A benchmark analysis of methods applied to tackle severe class imbalance for predicting converting sessions of an e-commerce business

Master's thesis

for acquiring the degree of Master of Science (M.Sc.)
in Economics and Management Science

at the School of Business and Economics of Humboldt Universität zu Berlin

Submitted by

Berkay Kocak

Student no: 614468

Study Program: Master of Economics and Management Science

Address, Phone: Sewanstrasse 177, 10319 Berlin. +49 1575 221 78 54

E-mail: kocakber@student.hu-berlin.de

Supervisor and Examiner: Prof. Dr. Stefan Lessmann

Chair of Information Systems

Second Examiner: Prof. Dr. Benjamin Fabian

Berlin, 24.11.2022

Table of Contents

Abstract - This study aims to predict the converting sessions of an e-commerce company in the automotive sector. The main challenge that this study addresses is the severe class imbalance in an e-commerce setting. The methods that are applied are of two families. Data-level methods include oversampling, undersampling, and hybrid sampling. Algorithm-level methods compose of cost-sensitive learning and thresholding. A comparative analysis is conducted to determine the best-performing method. Further thresholding is applied to optimize the confusion matrix and related scores.

Keywords— ***Big Data; Classification; Data Imbalance; Sampling; E-Commerce; Cost Sensitive Learning***

I) Introduction

The volume of data available for business has increased substantially in the past decades. The increase in the volume, variety, and velocity of the data enables the companies to track current performance efficiently and predict future performance. This enhancement in the data combined with the technologies also generates a new variety of applications in the realm of artificial intelligence. A large proportion of businesses, ranging from start-ups to big corporations, currently deploy machine learning and deep learning models for accurate predictions which in turn decrease costs and increase profits.

The increase in machine learning applications generates significant business value. However, it also accommodates some challenges to tackle to enhance the full potential. In the scope of this study, one of these challenges, namely 'Data Imbalance' in a binary classification setting, is addressed. Imbalanced data classification is a problem in which proportional class sizes of a dataset differ relatively by a substantial margin (Kaur et al., 2019).

The data imbalance problem is previously analyzed in several papers in different industries. The main categories for the applied papers may lie in fields such as financial management, medical diagnosis, and telecommunications (Haixiang et al., 2017). The popular application areas include churn prediction (Burez & Van den Poel, 2009) and disease prediction (Khalilia et al., 2011).

Though huge data are streaming constantly during online commerce, the data imbalance problem is still unaddressed due to insufficient analytical algorithms to handle huge datasets for smooth outliers (Dhote et al., 2020). This paper aims to predict the converting sessions for

a company operating in the automotive e-commerce industry in the presence of a highly imbalanced dataset.

There are various methods to tackle the class imbalance problem. However, there is no specific algorithm that is superior in all benchmark tests (Haixiang et al., 2017). Therefore, a benchmark analysis is conducted to discern the best method in this case.

The remaining sections of the paper are structured as follows. In Section II, the classification of imbalanced datasets is discussed in detail. In addition, the factors that turn class imbalance into a challenge are clarified. Section III reviews the methods against class imbalance and relates the characteristics of the problem to the techniques of the learners. Section IV examines the algorithms that are combined with the methods against imbalanced data. The experimental setting in which the models are trained is discussed in Section V and the comparative analysis is conducted in Section VI. Section VII analyzes the best-performing model further and attempts to improve the performance via thresholding. Section VIII mentions the limitations that are encountered and the future work while Section IX concludes the paper. The GitHub repository for the related work can be found in Section X.

II) Classification in a Highly Imbalanced Dataset

Binary classification is an instance of supervised learning where the outcome variable has two possible outcomes. These two outcomes are named majority class and minority class in the presence of data imbalance. The majority class dominates the minority class in terms of the number of observations seen in the dataset. In the scope of this study, the converting sessions constitute the minority class whereas the non-converting sessions are the majority class.

Li et al. (2016) discuss that when the percentage of minority class is reduced to 5%, it will be problematic to obtain a good anticipating model because of the small amount of information to understand about the rare event. However, whether the class imbalance is problematic depends on several factors (Huang & Dai, 2021):

- The scale of the overlapping space, which refers to the feature that different classes of samples have no clear boundary in the sample space. Denil and Trappenberg (2010) discuss the overlapping problem in an imbalanced setting.

- The number of noise samples, which refers to a few examples of one class far away from the core area of the class (López et al., 2013).
- The number of training samples, which refers to the model training samples (Yu et al., 2016).
- The degree of interclass aggregation, which refers to the feature that one class samples present two or more clusters in the sample space and these clusters can be distinguished as major and minor (Japkowicz et al., 2002).
- The dimension of the dataset, which refers to the number of features.

Although data mining approaches have been widely used to build classification models to guide commercial and managerial decision-making, classifying imbalanced data significantly challenges these traditional classification models (Haixiang et al., 2017).

