Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 2 (24)
ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

UDC 004.8 DOI: https://doi.org/10.30837/ITSSI.2023.24.161

D. TESLENKO, A. SOROKINA, A. KHOVRAT, N. HULIIEV, V. KYRIY

COMPARISON OF DATASET OVERSAMPLING ALGORITHMS AND THEIR APPLICABILITY TO THE CATEGORIZATION PROBLEM

The **subject** of research in the article is the problem of classification in machine learning in the presence of imbalanced classes in datasets. The **purpose** of the work is to analyze existing solutions and algorithms for solving the problem of dataset imbalance of different types and different industries and to conduct an experimental comparison of algorithms. The article solves the following tasks: to analyze approaches to solving the problem – preprocessing methods, learning methods, hybrid methods and algorithmic approaches; to define and describe the oversampling algorithms most often used to balance datasets; to select classification algorithms that will serve as a tool for establishing the quality of balancing by checking the applicability of the datasets obtained after oversampling; to determine metrics for assessing the quality of classification for comparison; to conduct experiments according to the proposed methodology. For clarity, we considered datasets with varying degrees of imbalance (the number of instances of the minority class was equal to 15, 30, 45, and 60% of the number of samples of the majority class). The following **methods** are used: analytical and inductive methods for determining the necessary set of experiments and building hypotheses regarding their results, experimental and graphic methods for obtaining a visual comparative characteristic of the selected algorithms. The following **results** were obtained: with the help of quality metrics, an experiment was conducted for all algorithms on two different datasets – the Titanic passenger dataset and the dataset for detecting fraudulent transactions in bank accounts. The obtained results indicated the best applicability of SMOTE and SVM SMOTE algorithms, the worst performance of *Borderline* SMOTE and *k-means* SMOTE, and at the same time described the results of each algorithm and the potential of their usage. **Conclusions**: the application of the analytical and experimental method provided a comprehensive comparative description of the existing balancing algorithms. The superiority of oversampling algorithms over undersampling algorithms was proven. The selected algorithms were compared using different classification algorithms. The results were presented using graphs and tables, as well as demonstrated in general using heat maps. Conclusions that were made can be used when choosing the optimal balancing algorithm in the field of machine learning.

Keywords: categorization; machine learning; methods of balancing; data generation methods; dataset; unbalanced datasets.

Introduction

Unbalanced data classification is a problem in which the proportional sizes of the classes in a dataset differ significantly. In this case, at least one class has only a few samples – the minority class – and the rest falls into another class – the majority class (Fig. 1).

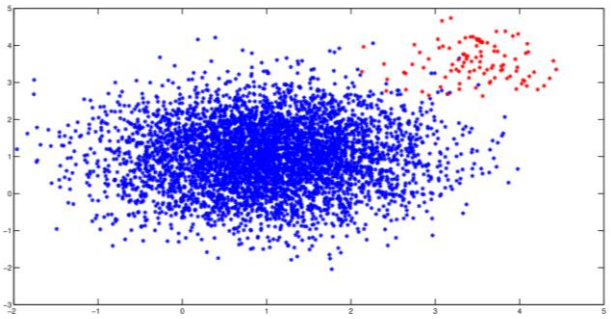

Fig. 1. An example of an unbalanced data problem [1]

This problem affects the performance of the classifier, which can be seen from the fact that when training on an unbalanced dataset, the algorithm begins to adjust to the influence of a more numerous class, which causes a shift in accuracy. This means that machine learning decisions are made with different efficiency on the majority and minority classes, and the classifier distinguishes samples of certain classes with greater accuracy, namely samples of the majority classes, and the results of identifying samples with low accuracy are in the minority classes. This is because classifiers strive to achieve the best possible results during training. Since "normal" observations are predominant in number, ML algorithms focus on learning the behavior of the "normal" class [2]. Consequently, the model can achieve greater accuracy due to the fact that it pays more attention to studying the properties and identifying the majority class rather than a uniform distribution of powers. This is because, for example, in very unbalanced datasets, the algorithm will have good accuracy even if it always categorizes any instances as members of the majority class [3].

One of the main obstacles in learning from imbalanced data is that the minority class is usually the class of interest, which is often the case in applications such as medical diagnosis, face recognition, tampering, error, or fraudulent transaction detection [5]. The most

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2023. No. 2 (24)

popular methods used to eliminate or ignore data imbalance are synthesizing new instances of the minority class, oversampling the minority class, undersampling the minority class, and a method of tuning the cost function of learning algorithms to make misclassification of minority class samples more important than misclassification of majority class samples [6]. This makes it possible to achieve a more unbiased attitude of ML algorithms to classes.

Analysis of recent research and publications

The problem of unbalanced data distribution is quite common in applied problems. There are three main approaches to classification based on unbalanced data. Also, approaches to solving imbalance problems in data classification are sometimes divided into the following: preprocessing methods, *cost-sensitive* learning methods, hybrid methods, and algorithmic approaches.

The study will focus on methods of preprocessing datasets. Below is a diagram illustrating the hierarchy of approaches (Fig. 2).

Pre-processing approaches are those that are performed on the training data. They are divided into *Sampling Methods* and *Feature Selection and Extraction*.

Pre-processing methods are used to obtain more balanced training data. Pre-processing approaches are also called data-driven approaches and work by directly acting on the data space in an attempt to reduce the imbalance ratio between classes.

