30% of the original dataset. The latter, showed that the greatest increase of classification accuracy occurred when the minority classes had an increase of 25%.

Finally, the last article described in the survey was by Chai et al. [23]. The study focused on finding the feasibility of using classification to automatically identify health information technology (HIT) incidents, using undersampling techniques and logistic regression as the classification algorithm. The results revealed that the use of undersampling where the majority classes were reduced in 50% of the original dataset indicated that classification performance was unaffected.

The last article is about the 85 SMOTE variations developed by Kovács [24] that are available on GitHub. The study [25] evaluated the performance of all the 85 variants in a total of 104 imbalanced datasets, with the purpose of understanding what were the oversampling methods that provided better results under general situations and, thus, obtaining a new baseline model. Kovács discovered that oversampling is as expected an interesting technique to improve the performance of imbalanced datasets. Regarding specific variants, polynom-fit-SMOTE and SMOTE-IPF presented high quality results, specially for data with changing or unknown characteristics. Moreover, it was also visible that simpler techniques were more efficient than more mathematically complex variants, namely in extremely imbalanced datasets.

However, in the end, it was noticeable that no variation could provide outstanding results in each of the 104 datasets used, as they have different characteristics and oversamplers can have many hyperparameters, which makes it much more complicated to design a variation that is able to fit all of the needs of every dataset.

After the analysis of these studies, it is clear that there are many different types of imbalance handling strategies, as can be observed in table I. This, with the existing number of distinct imbalanced datasets, makes the search for a method that is able to be useful in every occasion very difficult. Furthermore, it was also noticeable that SMOTE variations present high quality results, being very interesting to continue their study in order to better understand how SMOTE can be used to improve classification of imbalanced datasets.

TABLE I: Related Work Sampling Techniques

Reference	Uses Over-sampling	Uses Under-sampling	Sampling Techniques
Costa et al. [3]	✓	×	SMOTE, SafeLevel-SMOTE, Borderline-SMOTE, ADASYN, AHC, ADOMS, ROS, SMOTE-TL, SMOTE-ENN
Santos et al. [12]	✓	×	SMOTE, ROS, ADASYN, Borderline-SMOTE, SafeLevel-SMOTE, SMOTE-ENN, ADOMS, AHC, CBO, MWMOTE, SPIDER
Zhang et al. [16]	✓	✓	RUS, ROS, SMOTE, MetaCost, Bagging, AdaBoost, EasyEnsemble, UnderBagging, EM1vs1, EDBC
Fernandez et al. [18]	✓	✓	ROS, RUS, SMOTE
Rio et al. [19]	✓	✓	ROS, RUS
Triguero et al. [20]	×	✓	RUS, EUS
Park and Ha [21]	✓	×	SMOTE
Park et al. [22]	✓	×	SMOTE
Chai et al. [23]	×	✓	-
Kovács [24]	✓	×	85 SMOTE Variants

III. Experimental Setup

In this section it is given a brief description of the datasets used and associated pre-processing operations as well as of the experimental pipeline and its main stages.

A. Datasets

This study was conducted using 53 datasets, which included both binary as well as multiclass sets, available on GitHub and UC Irvine machine learning repository. Several differences were found upon the analysis of the files, concerning their format as there were present .dat, .data, .csv, .txt and even an .arff file. There were also cases were the dataset was split in several files, which had to be formerly grouped. Other factors were also relevant, mainly the number of parameters per data set where the average was approximately 11 and the number of rows per file which had an average of 1929 rows. Additionally, the imbalance ratio, which is one major factor for this study, also varies from a range of values as there are imbalanced

datasets where the positive class has a 3% ratio and others with 35%. All these characteristics can be seen in table II of the Annex (section VI).

Several pre-processing operations are performed on these datasets as well. Firstly, missing values in the predictive attributes are replaced by either the mean or the mode of that column, depending if the attribute is numerical or categorical, respectively. In .dat files with the appropriate headers, this is known beforehand, otherwise it is inferred by the total number of different values present. After this, the categorical predictive attributes are transformed in N binary indicator attributes for compatibility reasons, with N being equal to the number of different values of that attribute. Finally, if the dataset represents a multi-class problem, it is converted to a binary one, with the class possessing the fewer instances above a certain threshold representing the minority class, and all the other classes the majority one.

B. Experimental Pipeline

The experimental pipeline can be seen in Figure 1. For each dataset, the data is first split in a training and a testing set in a stratified manner, after undergoing the pre-processing operations mentioned in the previous section. The training set is then used for the extraction of the original meta-features and for the training of the classifier, in this case a Random Forest ensemble. The classifier's performance is then evaluated using the test set, with the calculated AUC being recorded.

Next, each SMOTE variant is applied a maximum of three times to the training dataset, each iteration with a different minority-majority class ratio: 50%, 80% and 100%. Note that in some datasets not all ratios were applied since this depends on their original ratio; for example, if in a dataset the minority-majority ratio is already at 60%, to achieve a ratio of 50%, examples would have to be taken out of the minority class.

The meta-features are then extracted again from the over-sampled set, with a new classifier being trained on this data. Finally, its performance is evaluated in the same manner as in the previous classifier.

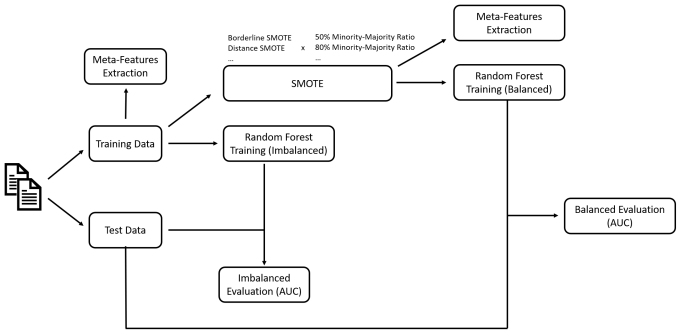


Fig. 1: Experimental Pipeline

IV. Results

A. General Tendencies

On average, SMOTE variants using a ratio of 0.5 performed slightly better. It is also interesting to note that 22 datasets didn't see their original AUC improved nor deteriorated (on average) by using a SMOTE algorithm and 12 saw their original AUC worsen (on average), which, in total, accounts for $\sim$64% of the total number of datasets. This made it difficult to evaluate the positive results since they weren't numerous enough to distinguish patterns.

Some dataset characteristics were analysed in order to evaluate their impact on the success of SMOTE. The results of this assessment are presented below.

1) Number of Attributes: Regarding the number of attributes, datasets with fewer parameters seemed to have a greater average AUC difference. Both the greatest and smallest average AUC differences were concentrated in the range of 4-15 attributes, which is inconclusive. For datasets over 21 features, there weren't sufficient examples to draw sustainable evidence.