The class imbalance is substantially more problematic in our dataset due to several reasons. The data is severely imbalanced where the minority class makes up only 0.4% of the sessions on the website. The scale of overlapping space is high for the observations. The relative imbalance and the absolute imbalance are severe due to the lack of observations from the minority class.

The following attempts are conducted to balance the dataset.

- The sessions that spent less than 30 seconds (bounced sessions) are disregarded.
- The sessions created by bots are disregarded.
- The sessions where the platform is not the website are disregarded.

The company owns a mobile app as a second platform. However, the tracking of the performance is not yet fully adapted. Therefore, it is problematic to derive the same features as on the website which is the reason for the elimination of this platform.

These steps have reduced the data to 300,000 observations. However, the minority class constitutes only around 1200 observations in the sample. Most classifiers tend to implicitly consider their data as balanced, hence standard classifiers are biased toward the majority (Somasundaram & Redi, 2016). Therefore, this paper employs different algorithms and methodologies to tackle the problem of class imbalance. These methodologies are discussed in detail in Section III.

III) Methods against Class Imbalance

Class imbalance is a very common problem encountered in machine learning applications. Most classifiers like decision trees and neural networks function effectively when the distribution of the response variable is balanced in the dataset (Li et al., 2016). However, when the imbalance ratio in the dataset increases, it is a harder task for algorithms to learn from the minority class due to the lack of data. There have been several studies that focus on the methods to tackle class imbalance in various ways. The variety of the studies in the literature gives rise to different methods and distinct aggregation of these methods into different families.

Wang et al. (2021) analyze the classification methods for imbalanced data in detail and summarizes them from the aspects of data sampling, algorithm-level, feature-level, and deep learning methods.

Park et al. (2015) classify the methods into three different subcategories.

1) Algorithm-level approach: the existing classifier becomes more biased toward the class of interest
2) Data-level approach: it alleviates the size difference between the classes
3) Cost-sensitive learning approach: it gives more weight to the minority class to impose a higher misclassification cost for that class.

Somasundaram and Reddy (2016) argue that the data imbalance can be handled via two different methods which are modifying the existing algorithms to provide increased weightage to the minority classes or resampling. The paper focuses on the sampling methodologies in highly imbalanced datasets where the imbalance ratio is ranging from 3274:1 to more balanced settings.

Patel et al. (2020) suggest four different approaches to tackle class imbalance. The first approach applies data pre-processing/re-sampling on data. Therefore, the method is also named as ‘‘Data-Level Solution’’. The method alters the data rather than the classifiers. The second strategy takes the natural distribution of data as input. However, classifiers are modified to target class imbalance. The approach is named as ‘‘Algorithmic-Level Solution’’ due to changes in classification algorithms. Other known strategies are ‘‘Cost-Sensitive Approaches and Ensemble Techniques’’. Finally, the imbalance is also dealt with “Feature Selection and Evolutionary Approaches”.

Kaur et al. (2019) similarly propose three broad approaches to address imbalance issues in data classification which are pre-processing methods, algorithm-centered approaches, and hybrid approaches. Longadge and Dongre (2013) propose that applying two or more techniques i.e. hybrid approaches such as SMOTEBoost and RUSBoost yields better results for the class imbalance problem.

The hybrid method combines multiple classification techniques from the above-mentioned categories. Several methods such as Balance Cascade and Easy Ensemble are used to train a group of classifiers on undersampled subgroups (Kumar et al., 2021). Pre-training and fine-tuning on the original imbalanced dataset are included in these methods.

Haixiang et al. (2017) address imbalanced learning via two basic strategies which are preprocessing and cost-sensitive learning. Preprocessing approaches enclose two different spaces. In the sample space, resampling methods are analyzed while in the feature space, feature selection methods are investigated.

Huang and Dai (2021) discuss the key methods to handle class imbalanced problems from data-driven and algorithm-driven perspectives. Data-driven methods include undersampling, oversampling, and hybrid sampling. On the other hand, algorithm-driven approaches contain Cost-Sensitive Learning, Active Learning, Decision Adjustment Learning, Feature Extraction Learning, and Ensemble Learning.

López et al. (2012) conduct an analysis to evaluate the performances of data sampling and cost-sensitive approaches for learning in the presence of imbalanced data. After experiments, they come to the result that, in general, both strategies yield well results for class imbalance and to determine the best among both, further data intrinsic characteristic analysis is needed.

Buda et al. (2018) mention that algorithm-level methods include hybrid methods like ensemble and classical methods like thresholding, one-class classification, and cost-sensitive learning.

Kumar et al. (2021) also propose one-class classification method which observes objects from the majority class rather than learning from both classes. An identity function is trained to implement associative mapping. The new example in the test set is classified based on the absolute errors and the sum of squared errors.

Tsai and Liu (2021) analyze one class-classifiers against imbalanced data. They underline that in the future, the ensemble-based class imbalanced learning methods will be one of the main research directions. However, ensemble learning embodies the disadvantages of long training time and high computational complexity.