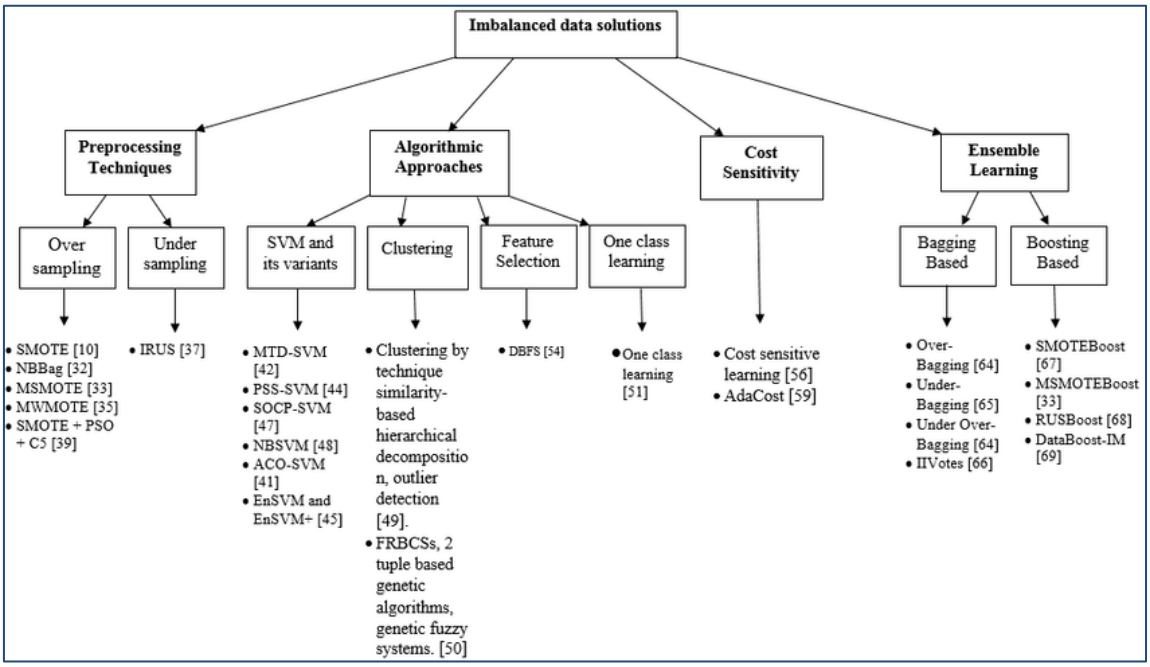

Fig. 2. Classification of approaches for unbalanced data [7]

Sampling Methods is a simple and popular approach for balancing the class distribution of training data. The original data space is balanced by using one of the methods to eliminate redundant instances or generate somehow insufficient information in the sample. The main idea of resampling data instances is to obtain balanced classes. This process is repeated until a balanced dataset is achieved. The resampling approach is based on techniques that eliminate the imbalanced set by adding or removing samples from the dataset to reduce the biased behavior of the unbalanced dataset, thus resizing the training dataset.

Another approach, *Feature Selection and Extraction*, is the selection of a subset of relevant features or attributes from large data sets. It helps to improve the performance of the classifier.

The sampling approach, in turn, is divided into t hree more variants:

– undersampling of the majority class – creating a subset of the original data by removing selected samples from the class, i.e., selecting the dominant data from the majority class and selecting a number of examples of each category that is too large compared to the others (Fig. 3);

Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 2 (24)
ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

– oversampling of the minority class – creating a superset of the original dataset, or forming new samples from existing ones, or replicating existing ones, i.e. replicating examples of the minority class and artificially "reproducing" examples of all categories with a smaller presence (Fig. 3);

– hybrid methods that combine the previous two approaches for a more natural distribution of data.

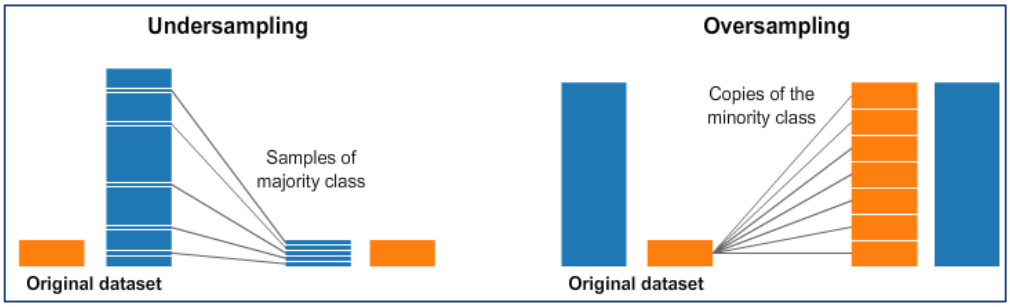

Fig. 3. Undersampling and oversampling [8]

The simplest methods of preprocessing a dataset are *Random Undersampling* (RU) and *Random Oversampling* (RO). The main disadvantage of random oversampling is that it can discard potentially useful data that is important for training. On the other hand, for random oversampling, overfitting can occur, as this process generates exact copies of existing instances. To prevent this situation, other solutions have been proposed. For example, in the SMOTE algorithm (Synthetic Minority Oversampling Technique), "artificial" minority class instances are generated by interpolating several randomly selected adjacent minority instances (nearest neighbors) to increase the number of minority class instances in the training set. Nearest neighbors are found by Euclidean distance. The SMOTE algorithm generates the same amount of synthetic data for each original minority instance without considering the neighboring examples from the majority classes. This can increase the frequency of overlap between classes. Therefore, some variants of the method have been proposed to reduce the noticeable limitations, namely *Borderline*-SMOTE, SMOTE SVM, etc.

Overview of oversampling algorithms

One way to deal with the problem of unbalanced data is to create new samples in underrepresented classes. The most naïve strategy is to create new samples by randomly selecting and replacing the current available samples [9]. In the RO algorithm, the selected samples are simply copied randomly in order to increase the importance of the minority class.

As a result, the majority class does not dominate the others during the learning process. Thus, all classes are represented by the decision function. In addition, *Random Oversampler* allows to select data of mixed type.

The SMOTE algorithm synthesizes new minority instances between existing (real) minority instances. SMOTE draws lines between the instances of the minority class and then represents the new, synthetic minority instances somewhere on these lines [10]. If there are instances in the minority class that are distant and appear in the majority class, this creates a problem for SMOTE. The algorithm can start creating a linear bridge with the majority class (Fig. 4).