2) Average Landmark Classification: The average landmark classification is the average of the AUC of multiple classifiers on the original datasets. The presented SMOTE variants were able to improve the original AUC of datasets with all different average landmark classifications, except for those who had an average landmark classification greater than 0.96.

3) Initial Imbalance Ratio: There was a higher concentration of improved AUC (on average) in less imbalanced datasets (imbalance ratio > 0.8). With SMOTE variants using a ratio of 0.5, there were no improvements in more imbalanced datasets (imbalance ratio < 0.8), probably meaning that there weren't enough synthetic examples to make up for the imbalance.

4) Dataset Size: Almost all of the datasets that had their original AUC improved had less than 1,000 examples. However, most of the datasets that didn't have their original AUC improved also had less than 1000 examples. There were few datasets with size out of this range ($>$ 1,000), leaving us with inconclusive results.

B. Specific Cases

Some odd cases were identified in which the AUC difference between the AUC scores after and before applying SMOTE had an opposite tendency than the average for that dataset. For each of these cases, the correlation between the meta-features and the difference between the AUC scores was calculated. After this, it was observed where the most correlated meta-features' value, for that specific smote variant, is located by analysing the box plot of that meta-feature' values for a specific dataset.

1) MSMOTE-abalone19: On the dataset abalone19 the most correlated meta-features that influenced the AUC difference were:

- density
- leaves corrob mean
- leaves corrob sd
- naive bayes mean
- tree shape mean
- tree shape sd

In this case, in all three ratios used resulted in a odd case.It is possible to verify that for MSMOTE these values are all inserted outside of were most of the values for the other variants are. Also, for the naive bayes mean meta-feature, the values were very off the bounds of the box plot. This can be seen in figures 10, 11, and 12

2) DSMOTE yeast-0-3-5-9 vs 7-8 - 1.0 Oversampling: On the dataset yeast-0-3-5-9_vs_7-8 the most correlated meta-features that influenced the AUC difference were:

- density
- best node mean
- leaves corrob sd
- l2 mean
- n4 mean
- n4 sd

For this variant, only one ratio had a odd behavior, and it is possible to see that two of the meta-features values were outside the bounds of the box plot, and the remaining were in the limits of the same. The results are present in figure 13.

V. Conclusions

This work analyzed the effects on predictive performance when applying SMOTE to a dataset, with 35 variants of the oversampling algorithm being tested on 53 different datasets. For that purpose, a custom pipeline was developed, with results showing that small datasets with already high landmark classifications don't gain much from the use of this algorithm. Another observation was that more imbalanced datasets benefit from a higher SMOTE ratio (higher number of synthetic examples), however, most results were inconclusive due to the lack of successful examples. It was possible verify that density meta-feature had an impact in all of the odd cases analyzed, so it is possible that this meta-feature can have an influence the results. Also, the leaves corrb mean and leaves corrob sd also appeared in both cases with a large correlation with the AUC scores difference.

Future work would include repeating the experimental setup on additional datasets with characteristics that complement the previously seen ones (bigger datasets, with higher number of attributes and with smaller landmark classification), in order to be able to draw more conclusions.

References

[1] Aouatef Mahani and Ahmed Riad Baba Ali. Classification Problem in Imbalanced Datasets. In *Recent Trends in Computational Intelligence*. IntechOpen, may 2020.

[2] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16:321–357, jun 2011.

[3] Afonso José Costa, Miriam Seoane Santos, Carlos Soares, and Pedro Henriques Abreu. Analysis of Imbalance Strategies Recommendation using a Meta-Learning Approach. Technical report, 2020.

[4] Cláudio Rebelo de Sá, Wouter Duivesteijn, Carlos Soares, and Arno Knobbe. Exceptional preferences mining. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9956 LNAI, pages 3–18. Springer Verlag, 2016.

[5] Chumphol Bunkhumpornpat, Krung Sinapiromsaran, and Chidchanok Lursinsap. Safe-level-SMOTE: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 5476 LNAI, pages 475–482. Springer, Berlin, Heidelberg, 2009.

[6] Hui Han, Wen Yuan Wang, and Bing Huan Mao. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *Lecture Notes in Computer Science*, volume 3644, pages 878–887. Springer Verlag, 2005.

[7] Haibo He, Yang Bai, Edwardo A. Garcia, and Shutao Li. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *Proceedings of the International Joint Conference on Neural Networks*, pages 1322–1328, 2008.

[8] Julien Ah-Pine. An Efficient and Effective Generic Agglomerative Hierarchical Clustering Approach. Technical report, 2018.

[9] Sheng Tang and Si Ping Chen. The generation mechanism of synthetic minority class examples. *5th Int. Conference on Information Technology and Applications in Biomedicine, ITAB 2008 in conjunction with 2nd Int. Symposium and Summer School on Biomedical and Health Engineering, IS3BHE 2008*, pages 444–447, 2008.

[10] Gustavo E. A. P. A. Batista, Ronaldo C. Prati, and Maria Carolina Monard. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, 6(1):20–29, jun 2004.

[11] Dennis L. Wilson. Asymptotic Properties of Nearest Neighbor Rules Using Edited Data. *IEEE Transactions on Systems, Man and Cybernetics*, 2(3):408–421, 1972.

[12] Miriam Seoane Santos, Jastin Pompeu Soares, Pedro Henrigues Abreu, Helder Araujo, and Joao Santos. Cross-validation for imbalanced datasets: Avoiding overoptimistic and overfitting approaches [Research Frontier]. *IEEE Computational Intelligence Magazine*, 13(4):59–76, nov 2018.

[13] Taeho Jo and Nathalie Japkowicz. Class imbalances versus small disjuncts. *ACM SIGKDD Explorations Newsletter*, 6(1):40–49, jun 2004.

[14] Sukarna Barua, Md Monirul Islam, Xin Yao, and Kazuyuki Murase. MWMOTE - Majority weighted minority oversampling technique for imbalanced data set learning. *IEEE Transactions on Knowledge and Data Engineering*, 26(2):405–425, feb 2014.

[15] Jerzy Stefanowski and Szymon Wilk. Selective pre-processing of imbalanced data for improving classification performance. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 5182 LNCS, pages 283–292. Springer, Berlin, Heidelberg, 2008.

[16] Xueying Zhang, Ruixian Li, Bo Zhang, Yunxiang Yang, Jing Guo, and Xiang Ji. An instance-based learning recommendation algorithm of imbalance handling methods. *Applied Mathematics and Computation*, 351:204–218, 2019.