This paper addresses the problem of class imbalance via two main methods; changing the data (Somasundaram & Reddy, 2016; Kubat & Matwin, 1997; Japkowicz, 2000) and changing the algorithm (Longadge & Dongre, 2013; Kumar et al., 2021; Buda et al., 2018; Huang & Dai; 2021).

As mentioned earlier, the data-driven approach aims to balance the class distribution whereas the algorithm-driven approach attempts to redress the learning algorithm or classifier without altering the training set. Data-driven methods in this paper include undersampling, oversampling, and hybrid sampling. The algorithm-driven approaches involve cost-sensitive learning and thresholding. The methods along with their advantages and disadvantages are shown in Table 1.

One-class classification is not analyzed in this paper since the data set has many observations. Therefore, the minority class also offers the algorithms a learning space. Disregarding this space yields a worse performance as the initial experiments on the data show.

The approaches such as feature selection and evaluation metrics are not analyzed as separate methods against class imbalance. They are rather seen as fundamental steps of any project in this field. Section V-a explains the evaluation metrics that are chosen in detail.

Several methods propose ensemble learners as a solution to the class imbalance problem. Therefore, hybrid approaches usually contain a method analyzed in this paper in combination with an ensemble learner. This paper analyzes the learners in a separate manner in Section IV.

Although there are various papers in the literature that seek to mitigate the problem of class imbalance, the number of papers that analyze severe imbalance with IR less than 1:100 is very few. There is also a significant research gap in terms of e-commerce. Therefore, this paper focuses on solving the class imbalance problem in a highly imbalanced e-commerce setting.

	Approach	Advantages	Disadvantages
Data-Level	Undersampling	• Needs lower computational power and therefore faster	• Might lose informative instances from the majority class
	Oversampling	• No loss of informative samples	• Adds artificial data to the sample • Needs higher computational power
	Hybrid Sampling	• Balance between losing informative samples and adding artificial data	• Adds artificial data to the sample • Might still lose informative instances
Algorithm-Level	Cost Sensitive Learning	• Assigns costs to the misclassification and therefore better for business cases	• The costs need to be determined a priori

Table 1: Advantages and Disadvantages of the methods that are applied

a) Data-Level Methods

This section focuses on the data-level methods. These methods concentrate on changing the ratio between the majority and the minority class via adding new artificial data or removing observations from the dataset. These approaches are also frequently named as resampling methods in the literature. Resampling can be accomplished either by oversampling the minority class or undersampling the majority class (Anand et al., 2010), or a combination of these two methods.

Abouelenien et al. (2013) argue that pre-processing stage methods involve several re-sampling methods, such as oversampling and undersampling or consolidating the two sampling methods as a motive to obtain an approximately equal count of samples in the classes.

In this paper, three major resampling approaches, namely undersampling, oversampling and hybrid sampling are analyzed. Resampling methods are more versatile because they are independent of the selected classifier (López et al., 2013). Therefore, they are suitable to apply in many different settings.

While oversampling introduces new artificial data to the dataset, undersampling leads to information loss. Akbani et al. (2004) propose that oversampling should be avoided in biological data since certain instances are biologically not possible to exist although the algorithm might artificially create them.

García et al. (2012) analyze the effect of imbalance ratio and classifier properties on a variety of resampling approaches. They conclude through experiments that the performance of oversampling and undersampling approaches are equivalent in the presence of a low imbalance

ratio. However, for highly imbalanced datasets, oversampling should be preferred for better classification.

On the other hand, oversampling tends to increase the size of the data space, which results in a situation of overfitting and takes more time in the training phase (Kaur et al., 2019). However, undersampling removes some samples randomly from the majority class which creates the possibility of losing informative samples (Kaur et al., 2019).

This paper applies five different resampling approaches to the problem. Due to the high imbalance in the dataset, oversampling methods are favored. One undersampling (RUS), three oversampling (SMOTE, Borderline SMOTE, ADASYN), and one hybrid sampling (SMOTE + RUS) approach are picked against class imbalance. The methods that are applied and the reasoning behind the selection of the methods are described in detail in the following sections.

1) Undersampling

Undersampling is the first resampling method analyzed in this paper. It has several advantages over other resampling methods. Any methodology can be trained faster and with significantly less memory usage due to decreased sample size. However, this comes at the cost of losing informative instances from the majority class. There have been several studies that deploy a variety of undersampling methods against imbalanced datasets.

Kubat and Matwin (1997) propose geometric mean-based undersampling against class imbalance and Kubat et al. (1998) deploy the SHRINK system to detect oil spills in an imbalanced dataset. Kumar et al. (2021) put forward undersampling techniques like Condensed Nearest Neighbour, Edited Nearest Neighbour, Neighbourhood Cleaning, Tomek Links, and One-sided selection as possible solutions.

Khalilia et al. (2011) employ an ensemble learning approach. The method is based on repeated random sub-sampling. This technique divides the training data into multiple sub-samples while ensuring that each sub-sample is fully balanced. The study aims to predict disease risks from a highly imbalanced dataset.