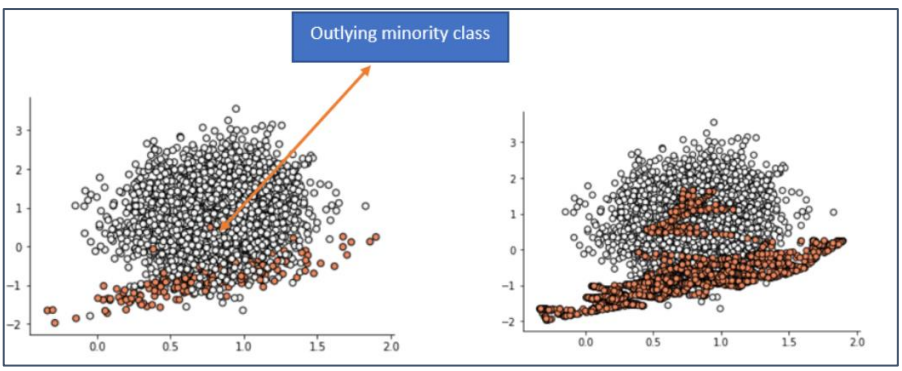

Fig. 4. Disadvantage of the SMOTE algorithm [4]

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2023. No. 2 (24)

In this regard, SMOTE offers three additional options for creating samples: *Borderline*-SMOTE, SMOTE SVM, and SMOTE *k-means*. These methods focus on samples near the boundary of the optimal decision function and will create samples in the direction opposite to that of the nearest neighbor class.

Recent work on data imbalance has pointed out some important issues related to performance degradation, namely:

– the presence of small disjuncts; this means that the minority class can be divided into many subclusters with very few examples in each, surrounded by examples of the majority class [11];

– overlap between classes; there are often examples from different classes with very similar characteristics, in particular if they are located in areas around the boundaries of solutions between classes.

The problem mentioned in the previous paragraph can be partially solved by the *Borderline*-SMOTE algorithm. It divides samples into three groups (Fig. 5).

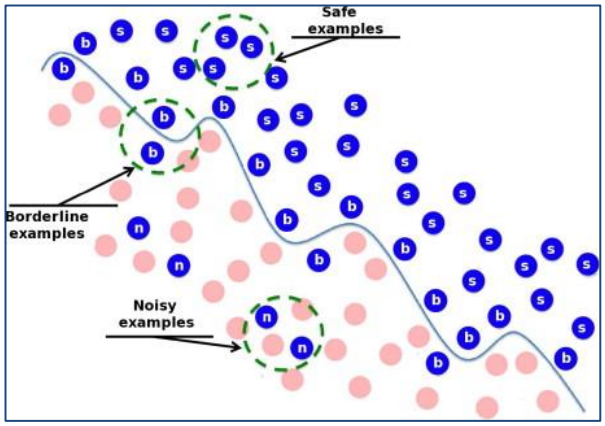

Fig. 5. Types of samples *Borderline*-SMOTE [11]

These groups are:

– *safe* samples – placed in relatively homogeneous areas;

– *noisy* samples from one class present in safe areas of another class;

– *borderline* samples are located in the area surrounding the class boundaries, where either the minority and majority classes overlap or the samples are very close to a complex boundary shape.

The *Borderline*-SMOTE algorithm selects a point that is bordered by another class (but not noise) and performs the same actions as a regular SMOTE.

In SVM-SMOTE, the borderline is approximated by support vectors after training the SVM classifier on the training set. Synthetic data is created randomly along the lines connecting each support vector of the minority class with a number of its nearest neighbors. The peculiarity of SMOTE SVM compared to *Borderline*-SMOTE is that it synthesizes more data away from the area of overlapping classes. It focuses more on the areas where the data is separated [12]. Below is a comparison of the SMOTE algorithm types (Fig. 6).

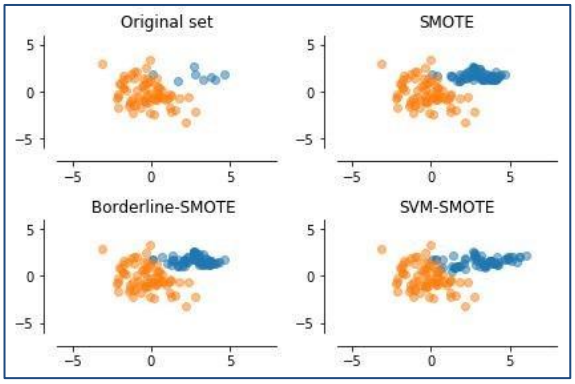

Fig. 6. Distribution of samples after applying SMOTE algorithms

Even from the figure, it can be seen that SVM-SMOTE reinforces samples on all minority class boundaries, unlike *Borderline*-SMOTE, which reinforces class boundaries only on the border with another class.

The *k-means*-SMOTE method uses a simple and popular *k*-means clustering algorithm combined with SMOTE resampling to rebalance the data sets. It manages, unlike conventional SMOTE, to avoid noise generation by resampling instances only in safe areas. Furthermore, its focus is on both inter-class imbalance and intra-class imbalance, dealing with the problem of small disjuncts by expanding small minority areas. SMOTE can generate minority samples in majority areas in the presence of noise. Most noiseless samples are generated in already dense minority areas, which contributes to the intra-class imbalance [13].

The *k*-means algorithm works by iteratively repeating two instructions. First, it assigns each observation to the closest of the *k* cluster centroids. Secondly, it updates the position of the centroids so that they are centered between the observations assigned to them. The algorithm converges when no more observations are assigned. It is guaranteed to converge to a typical local optimum in a finite number of iterations.

ADASYN is similar to SMOTE and its derivative algorithm, but it has an important difference. It shifts the sampling space (i.e., the probability that any particular point will be selected for copying) to points that are not located in homogeneous neighborhoods.

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 2 (24) ISSN 2524-2296 (online)

The main idea of ADASYN is to create an appropriate number of synthetic alternatives for each observation belonging to the minority class. The concept of "appropriate number" depends on how difficult it is to study the original observation. In particular, an observation from a minority class is "hard to learn" if there are many examples from the majority class with features similar to that observation.

Conducting the experiment

Two datasets were chosen for the experiment: "Credit Cards" [14] and "Titanic" [15].