[17] Joffrey L. Leevy, Taghi M. Khoshgoftaar, Richard A. Bauder, and Naeem Seliya. A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5(1):42, dec 2018.

[18] Alberto Fernández, Sara del Río, Nitesh V. Chawla, and Francisco Herrera. An insight into imbalanced Big Data classification: outcomes and challenges. *Complex & Intelligent Systems*, 3(2):105–120, jun 2017.

[19] Sara Del Río, José M Benítez, and Francisco Herrera. Analysis of Data Preprocessing Increasing the Oversampling Ratio for Extremely Imbalanced Big Data Classification. *2015 IEEE Trustcom/BigDataSE/ISPA*, 2, 2015.

[20] Triguero, M. Galar, D. Merino, J. Maillo, H. Bustince, and F. Herrera. 2016 IEEE Congress on Evolutionary Computation, CEC 2016, 2016.

[21] Seoung Hun Park and Young Guk Ha. Large imbalance data classification based on MapReduce for traffic accident prediction. In *Proceedings - 2014 8th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2014*, pages 45–49. Institute of Electrical and Electronics Engineers Inc., 2014.

[22] Seong hun Park, Sung min Kim, and Young guk Ha. Highway traffic accident prediction using VDS big data analysis. *Journal of Supercomputing*, 72(7):2815–2831, jul 2016.

[23] Kevin E.K. Chai, Stephen Anthony, Enrico Coiera, and Farah Magrabi. Using statistical text classification to identify health information technology incidents. *Journal of the American Medical Informatics Association*, 20(5):980–985, 2013.

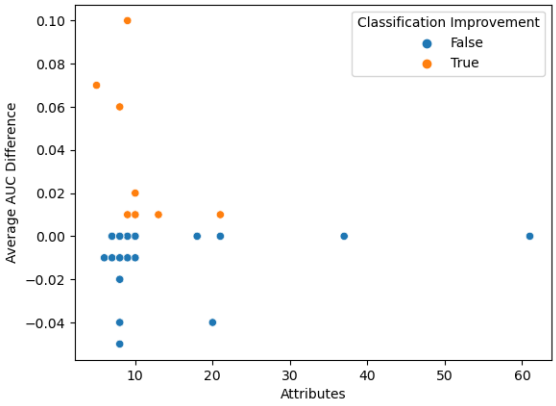
[24] György Kovács. Smote-variants: A python implementation of 85 minority oversampling techniques. *Neurocomputing*, 366(June):352–354, nov 2019.

[25] György Kovács. An empirical comparison and evaluation of minority oversampling techniques on a large number of imbalanced datasets. *Applied Soft Computing*, 83(July):105662, oct 2019.

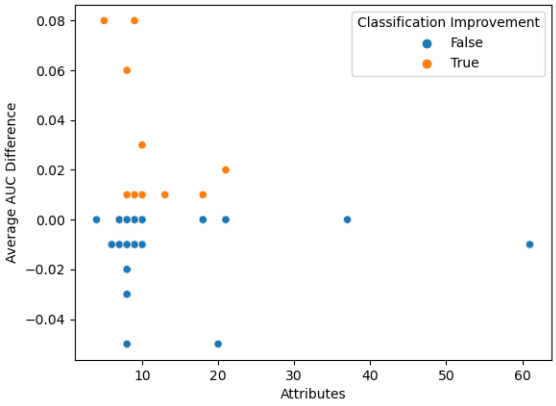
VI. ANNEX

TABLE II: Dataset's Description

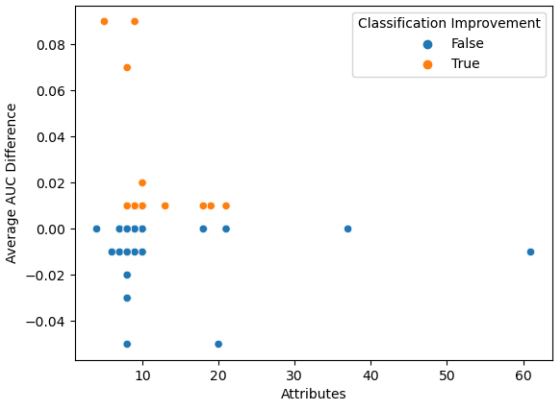
Dataset	Percentages (minority-majority)	Nº of variables	Nº of examples	Binary / Multiclass	Format
Absenteeism at work Data Set			740	Multiclass	.csv
Activity Recognition from Single Chest-Mounted Accelerometer Data Set				Multiclass	.csv
Audit Data Data Set	39% - 61%	18	777	Binary	.csv
AutoUniv Data Set				Binary	mixed
Balance Scale Data Set			625	Multiclass	.data
Bank Marketing Data Set			45211	Binary	.csv
abalone-17_vs_7-8-9-10	3% - 97%	8	2338	Binary	.dat
abalone-19_vs_10-11-12-13	2% - 98%	8	1622	Binary	.dat
abalone-20_vs_8-9-10	1% - 99%	8	1916	Binary	.dat
abalone-21_vs_8	2% - 98%	8	581	Binary	.dat
abalone-3_vs_11	3% - 97%	8	502	Binary	.dat
abalone19	1% - 99%	8	4174	Binary	.dat
abalone9-18	6% - 94%	8	731	Binary	.dat
Activity recognition with healthy older people using a batteryless wearable sensor Data Set				Multiclass	.csv
Adult	24% - 76%	14		Binary	.data
Anuran Calls (MFCCs) Data Set			7195	Multiclass	.csv
car-good	4% - 96%	6	1728	Binary	.dat
car-vgood	4% - 96%	6	1729	Binary	.dat
cleveland-0_vs_4_no_null	8% - 92%	13	178	Binary	.dat
dermatology-6	6% - 94%	34	358	Binary	.dat
ecoli1	23% - 77%	7	336	Binary	.dat
Glass Identification	4% - 96%	10	214	Multiclass	.data
glass-0-1-2-3_vs_4-5-6	24% - 76%	9	214	Binary	.dat
glass-0-1-6_vs_2	9% - 91%	9	192	Binary	.dat
glass0	33% - 67%	9	214	Binary	.dat
glass1	36% - 64%	9	214	Binary	.dat
glass2	22% - 78%	9	214	Binary	.dat
glass4	6% - 94%	9	214	Binary	.dat
glass5	4% - 96%	9	214	Binary	.dat
glass6	14% - 86%	9	213	Binary	.dat
haberman	26% - 74%	3		Binary	.dat
Hepatitis	21% - 79%	19	155	Binary	.data
pc1	7% - 93%	21	1108	Binary	.dat
pima	35% - 65%	8	767	Binary	.dat
Statlog (German Credit Data)	30% - 70%	20	1000	Binary	.data
Thyroid Disease	5% - 95%	21	3163	Binary	.data
vehicle0	23% - 77%	18	845	Binary	.dat
vehicle1	26% - 74%	18	845	Binary	.dat
vehicle2	26% - 74%	18	845	Binary	.dat
vehicle3	25% - 75%	18	845	Binary	.dat
wisconsin	35% - 65%	9	683	Binary	.dat
yeast-0-2-5-6_vs_3-7-8-9	10% - 90%	8	1004	Binary	.dat
yeast-0-2-5-7-9_vs_3-6-8	10% - 90%	8	1004	Binary	.dat
yeast-0-3-5-9_vs_7-8	10% - 90%	8	506	Binary	.dat
yeast-0-5-6-7-9_vs_4	10% - 90%	8	528	Binary	.dat
yeast-1_vs_7	7% - 93%	7	459	Binary	.dat
yeast-1-2-8-9_vs_7	3% - 97%	8	947	Binary	.dat
yeast-1-4-5-8_vs_7	4% - 96%	8	693	Binary	.dat
yeast-2_vs_4	10% - 90%	8	514	Binary	.dat
yeast-2_vs_8	4% - 96%	8	482	Binary	.dat
yeast1	29% - 71%	8	1484	Binary	.dat
yeast3	11% - 89%	8	1484	Binary	.dat
yeast4	3% - 97%	8	1484	Binary	.dat
yeast5	3% - 97%	8	1484	Binary	.dat
yeast6	2% - 98%	8	1484	Binary	.dat