Park et al. (2015) propose Decision Boundary Focused Undersampling (DBFUS). This method performs undersampling to a highly imbalanced dataset (1:33) in e-commerce. DBFUS considers the distance to the decision boundary so that more samples are taken near the decision boundary when performing undersampling from the majority class.

Somasundaram and Reddy (2016) observe from the experiments that undersampling performs better than oversampling in a highly imbalanced setting. Therefore, they prefer the undersampling component as the proposed framework in their study. However, KDD Cup 1999 Dataset analyzed in this paper is a simulated dataset. Therefore, the results can significantly differ in a business setting with real data.

Seiffert et al. (2009) present a hybrid approach to solve the data imbalance problem which is called RUSBoost. It combines random undersampling with the boosting framework.

The simplest yet most effective method is random undersampling (RUS), which involves the random elimination of majority class examples (Tahir et al., 2009). Galar et al., (2011) conduct a comparative analysis of various ensemble learning algorithms and conclude that the RUSBoost algorithm has good performance and least complex among all.

Several different studies in the literature propose random undersampling as a method with good performance and less complexity. Therefore, this study utilizes random undersampling. The initial results that are discussed in Section VI show that the performance of oversampling is significantly higher for our case compared to undersampling. Therefore, no further frameworks of this family are deployed to tackle the problem.

2) Oversampling

The second resampling approach to tackle class imbalance, namely oversampling, increases the sample size by adding artificial observations of the minority class to the dataset to balance the data. This method does not lead to any information loss in the dataset. However, it needs higher memory and longer runtime to provide results. Similar to undersampling, there are a large variety of different oversampling methodologies in the literature.

Chawla et al. (2002) propose Synthetic Minority Oversampling Technique (SMOTE) to tackle the class imbalance problem. They use the feature-based similarity to generate synthetic examples of the minority examples based on the k-nearest neighbor. This approach enhances the decision boundary of the traditional classifier close to minority examples.

SMOTE is applied to a large variety of datasets to counter class imbalance (Wu & Meng, 2016; Wang et al., 2021; Haixiang et al., 2017; Kaur et al., 2019; Huang et al., 2021, Kumar et al., 2021). Ramyachitra and Manikandan (2014) even mention SMOTE as the most

popular approach. Haixiang et al. (2017) mention that SMOTE is slightly more effective in recognizing outliers among different resampling methods.

Han et al. (2005) develop Borderline-SMOTE which is an improved version of SMOTE. Borderline-SMOTE divides the instances into three zones that are defined by specific numerical ranges. The method makes use of the number of negative instances in the K-nearest domains to determine the noise, bounds, and safety. Borderline-SMOTE uses the same oversampling technique as SMOTE, but it only oversamples border instances of the minority class instead of oversampling all instances of the minority class (Huang & Dai, 2021). Therefore, when the dataset has a few noise samples, the method's classification result is better than SMOTE (Wang et al., 2021). Borderline SMOTE is also better suited to two-class problems than SMOTE (Wang et al., 2021).

Zhang and Li (2014) propose a random walk oversampling approach to imbalanced data classification. They employ three traditional classifiers and observe the effect of oversampling on them. The study concludes that oversampling significantly influences the performances of classifiers and, thus, is helpful in the classification of imbalanced datasets.

He et al. (2008) propose the ADASYN approach using the weighted distribution of different minority class examples based on their difficulty level of learning. If a specific instance is harder to learn by the algorithm, more synthetic observations are created according to this specific instance compared to an easier observation from the minority class. Therefore, this approach reduces the biasness by shifting the decision boundary aiming at the difficult minority instances (Patel et al., 2020).

ADASYN proves to be an efficient way of handling imbalanced data in the following ways (Kaur et al., 2019):

(1) It tends to reduce the situation of imbalance data where the hyperplane always gets biased toward the majority class

(2) Generates the classification hyperplane so efficiently that it automatically leans in the direction of instances that are difficult to learn

Based on the literature review, this study deploys three different oversampling techniques to the problem at hand which are namely SMOTE, Borderline SMOTE, and ADASYN. The results of the methods are discussed in detail in Section VI.

3) Hybrid Sampling

The final resampling approach analyzed in the scope of this study is the hybrid sampling method. In this method, oversampling is applied to the minority class while the majority class is undersampled. Therefore, this technique offers a fine balance between losing informative instances and needing a higher computational power. In other words, the proposed method reduces the chances of losing informative instances while it creates fewer data points synthetically (Kaur et al., 2019).

Ling and Li (1998) propose a hybrid sampling approach in which they use the lift curve as the analysis criteria. Solberg (1996) proposes another hybrid method for oil slick classification. Identifying the effect of imbalance in a dataset based on undersampling, resampling, and recognition-based induction scheme is proposed by Japkowicz (2000).

Qian et al. (2014) also deploy hybrid sampling for class imbalance. The ratio of minority class and majority class instances determines the scale of re-sampling. The experimental results show that the imbalance ratio and the number of features in the dataset correlate with the performance of the algorithm.