They are different tasks with different types of data, which made it possible to analyze the performance of different oversampling algorithms on different tasks. Some of them were immediately unbalanced, some were not. Therefore, to make the experiment more clear, we decided to modify the number of instances in the original training datasets. For each dataset, four initial modifications were created that had exactly the same imbalance, where the number of samples in the minority class was equal to 15, 30, 45, and 60% of the majority class. Based on the results obtained, comparative tables, graphs, and diagrams were created to show the degree of accuracy achieved by the algorithms after eliminating the imbalance in different ways. For comparison purposes, the models were also trained on the original datasets to understand how much the chosen approaches improve the training efficiency of ML algorithms.

For some datasets, features were normalized. A scaling technique in which values are shifted and scaled so that they ultimately range from 0 to 1 [16]. Below is the normalization formula:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}}, \quad (1)$$

where X_{max} and X_{min} – the maximum and minimum values of the function, respectively.

For many features represented by strings or lists (e.g., class A, B, and C), normalization cannot be applied, so we had to use the *one-hot* encoding algorithm. This is a method of transforming data to prepare it for the algorithm and get a better prediction. With *one-hot*, each categorical value is converted into a new categorical column, and these columns are assigned a binary value of 1 or 0 [17], [18].

The Credit Cards dataset was chosen because it represents an important application problem: companies want to recognize fraudulent credit card transactions so that their customers are not charged for goods they did not purchase. The dataset contains a set of two types of transactions: normal and fraudulent.

This dataset contains transactions that occurred over two days in September 2013 on European credit cards, where 492 frauds occurred out of 284.807 transactions. The dataset is very unbalanced, with the minority class (fraud) accounting for 0.172% of all transactions. It contains only numeric input variables, which are the result of a PCA (principal component analysis) transformation. Due to confidentiality issues, the authors are unable to provide the original features and additional information about the data. The features are represented as V1, V2, ... V28 and are the principal components obtained by PCA. The only features not transformed by PCA are "Time" and "Quantity". The "Time" function contains the seconds elapsed between each transaction and the first transaction in the dataset. The "Quantity" function (the number of transactions) can be used for cost-dependent learning. The "Class" function is a response variable, and it takes the value of 1 in case of fraud and 0 otherwise.

The "Titanic" dataset was chosen first as one of the most famous datasets. It is a dataset of passengers of a ship that was in the center of a catastrophic event. The main task of the models is to predict whether a passenger will survive based on data about their tickets, financial status, and relatives.

The dataset contains the following features:

– survival is a class variable, 0 indicates that the passenger did not survive, and 1 indicates that he or she did;

– pclass – 1, 2, 3 – passenger's ticket class;

– sex – sex of the passenger;

– age – passenger's age;

– sibsp – siblings / spouses (number of brothers, sisters and spouses on board);

– parch – parents / children (number of parents and children on board);

– ticket – ticket number (deleted field as unnecessary information);

– fare – ticket price;

– cabin – cabin number;

– embarked – port of embarkation.

This dataset does not solve any applied problem, but the categories of passengers are the most random, so it was interesting to analyze this dataset.

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2023. No. 2 (24)

To check the quality of the datasets created by oversampling, they were examined and tested by training various ML algorithms. The study used classification algorithms related to supervised machine learning. The selected algorithms were given a training set of features and a test set of training labels known in advance. The models were trained on this data, and during model validation, they received a test dataset without labels.

The following classification models were used:

– logistic regression;
– decision tree;
– support vector machine;
– *k*-nearest neighbors;
– naive Bayesian classifier.

To analyze and compare the results obtained, it is necessary to use certain metrics. The main metric for data analysis is accuracy (Fig. 7).

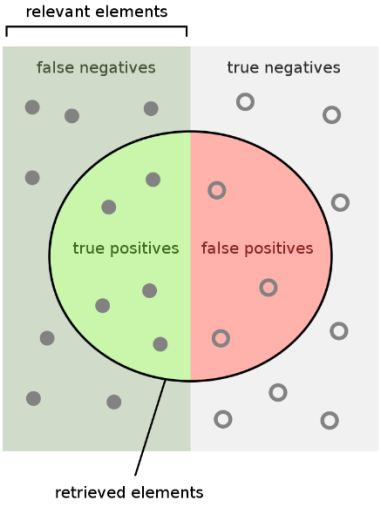

Fig. 7. Schematic representation of metrics

Model accuracy, or *Accuracy*, determines how many correct predictions are made out of those generated by the model. In other words, it can be represented by the following formula:

$$Accuracy = \frac{true\ positive + true\ negative}{true\ positive + true\ negattive + false\ positive + false\ negative}, \quad (2)$$

where *true positive* is the number of correctly predicted items;

true negative is the number of correctly unpredicted items;

false positive is the number of incorrectly predicted items;

false negative is the number of incorrectly unpredicted items.

The study involved training five machine learning models on three datasets. Each set was presented in four modifications (with 15, 30, 45, and 60% imbalance). Each modification was subjected to oversampling, namely six variants. The model was also trained on the unbalanced datasets for further comparison with the experimental datasets. A total of 280 experiments were conducted, the results of which are documented in the form of tables, graphs, and bar charts.

It was decided to present only two tables (one modification of 15% of each dataset) and to include only these results in the report, as they most clearly reflect the experiment. The first one shows the results on the Credit Cards dataset.

Table 1. *Model accuracy (Credit Cards dataset, 15% modification)*

Oversampling algorithm / **Model training algorithm**	**Basic unbalanced dataset**	**ADASYN**	**Borderline SMOTE**	**K-Means SMOTE**	**SMOTE**	**SVM SMOTE**	**Random over-sampler**
Decision Tree Classifier	0.995	0.995	0.995	0.995	0.995	0.995	0.995
K-Neighbors Classifier	0.975	0.990	0.985	0.980	0.990	0.985	0.985
Logistic Regression	0.915	1.000	1.000	0.975	0.975	1.000	0.975
Naive Bayes Classifier	0.990	0.980	0.980	0.995	0.995	0.990	0.990
SVM Classifier	0.975	0.995	0.995	0.990	0.995	0.995	1.000

Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 2 (24)
ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

As you can see from the table, the results are quite different. Some algorithms, such as the *k*-nearest neighbor classifier, logistic regression, and support vector machine, improved their results, while the decision tree showed exactly the same results on all algorithms, and the naive Bayesian classifier generally did a better job on unbalanced data.