(a) SMOTE variants using ratio of 0.5

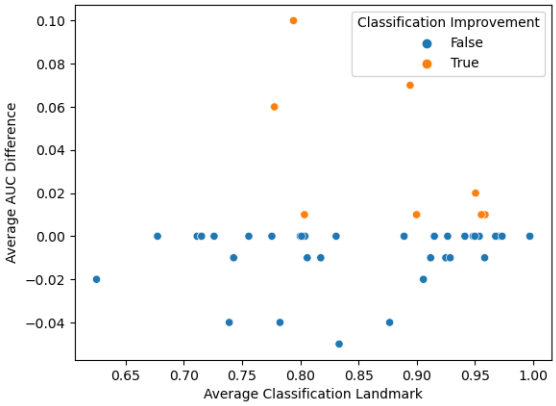


(b) SMOTE variants using ratio of 0.8

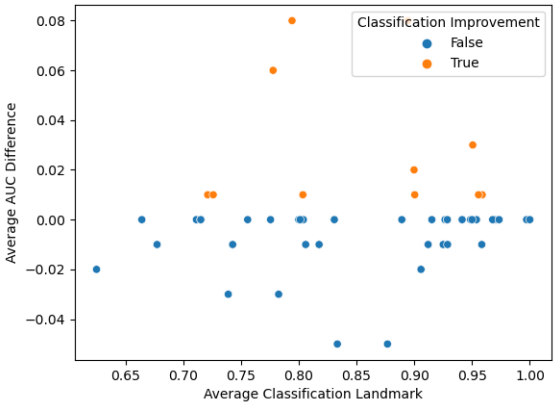


(c) SMOTE variants using ratio of 1.0

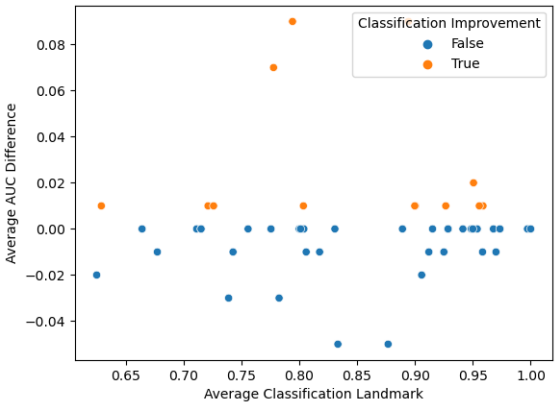
Fig. 2: Number of attributes and the average AUC difference of a dataset



(a) SMOTE variants using ratio of 0.5

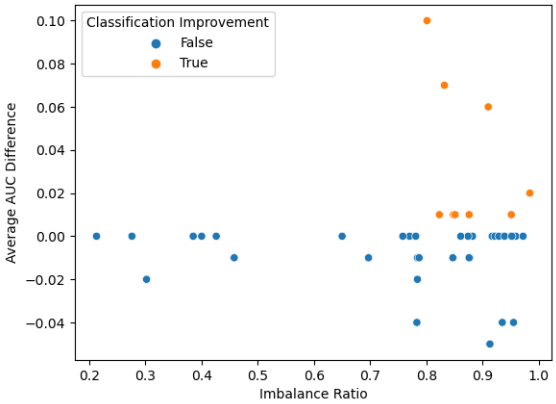


(b) SMOTE variants using ratio of 0.8

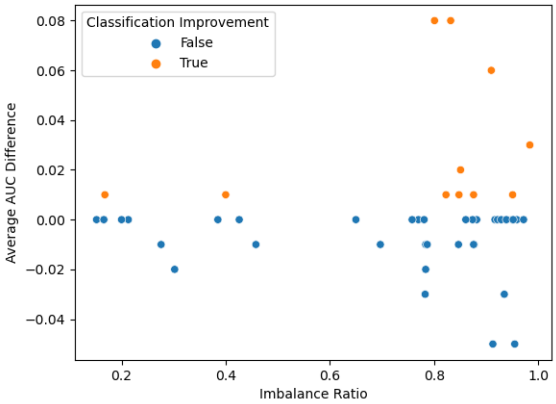


(c) SMOTE variants using ratio of 1.0

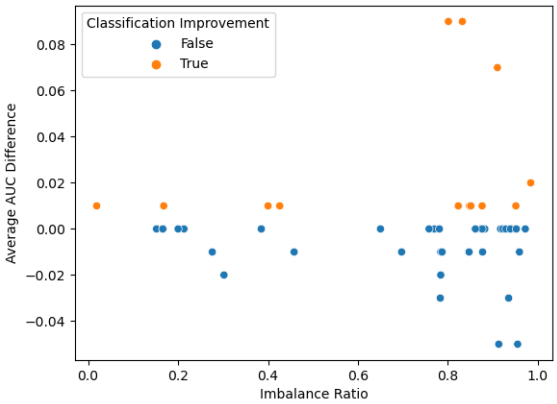
Fig. 3: Average landmark classification and average AUC difference of a dataset