Park et al. (2015) mention that a hybrid of undersampling and oversampling is left for future work in their paper. Haixiang et al. (2017) suggest a combination of SMOTE and undersampling as an alternative when the training sample size is too large. The experiments that Wang et al. (2021) conducted in their study show that oversampling and undersampling are not the best methods to use but combining them might be useful.

Therefore, this study also includes one hybrid sampling method. Random undersampling and SMOTE are combined to tackle the class imbalance problem in the dataset.

b) Algorithm-Level methods

The algorithm-centered approach includes assumptions created to favor the minority class and changing the costs to get the balance classes (Kaur et al., 2019). Huang and Dai (2021*)* similarly argue that the algorithm-level methods that are improving standard classifiers are mainly based on cost-sensitive learning and threshold moving.

Other approaches that are offered in the literature include one-class classification and hybrid methods. One class classification is not deemed as a proper method in our case due to the large number of observations in the dataset. Thus, there is a notable number of instances of the minority class to learn from as mentioned earlier. In addition, the initial results also prove that these methods are not well suited to the problem at hand.

Hybrid methods are also mentioned in a variety of papers. These methods mainly make use of an ensemble learner and resampling method or an ensemble learner and cost-sensitive learning combination. Although this paper also employs cost-sensitive learning and resampling methods with ensemble learners, these are not observed under the view of hybrid methods. Instead, combined algorithms with these methodologies are explained in detail in Section IV.

1) Cost-Sensitive Learning

Cost-sensitive learning modifies the learning rate to increase the contribution of higher-cost instances when updating the weightage. Then we can train by reducing the misclassification cost instead of standard loss function equivalent to oversampling (Kumar et al., 2021).

Haixiang et al. (2017) state that when classifying big data streams, cost-sensitive learning is computationally more efficient than data sampling techniques. Thus, it is more suitable to use.

In addition, the prediction in the management field in comparison to other fields is often driven by profit rather than accuracy. Therefore, cost-sensitive learning is often utilized and the cost of misclassification can be decided by experts or managers (Haixiang et al., 2017).

Cost-sensitive learning (Pazzani et al., 1994) is one of the frequently used technologies to solve the problem of class imbalance and its goal is to minimize the cost of overall misclassification. Huang and Dai (2021) underline that the design of the cost matrix is very important for classification and propose the following design types:

- Empirical weighted design, in which the cost coefficients of the samples of the same class are the same (Zong et al., 2013).

- Fuzzy weighted design, in which the cost coefficients of the same class are different in different positions of sample spaces (Dai, 2015).

- Adaptive weighted design, which is iterative and dynamic, converging to the global optimum in an adaptive way (Sun et al., 2007).

One popular approach for cost-sensitive learning is to increase the penalty associated with the minority class using support vector machines which is referred to as weighted SVM (Osuna et al., 1997; Veropoulos et al., 1999; Akbani et al., 2004).

Singh and Purohit (2015) mention that in various data imbalance contexts like medical diagnosis, intrusion detection, and fraud detection, cost-sensitive learning performs better in comparison to external approaches such as resampling because in these types of datasets the cost of misclassification as well as class distribution is imbalanced. However, one shortcoming of cost-sensitive learning is that misclassification costs must be calculated a priori. Therefore, they must be determined for each problem separately. This requires either many trials or expert knowledge of the topic to decide.

Krawczyk et al. (2014) also state that it is difficult to set values in the cost matrix. They further state that in most cases the misclassification cost is unknown from the data and cannot be given by an expert. Nevertheless, there is an alternative way to address this difficulty, by setting the majority class misclassification cost at 1 while setting the penalty minority class value as equal to the IR (Castro & Braga, 2013; López et al., 2015).

Sun et al. (2007) propose the AdaCX algorithm, which combines cost-sensitive learning with the AdaBoost algorithm, aiming to give a larger weight to the minority class.

In the scope of this project, cost-sensitive learning is combined with Logistic Regression, Random Forest, and Extreme Gradient Boosting to predict the converting sessions. Section VI proves that cost-sensitive learning is indeed the most efficient approach for the problem at hand.

2) Thresholding

Thresholding is a set of methods that are used to remodel the decision boundary in a classifier via threshold moving (Kumar et al., 2021). There are a large variety of algorithms that generate probability estimates and these estimates are converted into predictions of a certain class. Thresholding aims to move this decision boundary that turns the probability into the binary outcome.

This paper does not focus on thresholding for every method that is applied. Rather, all the methods are compared according to the metrics that take into account every possible threshold such as the area under the curve of the Precision-Recall curve or average precision. Thresholding is applied to only the best-performing algorithm to come up with the confusion matrix and performance metrics such as the F1 score for the best-performing algorithm. The related analysis can be found in section VII.

IV) Combined Algorithms

Standard classifiers such as logistic regression (LR), decision tree (DT), and support vector machine (SVM) are suitable for balanced training sets. However, when they face with imbalanced scenarios, these models often provide suboptimal classification results (Ye et al., 2019).