The following table shows the results of model training on the "Titanic" dataset (Table 2).

Table 2. *Model accuracy (Titanic dataset, 15% modification)*

Oversampling algorithm / **Model training algorithm**	**Basic unbalanced dataset**	**ADASYN**	**Borderline SMOTE**	**KMeans SMOTE**	**SMOTE**	**SVM SMOTE**	**Random over-sampler**
DecisionTree Classifier	0.849	0.698	0.864	0.849	1.000	0.849	0.834
K-Neighbors Classifier	0.840	0.840	0.837	0.850	0.833	0.861	0.800
Logistic Regression	0.810	0.852	0.867	0.849	0.909	0.972	0.897
Naive Bayes Classifier	0.710	0.812	0.816	0.831	0.816	0.767	0.700
SVM Classifier	0.849	0.879	0.906	0.993	1.000	1.000	0.997

Below are graphs showing the accuracy of the classifiers on datasets with varying degrees of imbalance that have been oversampled, as well as the original dataset, which has not been corrected in any way. Figure 8 demonstrates how the decision tree works on data that has been aligned in different ways. The graph shows the benefits of using oversampling methods compared to the baseline dataset on the "Credit Cards" dataset.

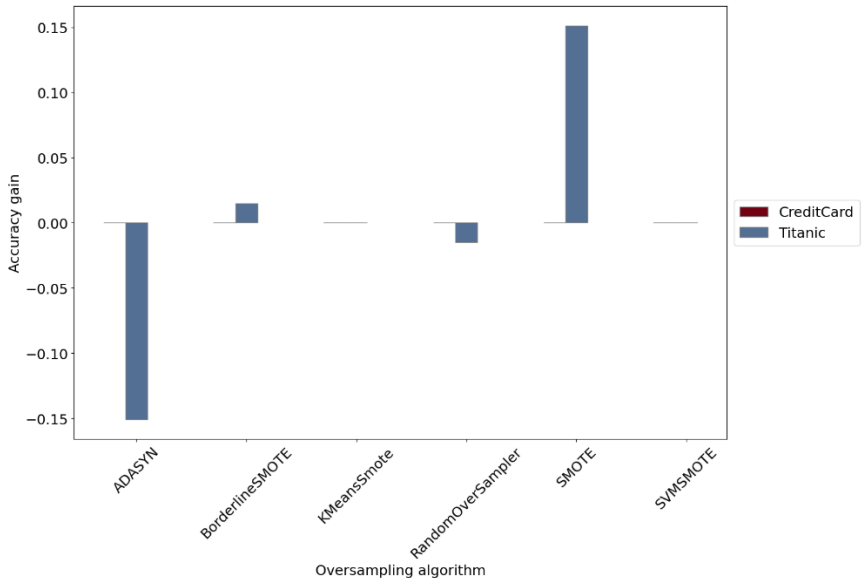

Fig. 8. Advantages of using decision tree algorithms on a 15% dataset

As can be seen from the figure, the Credit Cards dataset shows neither advantages nor disadvantages of using oversampling algorithms. Let's show the same graph for the logistic regression for the sake of representativeness (Fig. 9).

We can see the stable advantages of using each of the oversampling algorithms. The accuracy of logistic regression for the 60% dataset is shown in Fig. 10.

As can be seen in the graph, oversampling algorithms are most effective when there is a strong imbalance. After conducting all the experiments and calculating the average benefit of using each algorithm, we have the results shown in Fig. 11.

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) *Innovative technologies and scientific solutions for industries. 2023. No. 2 (24)*

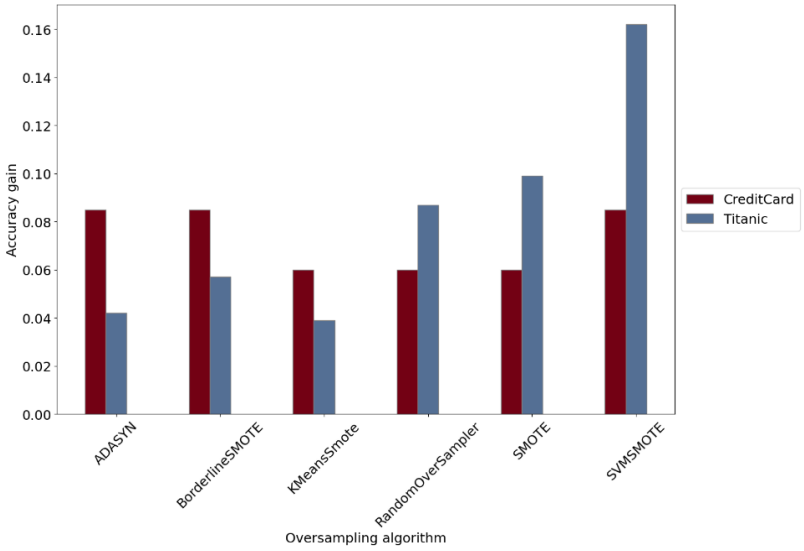

Fig. 9. Advantages of using algorithms with logistic regression on the 15%th dataset

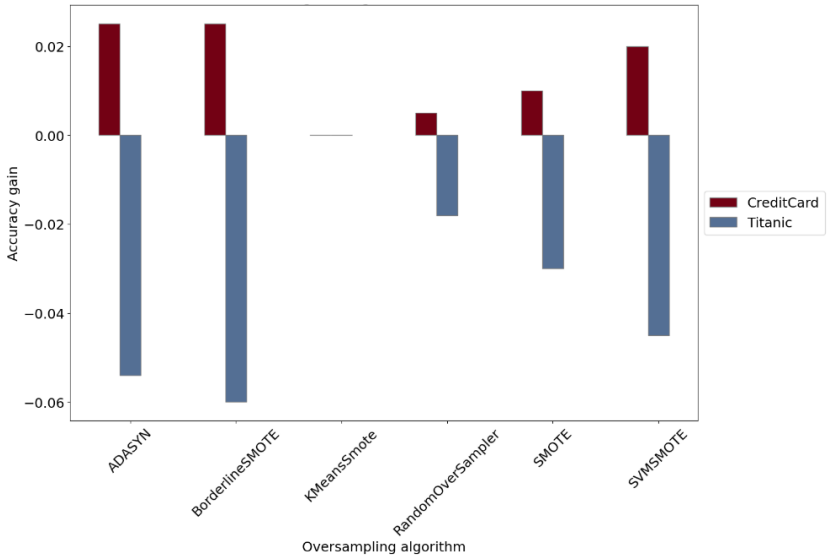

Fig. 10. Advantages of using logistic regression algorithms on 60% of the dataset