(a) SMOTE variants using ratio of 0.5

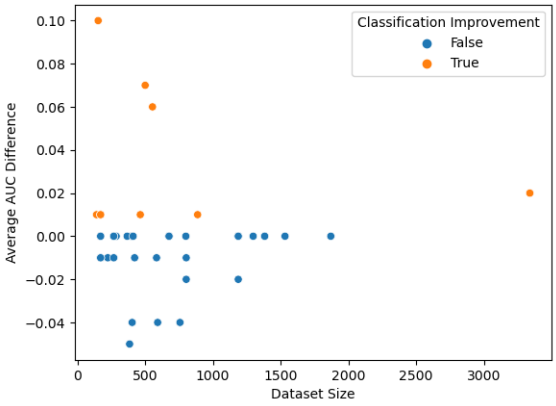


(b) SMOTE variants using ratio of 0.8

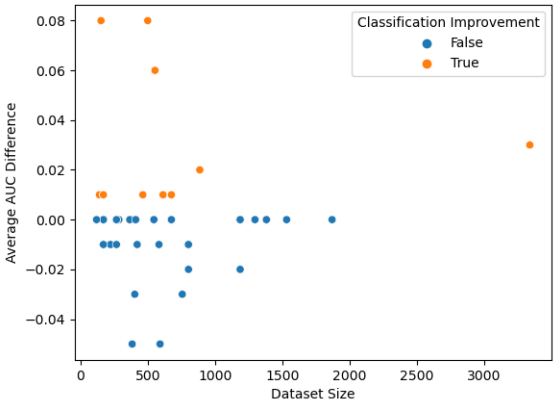


(c) SMOTE variants using ratio of 1.0

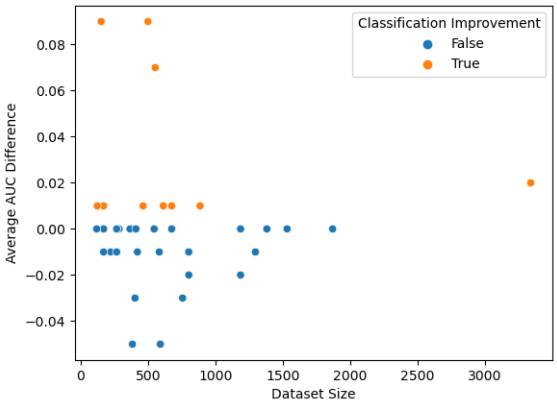
Fig. 4: Initial imbalance ratio and average AUC difference of a dataset



(a) SMOTE variants using ratio of 0.5



(b) SMOTE variants using ratio of 0.8



(c) SMOTE variants using ratio of 1.0

Fig. 5: Size and average AUC difference of a dataset

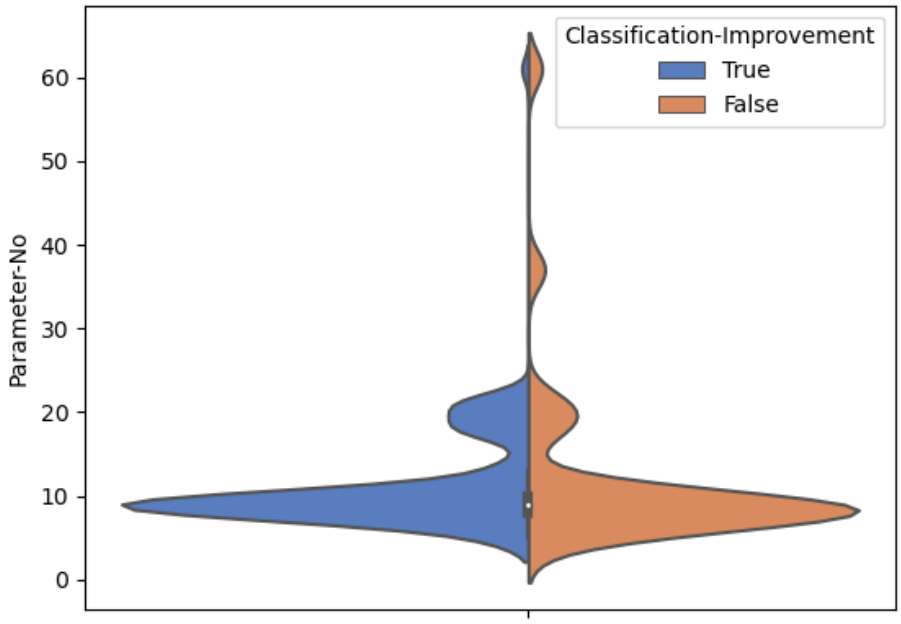


Fig. 6: Violin plot relating the number of attributes of a dataset and its classification improvement

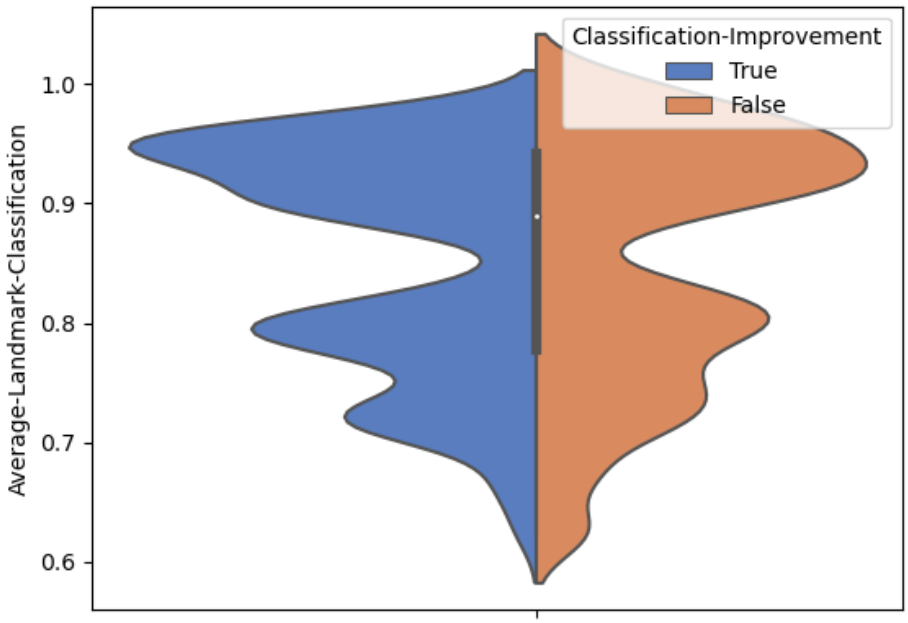


Fig. 7: Violin plot relating the average landmark classification of a dataset and its classification improvement