Ensemble-based classifiers are known to improve the performance of a single classifier by combining several base classifiers that outperform every independent one (López et al., 2013). Witten et al. (2017) and Schapire (1990) similarly argue that one weak classifier that is slightly better than random conjecture can be promoted to a strong classifier by ensemble learning.

There are two popular frameworks for ensemble learning. The first framework is named as Bagging (Breiman, 1996). In the bagging framework, classifiers are run in parallel to different subsamples of the data, and the best classification scheme is chosen according to majority voting. The representative algorithm is the Random Forest algorithm (Verikas et al., 2011; Kumar et al.,2021; Kaur et al.,2019). The other one is the Boosting framework (Li et al., 2014). Boosting is an iterative strategy where each model tries to reduce the error of its predecessor. The representative algorithm is the Extreme Gradient Boosting algorithm.

Bagging is first noticed by researchers against imbalanced settings due to its simplicity and then multifarious algorithms have been developed, such as the AsBagging algorithm (Tao et al., 2006) and UnderOverBagging algorithms (Wang & Yao, 2009). The former algorithm combines random undersampling with bagging while the second approach employs hybrid sampling with bagging. The combination of boosting framework with other methods is also investigated by researchers some algorithms are developed such as the SMOTEBoost algorithm (Chawla et al., 2003) and the RUSBoost algorithm (Seiffert, 2009). They are both boosting techniques combined with SMOTE and random undersampling techniques.

EasyEnsemble algorithm and BalanceCascade algorithm are proposed by Liu and Zhou (2013). EasyEnsemble algorithm combines Bagging with Boosting. The Adaboost algorithm is used as a basic classifier on balanced subsamples of the dataset. The technique deploys random undersampling to generate balanced training datasets. EasyEnsemble algorithm can lower the variance and deviation of classification result, which makes the classification result become stable and presents a stronger generalization ability (Huang & Dai 2021).

Galar et al. (2011) develop a taxonomy of ensembles for imbalance learning. They find that the ensemble technique increases the performance with sampling and a single classifier in the presence of imbalanced data. As the number of classifiers increases, the problem becomes more complex, but the model also provides better performance for these unevenly distributed datasets. They also conclude that bagging and boosting approaches provide a better classification of imbalanced data. Menardi and Torelli (2014) similarly argue that boosting and bagging improve performance.

Yang and King (2009) apply neural networks along with Adaboost, SVM, and Logistic regression in a highly imbalanced e-commerce setting where the imbalance ratio is 1:49. They conclude that Adaboost obtains the best lift score and AUC score for both tasks analyzed in this project while Logistic Regression attains the second-best lift score and AUC score. Neural networks produce the worst results in this data.

Yijing et al. (2016) suggest that using a specific ensemble classifier to tackle all kinds of imbalanced data is inefficient. Therefore, this study deploys two different ensemble frameworks namely, Extreme Gradient Boosting and Random Forest along with the techniques against class imbalance. Logistic Regression is also included as a benchmark model to observe performance enhancement.

a) Logistic Regression

Logistic regression is the first classifier that this study utilizes in combination with the class imbalance methods. As mentioned earlier, standard classifiers are not expected to be performing well in the presence of class imbalance. However, logistic regression is still involved in the study as a benchmark.

b) Extreme Gradient Boosting

Most papers that address class imbalance with boosting are based on the first applicable boosting algorithm, Adaboost, proposed by Freund and Schapire (1996). Wu and Meng (2016) deploy the Adaboost algorithm along with SMOTE in a highly imbalanced e-commerce setting. Hao et al. (2014) similarly utilize SMOTE along with boosting for handling imbalanced PubChem BioAssay data. The experiments show that the proposed method outperforms the combination of Random Forest with SMOTE based on sensitivity and G-mean.

Bahad and Saxena (2020) compare the performance of Adaboost and Gradient Boosting algorithms for predictive analytics. The study mentions the effectiveness of Gradient Boosting to handle a heterogeneous feature dataset. Although the performance of the classifier depends on several factors such as characteristics of the dataset, the study suggests Gradient Boosting achieves better prediction accuracy than AdaBoost.

Tanha et al. (2020) analyze 14 binary and multi-class boosting algorithms investigated on 19 multi-class conventional and big datasets with various imbalanced ratios. The study results reveal that the CatBoost algorithm outperforms most of the other boosting methods in terms of MAUC, G-mean, and MMCC on 15 conventional datasets. SMOTEBoost is considered as second best while XGBoost and Logitboost are expressed as third best algorithms. Adaboost performs worse than the aforementioned methodologies.

Jiang et al. (2022) model highly imbalanced crash severity data by ensemble methods. According to their analysis, Gradient Boosting appears to be the most accurate one in terms of the overall F1 scores and MM in a group of learners where Adaboost is present. Park et al. (2015) apply gradient boosting in a highly imbalanced e-commerce setting and receive a significantly high-performing score in a competition leading the scoreboard.