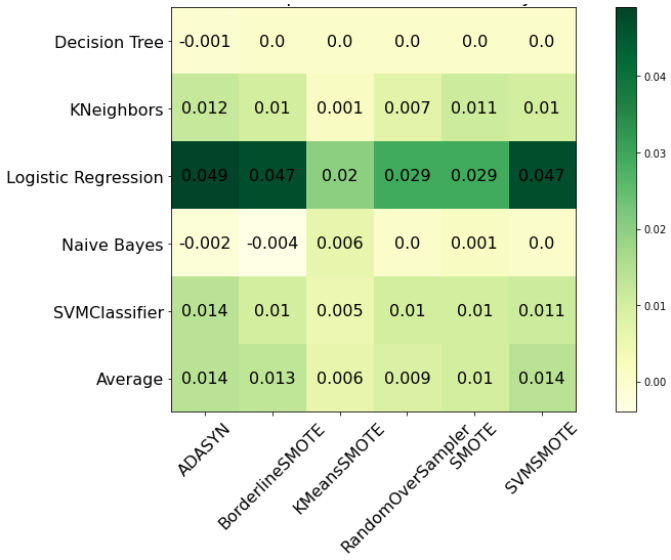

Fig. 11. Heat map of the benefits of using oversampling on the "Credit Cards" dataset

Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 2 (24)
ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

After creating a heat map for the "Titanic" dataset, we have the results shown in Fig. 12.

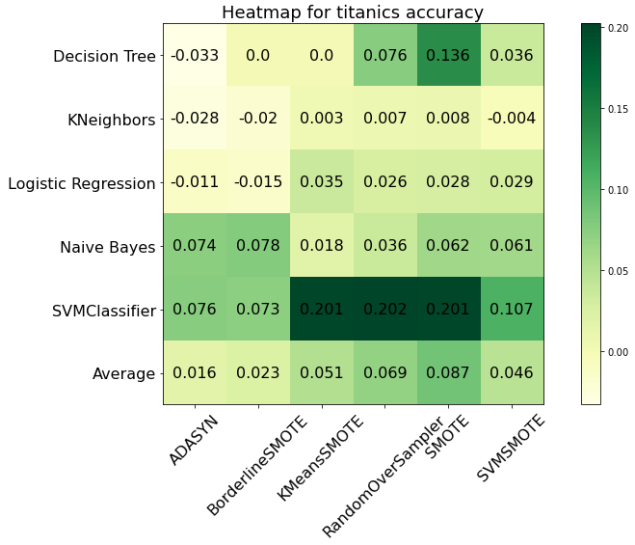

Fig. 12. Heat map of the benefits of using oversampling on the "Titanic" dataset

The bar charts show the absolute increase in accuracy of each model compared to the models trained on the unbalanced datasets.

The decision tree demonstrated an increase only for the "Titanic" dataset, but it was quite noticeable (up to 16%). For all other datasets, there was no difference compared to the original dataset, and for the Salary dataset, ADASYN even worsened the result, but by less than 1%.

The *k-nearest-neighbors* algorithm did not perform as well, and for the "Titanic" dataset, oversampling significantly worsened the results. However, as for the Credit Cards dataset, all balancing algorithms yielded an increase in accuracy of 1–2%. This is very noticeable, given that the accuracy of the models before the modification also had a performance of 90% accuracy or more.

Logistic regression yielded very satisfactory results of 15%. It improved the accuracy of both datasets after they were modified for balancing. SVM-SMOTE improved the accuracy of the model on the Titanic dataset by 16%. For the "Credit Cards" dataset, ADASYN, SVM-SMOTE, and *Borderline*-SMOTE performed well at all degrees of imbalance, and for the dataset with only 15% minority class, the accuracy increased by 8%.

For the naïve Bayesian classifier, only the "Titanic" dataset showed much better results with an increase of 4 to 12%. For the most part, the accuracy has not changed. The only exception is the k-means-SMOTE algorithm, which consistently improved the results of the algorithm on the "Credit Cards" dataset. Although, on the contrary, this algorithm was the only one that began to deteriorate the classifier's performance on the "Titanic" dataset.

For the SVM classifier, *Random Oversampling*, *k-means*-SMOTE, and regular SMOTE showed the greatest stability on the "Titanic" dataset, sometimes providing almost 30% increase in accuracy, which allowed us to obtain a 100% accurate model. For "Credit Cards", only ADASYN had such an accuracy, and SVM-SMOTE, *Borderline*-SMOTE, and regular SMOTE performed quite well, which helped to achieve an accuracy of 99.5–100%.

Bar charts show the absolute increase in accuracy of each model compared to models trained on unbalanced datasets. Analyzing the research, I would like to note that there were two different datasets – "Credit Cards" and "Titanic" – that belonged to different types and behaved differently during the training process.

The "Titanic" dataset is less applicable, and models trained on unbalanced datasets typically yielded 70–90% accuracy. Oversampling algorithms gave a huge increase in accuracy (in some cases up to 100%).

The second dataset was the "Credit Cards" dataset, which showed excellent results. Even with an unbalanced dataset, its accuracy rates reached 90% and higher. As for the balancing algorithms, they performed differently, but in most cases they still improved the results by 1–2%, which is an excellent result given such accuracy.