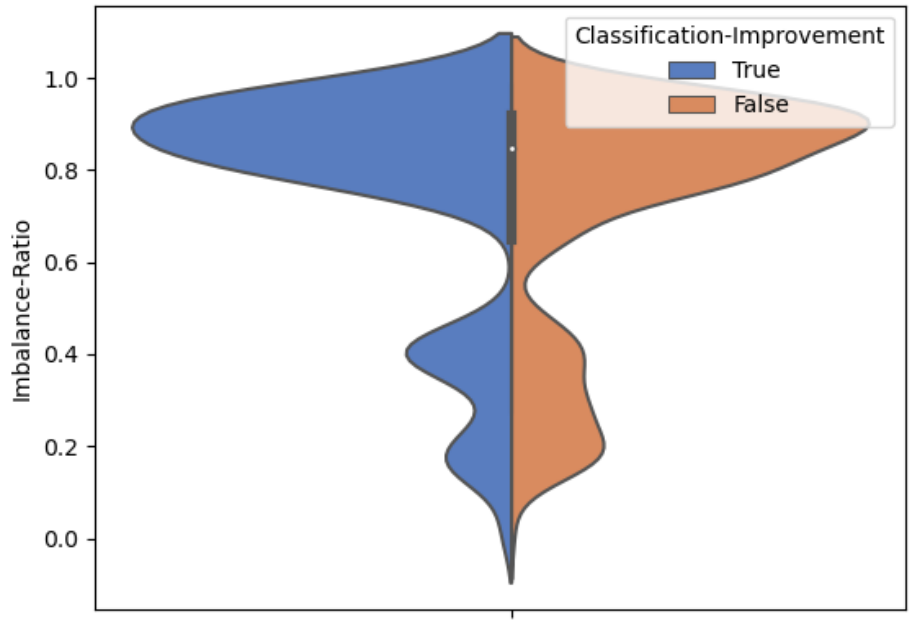


Fig. 8: Violin plot relating the initial imbalance ratio of a dataset and its classification improvement