As mentioned by Galar et al. (2011), boosting algorithms are usually combined with cost-sensitive learning and resampling techniques. Therefore, this study uses Extreme Gradient Boosting in combination with cost-sensitive learning and resampling methods to achieve high performance.

c) Random Forest

Random forest is the representative algorithm from the bagging methodologies. Since parallel ensembles have time-saving and ease-of-development advantages, they are recommended for solving practical problems (Haixiang et al., 2017).

In Bach et al. (2017), a comparative study of undersampling and oversampling is conducted on the predictions of osteoporosis diagnoses, where only 7.14 percent of the cases are that of the minority class. The focus is to identify which algorithms perform best while implementing these sampling techniques. They conclude that SMOTE along with the Random Forest classifier has the highest performance.

Khalilia et al. (2011) deploy SVM, boosting, and bagging algorithms in a highly imbalanced dataset. The experiments prove that on seven out of eight disease categories Random Forest outperformed the other classifiers in terms of AUC.

Somasonduram and Reddy (2016) similarly utilize several methods when the data has a severe imbalance ratio. They observe that most of the techniques perform well, however, the preference of the authors is towards the Random Forest classifier due to low false prediction rates and effective prediction of both the majority and minority classes.

The literature review and the results in Section VI prove that the ensemble methods Extreme Gradient Boosting and Random Forest produce accurate results.

V) Experimental Setting

The paper discusses the experimental setting in which the models are trained and the experimental results that are obtained in this section. There are three major subsections which are the evaluation metrics that the study utilizes, the model structure, and finally the hyperparameters of the models.

a) Evaluation Metrics

The choice of evaluation metrics is an integral part of imbalanced learning. Some papers even mention evaluation metrics as a separate method to tackle the class imbalance problem (Patel et al., 2020). The evaluation metrics that are commonly used in classification problems do not necessarily evaluate the performance objectively when there is a class imbalance. Table 2 illustrates major metrics that are applied in classification problems.

Metric	Formula
Accuracy	$\frac{TP+TN}{TP+FP+FN+TN}$
Sensitivity	$\frac{TP}{(TP+FN)}$
Specificity	$\frac{TN}{(TN+FP)}$
Precision	$\frac{TP}{(TP+FP)}$
Recall	$\frac{TP}{(TP+FN)}$
F-measure	$\frac{2*Precision*Recall}{Precision+Recall}$
G-MEAN	$\sqrt{sensitivity * specificity}$

Table 2: Major metrics in classification problems (Kaur et al., 2019)

The learning process guided by global performance metrics such as prediction accuracy induces a bias towards the majority class, while the rare episodes remain unknown even if the prediction model produces a high overall precision (Loyola-González et al., 2016). Anand et al. (2010) similarly discuss that, in the case of high imbalance, a naïve classifier classifying every observation as a majority class sample will result in high predictive accuracy. Therefore, accuracy is not a preferred performance metric.

The models focus on keeping both the true positives and true negatives in prediction high when trained with accuracy. However, keeping true positives high might be the priority in a highly imbalanced scenario such as predicting fraudulent transactions or predicting converting sessions as in the case of this paper. Therefore, the major focus should be placed on the minority class during the training phase and the application tolerates a small error rate in the majority class (López et al., 2013).

Sensitivity and specificity are two very common measures used in the medical community and increasingly in machine learning (Anand et al., 2010). Kubat et al. (1997) combine these two measures and suggests G-mean or balanced accuracy. A new performance measure, called the "adjusted geometric mean" (AGm) is also proposed (Batuwita & Palade, 2009). G-mean is a good performance metric when equal importance is given to the positive and negative instances. However, this is usually not the case for highly imbalanced datasets.

He and Garcia (2009) underline that the high level of imbalanced data makes it mandatory to examine the PR space. Precision and recall focus on keeping true positives high when false positives and false negatives are low. F-measure is a performance metric that

combines these two measurements. Performance metrics adapted to imbalanced data problems, such as F-measure($F\,m$), are less likely to suffer from imbalanced distributions as they take class distribution into account (Haixiang et al., 2017).

The F-measure comes up with the challenge of deciding a threshold to assign probability outcomes of the model to the classes. To tackle this problem while still observing the PR space, this paper employs PR AUC as the main performance metric. PR AUC proves to be a more robust and fair performance evaluation metric between different methods and combined algorithms. This metric takes into account all the possible thresholds in the space, and it calculates the area under the precision-recall curve.

Pedregosa et al. (2011) mention that the average precision score is not interpolated and is different from computing the area under the precision-recall curve with the trapezoidal rule, which uses linear interpolation and can be too optimistic. However, it is still widely used as an approximate value for the PR AUC. Therefore, this paper deploys the average precision score as the second evaluation metric due to its common usage and availability as a metric in sklearn libraries when training the models.

The chosen evaluation metrics make sure that the model predicts the minority class effectively which are the converting sessions whereas the misprediction rate is minimized. This can help the company in several ways in the future such as optimization of the inventory, and optimization of the user journey according to the converting sessions.