Conclusions

In the course of the study, the balancing of unbalanced datasets was carried out using various algorithms and applied to solving categorization problems. The paper analyzes the problems of a given domain and methods of balancing unbalanced datasets, and considers and investigates six balancing algorithms: *Random Oversampling*, SMOTE, *Borderline*-SMOTE, *k-means*-SMOTE, SVM-SMOTE, and ADASYN.

All the experiments were conducted on two datasets – "Credit Cards" and "Titanic", which demonstrated different performance. The results of the study are presented in the form of graphs, tables, and bar charts representing the accuracy indicators.

The best results were obtained for the "Credit cards" dataset. The data was well selected, the accuracy of the models increased after using almost all algorithm

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) *Innovative technologies and scientific solutions for industries. 2023. No. 2 (24)*

variants, the completeness increased significantly, but most importantly, the accuracy of the minority class improved. This indicates that it was possible to get rid of the problem without losing the accuracy of the model as a whole.

For the "Titanic" dataset, where there is no clear dependency (unlike "Credit Cards"), many of the algorithms show a positive trend, and the best RO accuracy is achieved by SMOTE and *k-means*-SMOTE on the SVM classifier model.

The obtained results can be applied if it is necessary to use an unbalanced dataset or for further research in the field of machine learning.

References

1. Mary, A. J., Claret, A. (2021), "Imbalanced Classification Problems: Systematic Study and Challenges in Healthcare Insurance Fraud Detection", *5th International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, P. 1049–1055. DOI: 10.1109/ICOEI51242.2021.9452828

2. Srinilta, C., Kanharattanachai, S. (2021), "Application of Natural Neighbor-based Algorithm on Oversampling SMOTE Algorithms", *7th International Conference on Engineering, Applied Sciences and Technology (ICEAST)*, Pattaya, Thailand, P. 217–220. DOI: 10.1109/ICEAST52143.2021.9426310

3. Das, R., Biswas, S. K., Devi, D., Sarma, B. (2020), "An Oversampling Technique by Integrating Reverse Nearest Neighbor in SMOTE: Reverse-SMOTE," *International Conference on Smart Electronics and Communication (ICOSEC)*, Trichy, India, P. 1239–1244. DOI: 10.1109/ICOSEC49089.2020.9215387

4. Feng, L. (2022), "Research on Customer Churn Intelligent Prediction Model based on Borderline-SMOTE and Random Forest," *IEEE 4th International Conference on Power, Intelligent Computing and Systems (ICPICS)*, Shenyang, China, P. 803–807. DOI: 10.1109/ICPICS55264.2022.9873702

5. Dudjak, M., Martinović, G. (2021), "An empirical study of data intrinsic characteristics that make learning from imbalanced data difficult", *Expert Systems with Applications*, Vol. 182, DOI: https://doi.org/10.1016/j.eswa.2021.115297

6. Liu, C., Jin, S., Wang, D. (2020), "Constrained Oversampling: An Oversampling Approach to Reduce Noise Generation in Imbalanced Datasets with Class Overlapping," *IEEE Access*, Vol. 10, P. 91452–91465. DOI: 10.1109/ACCESS.2020.3018911

7. Ali, H., Mohd Salleh, M., Saedudin, R., Hussain, K., Mushtaq, M. (2019). "Imbalance class problems in data mining: a review", *Indonesian Journal of Electrical Engineering and Computer Science*, No. 14(3), 1552. DOI: 10.11591/ijeecs.v14.i3.pp1552-1563

8. Medium (2022), "Undersampling and oversampling: An old and a new approach", available at: https://medium.com/analytics-vidhya/undersampling-and-oversampling-an-old-and-a-new-approach-4f984a0e8392 (last accessed: 10.05.2023)

9. Sandeep Kini, M., Devidas, Smitha, N. Pai, Sucheta Kolekar, Vasudeva Pai, Balasubramani, R. (2022), "Use of Machine Learning and Random OverSampling in Stroke Prediction", *International Conference on Artificial Intelligence and Data Engineering (AIDE)*, Karkala, India, P. 331–337. DOI: 10.1109/AIDE57180.2022.10060313

10. Blagus, R., Lusa, L. (2012), "Evaluation of SMOTE for High-Dimensional Class-Imbalanced Microarray Data", *11th International Conference on Machine Learning and Applications*, Boca Raton, FL, USA, P. 89–94. DOI: 10.1109/ICMLA.2012.183

11. Sáez, J., Luengo, J., Stefanowski, J., Herrera, F. (2015), "SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering", *Information Sciences*, Vol. 291, P. 184–203. DOI: https://doi.org/10.1016/j.ins.2014.08.051

12. Mahalakshmi, M., Ramkumar, M. P., Emil Selvan, G., S., R. (2022), "SCADA Intrusion Detection System using Cost Sensitive Machine Learning and SMOTE-SVM", *4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, Greater Noida, India, P. 332–337. DOI: 10.1109/ICAC3N56670.2022.10074251

13. Puri, A., Gupta, M. (2020), "Improved Hybrid Bag-Boost Ensemble With K-Means-SMOTE–ENN Technique for Handling Noisy Class Imbalanced Data", *The Computer Journal*, Oxford University Press, Vol. 65, No. 1, P. 124–138. DOI: 10.1093/comjnl/bxab039

14. "Titanic–Machine Learning from Disaster", (2022), available at: https://www.kaggle.com/competitions/titanic/data?select=gender_submission.csv (last accessed: 10.05.2023)

15. "Salary Prediction Classification", (2022), available at: https://www.kaggle.com/datasets/ayessa/salary-prediction-classification (last accessed: 10.05.2023)

16. Ni, N., Wu, H., Zhang, L. (2022), "Deformable Alignment and Scale-Adaptive Feature Extraction Network for Continuous-Scale Satellite Video Super-Resolution," *IEEE International Conference on Image Processing (ICIP)*, Bordeaux, France, P. 2746–2750. DOI: 10.1109/ICIP46576.2022.9897998

17. Yu, L., Zhou, R., Chen, R., Lai, K. K. (2020), "Missing data preprocessing in credit classification: One-hot encoding or imputation?" *Emerging Markets Finance and Trade*, Vol. 58, No. 2, P. 472–482. DOI: 10.1080/1540496X.2020.1825935