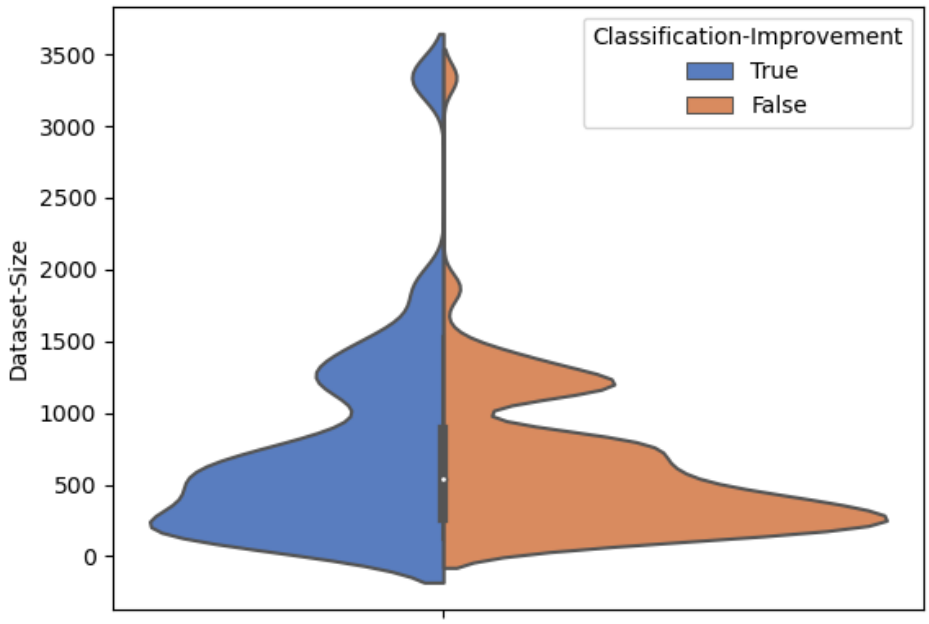


Fig. 9: Violin plot relating a dataset's size and its classification improvement

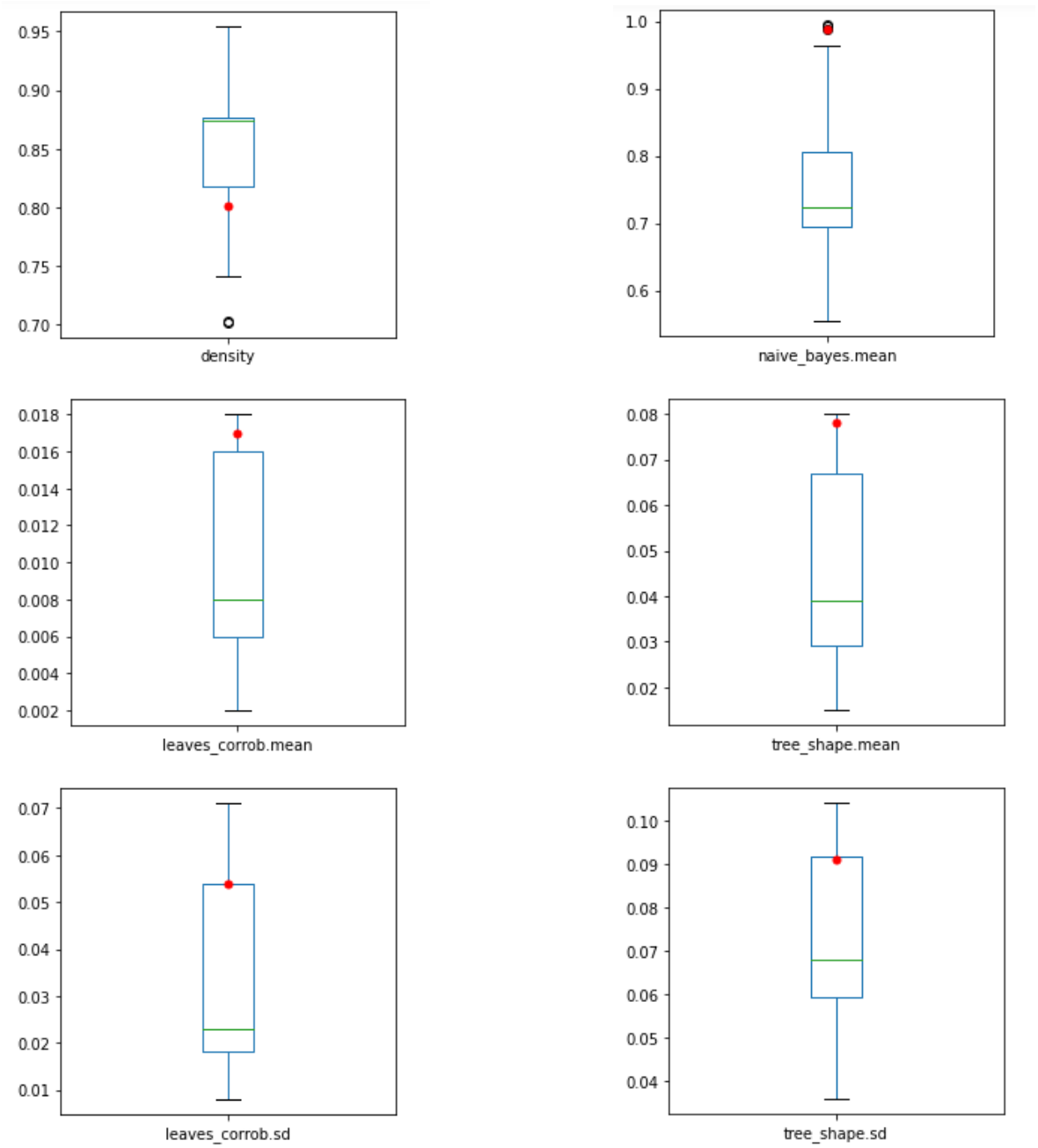


Fig. 10: Box plot of most correlated meta-features in abalone19 dataset and value for MSMOTE 0.5 ratio

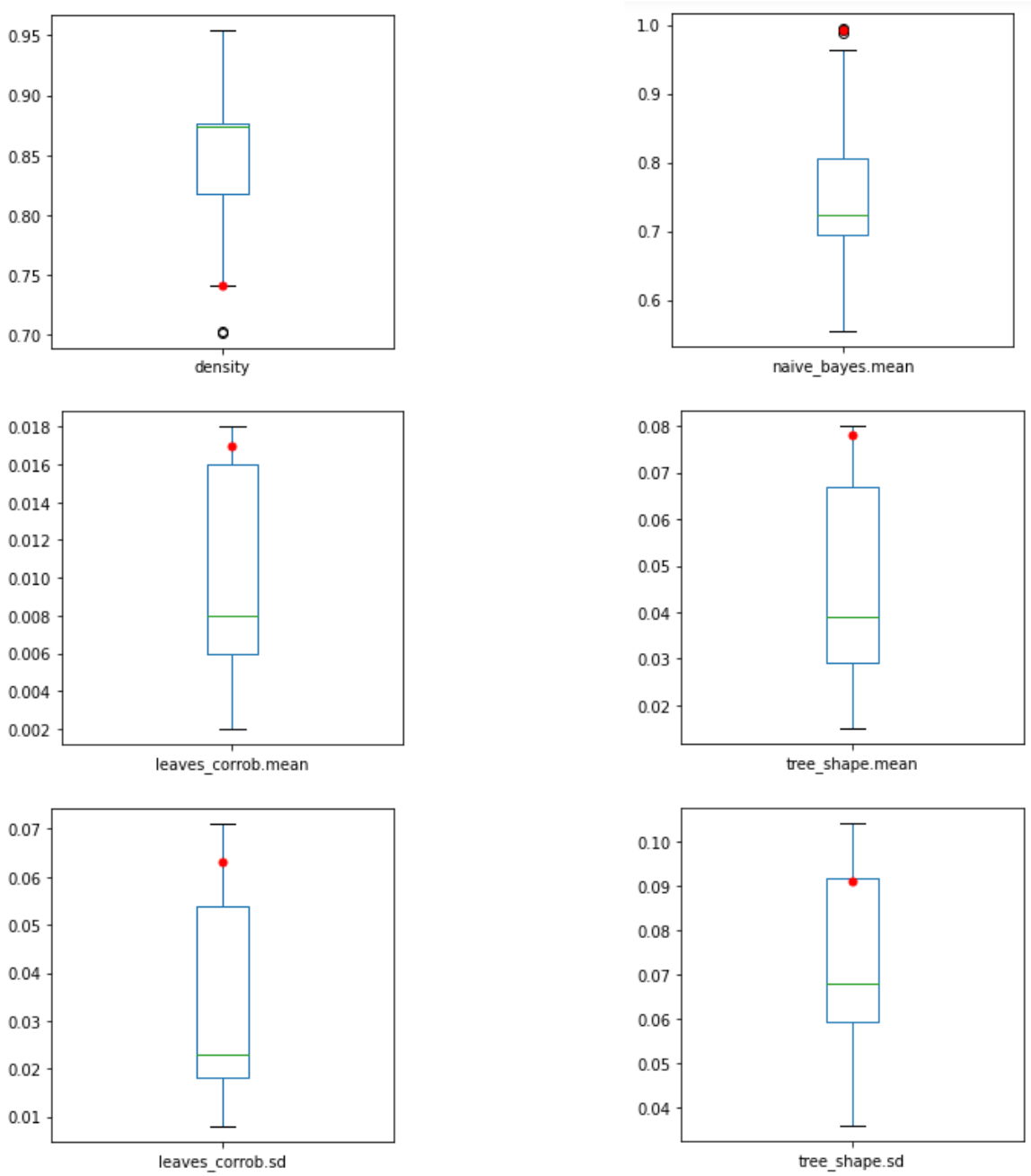


Fig. 11: Box plot of most correlated meta-features in abalone19 dataset and value for MSMOTE 0.8 ratio

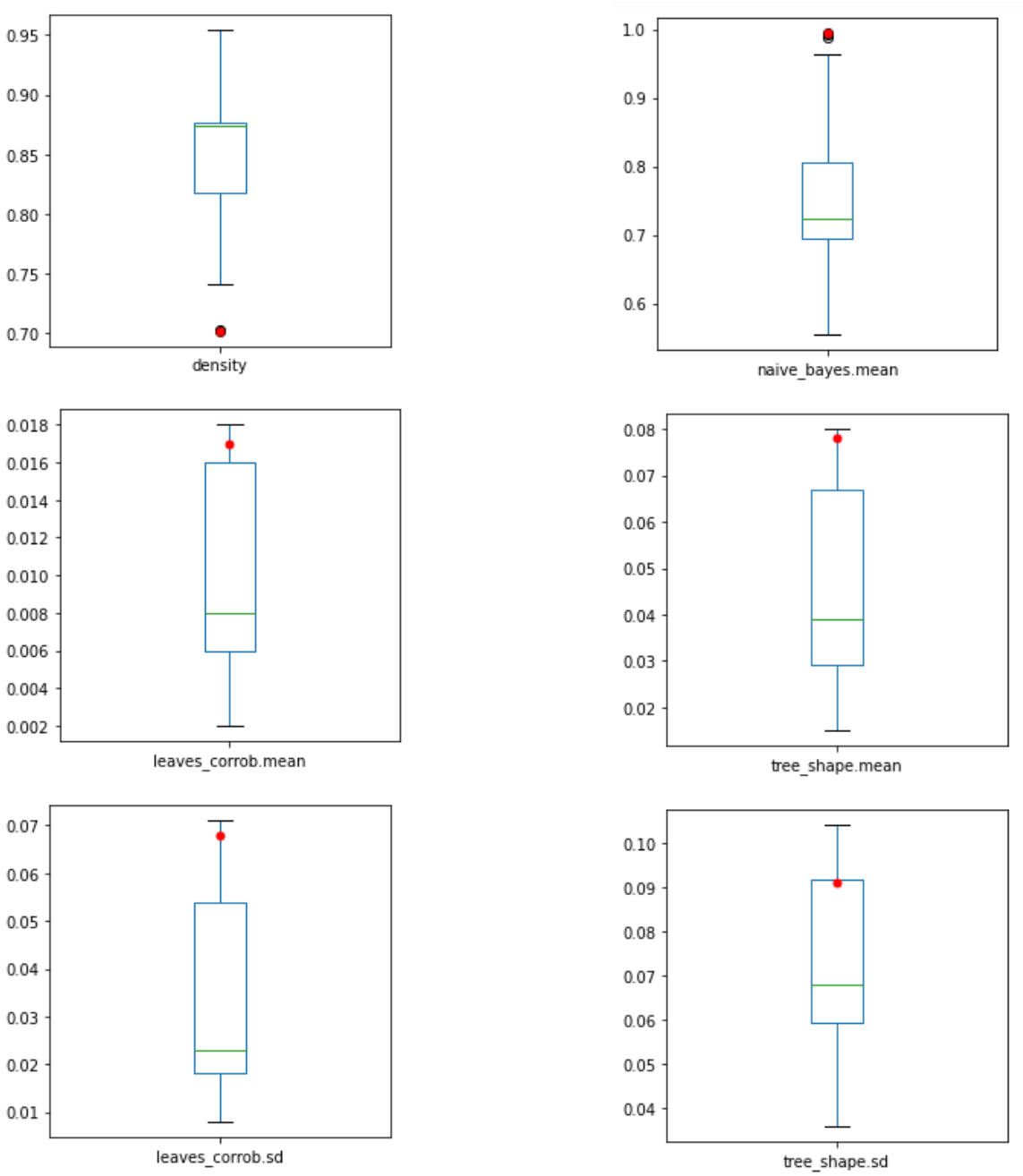


Fig. 12: Box plot of most correlated meta-features in abalone19 dataset and value for MSMOTE 1.0 ratio

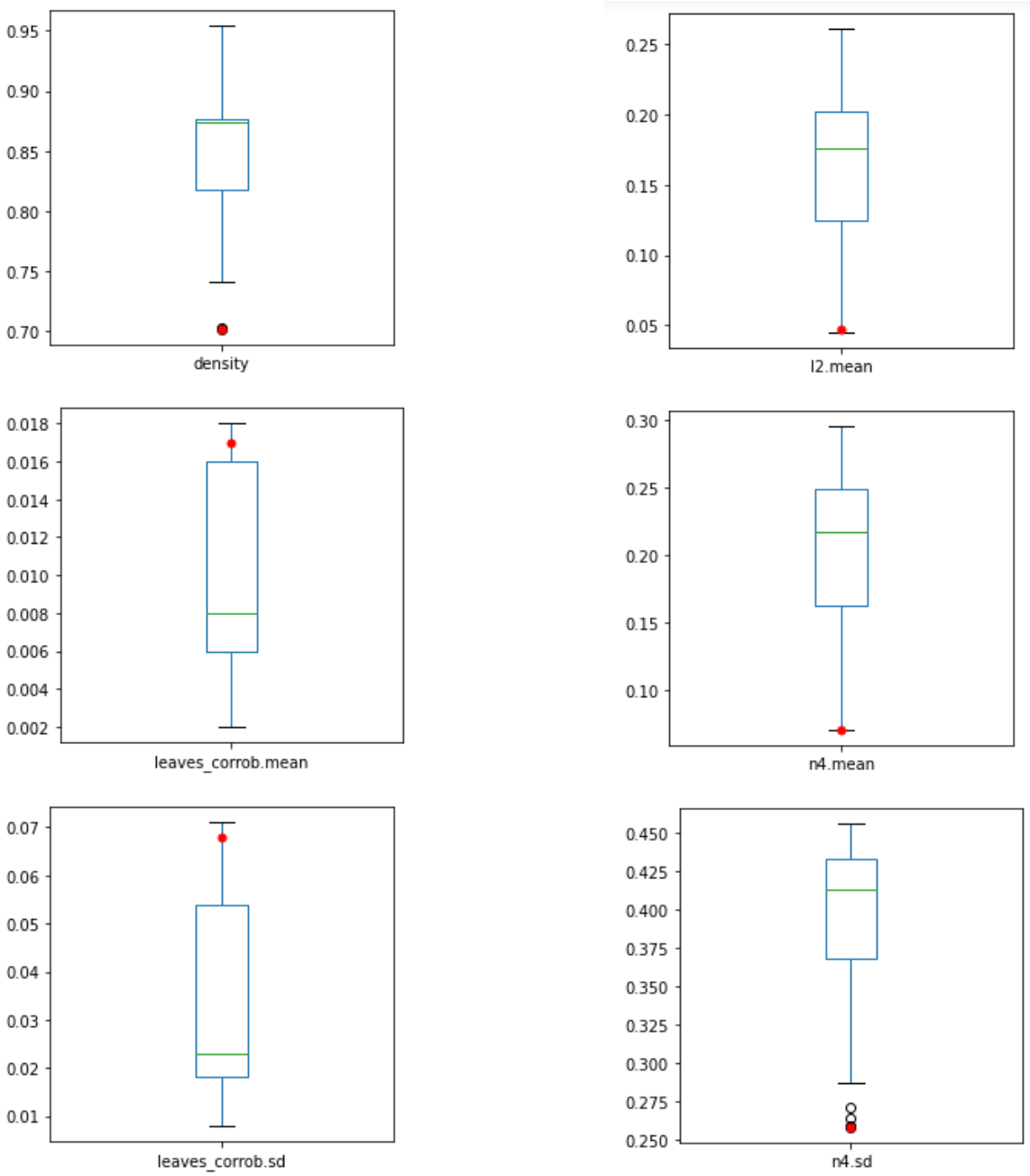


Fig. 13: Box plot of most correlated meta-features in yeast-0-3-5-9-vs-7-8 dataset and value for DSMOTE 1.0 ratio