18. Dahouda, M. K., Joe, I. (2021), "A Deep-Learned Embedding Technique for Categorical Features Encoding", *IEEE Access*, Vol. 9, P. 114381–114391. DOI: 10.1109/ACCESS.2021.3104357

Received 25.05.2023

Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 2 (24)
ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

Відомості про авторів / About the Authors

Тесленко Денис Максимович – Харківський національний університет радіоелектроніки, магістр зі спеціальності "Інженерія програмного забезпечення", Харків, Україна; e-mail: denys.teslenko@nure.ua; ORCID ID: https://orcid.org/0000-0002-6289-1633

Сорокіна Анна Сергіївна – Харківський національний університет радіоелектроніки, магістр зі спеціальності "Інженерія програмного забезпечення", Харків, Україна; e-mail: anna.sorokina@nure.ua; ORCID ID: https://orcid.org/0009-0006-7380-5223

Ховрат Артем Вячеславович – Харківський національний університет радіоелектроніки, магістр зі спеціальності "Інженерія програмного забезпечення", Харків, Україна; e-mail: artem.khovrat@gmail.com; ORCID ID: https://orcid.org/0000-0002-1753-8929

Гулієв Нурал Бахадур огли – Харківський національний університет радіоелектроніки, магістр зі спеціальності "Інженерія програмного забезпечення", Харків, Україна; e-mail: nural.huliiev@nure.ua; ORCID ID: https://orcid.org/0000-0003-2123-0377

Кирій Валентина Василівна – кандидат економічних наук, доцент, Харківський національний університет радіоелектроніки, доцент кафедри економічної кібернетики та управління економічною безпекою, доцент кафедри програмної інженерії (за сумісництвом), Харків, Україна; e-mail: valentyna.kyriy@nure.ua; ORCID ID: https://orcid.org/0000-0002-2537-264X

Teslenko Denys – Kharkiv National University of Radio Electronics, M. Sc. in Software Engineering, Kharkiv, Ukraine.

Sorokina Anna – Kharkiv National University of Radio Electronics, M. Sc. in Software Engineering, Kharkiv, Ukraine.

Khovrat Artem – Kharkiv National University of Radio Electronics, M. Sc. in Software Engineering, Kharkiv, Ukraine.

Huliiev Nural – Kharkiv National University of Radio Electronics, M. Sc. in Software Engineering, Kharkiv, Ukraine.

Kyriy Valentyna – PhD (Economic Sciences), Associate Professor, Kharkiv National University of Radio Electronics, Associate Professor at the Department of Economic Cybernetics and Management of Economic Security Department, Associate Professor at the Department of Software Engineering (part time), Kharkiv, Ukraine.

ПОРІВНЯННЯ АЛГОРИТМІВ ОВЕРСЕМПЛІНГУ НАБОРІВ ДАНИХ ТА ЇХ ЗАСТОСОВНОСТІ ДЛЯ ПРОБЛЕМИ КАТЕГОРИЗАЦІЇ

Предметом дослідження є питання класифікації в машинному навчанні за наявності незбалансованості класів у наборах даних. **Мета роботи** – аналіз наявних рішень і алгоритмів розв'язання проблеми незбалансованості в наборах даних різних типів і різних галузей та експериментальне порівняння алгоритмів. У статті виконуються такі **завдання**: аналіз підходів до вирішення проблеми – методи попереднього оброблення, методи навчання, гібридні методи й алгоритмічні підходи; визначення та опис алгоритмів оверсемплінгу, що найчастіше використовуються для балансування наборів даних; вибір алгоритмів класифікації, які будуть слугувати інструментом установлення якості балансування, перевіряючи застосовність отриманих після оверсемплінгу наборів даних; визначення метрик оцінки якості класифікації для порівняння; проведення експериментів за запропонованою методикою для виокремлення оптимальних і неоптимальних алгоритмів. Для наочності розглядалися набори даних із різним ступенем незбалансованості (кількість екземплярів класу меншості дорівнювала 15, 30, 45 та 60% від кількості зразків класу більшості). Використовуються такі **методи**: аналітичний та індуктивний – з метою визначення необхідного набору експериментів і побудови гіпотез щодо їх результатів; експериментальний та графічний – для наочної порівняльної характеристики обраних алгоритмів. Здобуто такі **результати**: за допомогою метрик якості досліджено всі алгоритми на двох різних датасетах – пасажирів "Титаніку" та з виявлення шахрайських транзакцій у банківських рахунках; доведено найкращу застосовність алгоритмів SMOTE та SVM SMOTE і виявлено найгірші показники у *Borderline*-SMOTE та *k-means*-SMOTE; описано результати кожного з алгоритмів і потенціал їх використання. **Висновки**. Застосування аналітичного та експериментального методу надало вичерпну порівняльну характеристику алгоритмів балансування. Доведено перевагу алгоритмів оверсемплінгу над алгоритмами андерсемплінгу. Вони порівнювалися за допомогою різних алгоритмів класифікації. Результати подано в графіках і таблицях, а також продемонстровано з допомогою теплових карт. Сформульовано висновки, що можуть бути використані у виборі оптимального алгоритму балансування у сфері машинного навчання.

Ключові слова: категоризація; машинне навчання; методи балансування; методи генерації даних; набір даних; незбалансовані набори даних.

Бібліографічні описи / Bibliographic descriptions

Тесленко Д. М., Сорокіна А. С., Ховрат А. В., Гулієв Н. Б. огли, Кирій В. В. Порівняння алгоритмів оверсемплінгу наборів даних та їх застосовності для проблеми категоризації. *Сучасний стан наукових досліджень та технологій в промисловості*. 2023. № 2 (24). С. 161–171. DOI: https://doi.org/10.30837/ITSSI.2023.24.161

Teslenko, D., Sorokina, A., Khovrat, A., Huliiev, N., Kyriy, V. (2023), "Comparison of dataset oversampling algorithms and their applicability to the categorization problem", *Innovative Technologies and Scientific Solutions for Industries*, No. 2 (24), P. 161–171. DOI: https://doi.org/10.30837/ITSSI.2023.24.161