Finally, further thresholding is conducted for the best-performing model which is chosen according to PR AUC and average precision scores. The confusion matrix and metrics such as the F1 score are only calculated for the best-performing model after the comparative analysis is conducted.

b) Model Structure

Figure 1 illustrates the model structure that this paper utilizes. The structure is a slightly different version of k-fold cross-validation. The study applies either the chosen sampling or cost-sensitive learning method and standardization to each fold of the cross-validation schema to prevent data leakage in the training phase. The applied method and standardization are then fitted to the test set in each fold. The final model trained on five different folds is then fitted to the test data to evaluate the performance of the models.

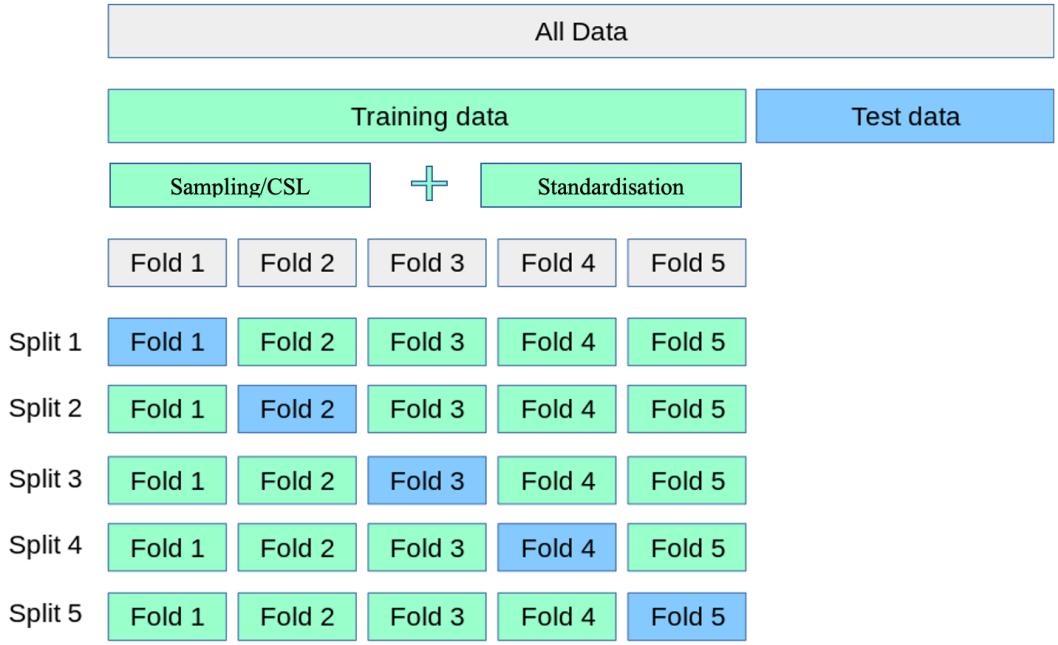

Figure 1: Model Structure (Pedregosa et al., 2011)

The study makes use of pipelines to apply the model structure mentioned above. Pipelines are an integral part of the sklearn library that enables users to apply the aforementioned models in an efficient way preventing data leakage.

c) Hyperparameter Tuning

This study utilizes a two-step hyperparameter tuning approach. The first step involves a Bayesian search with a very wide grid of parameter boundaries. In the second step, a very narrow grid is chosen according to the Bayesian search results.

Bayesian optimization keeps track of past results to choose the best hyperparameters for evaluation (Dewancker et al., 2015). The algorithm focuses to develop the output depending on the previous trials rather than an exhaustive search. Therefore, the outputs of the Bayesian search are utilized to choose a smaller grid in which an exhaustive search is conducted.

Park et al. (2015) conduct stratified 5-fold cross-validation on their training data and found that the following parameters work best for their models in a highly imbalanced e-commerce setting: n estimators: 5000, max leaf nodes: 20, min samples split: 1, learning rate: 0.17, max features: number of whole features. These hyperparameters are kept in the Bayesian grid for Random Forest and Extreme Gradient Boosting in our paper.

Bayesian search is applied for tuning the hyperparameters of XGB and Random Forest in cost-sensitive learning and random undersampling. It is not deployed for every approach due to performance and time constraints. In general, Bayesian search in Logistic Regression is not necessarily applied due to two main reasons:

1) Lack of hyperparameters to tune
2) Significantly low run time enables a large variety of parameters to be used in grid search

However, it is still deployed in the scope of this study for the sake of completeness to random undersampling. Table 3 illustrates the hyperparameters for each model, the grid boundaries chosen for the Bayesian search, the best parameter values found after the first step, and finally the values chosen for the grid search.

Algorithm	**Hyperparameters**	**Bayesian Search Values**	**Best Results**	**Grid Search Values**
Logistic Regression	penalty	L2	L2	L2
	max_iter	Integer (1000,5000)	3538	3538, 5000
	C	Real (0.001,1000)	911.81	911.8, 1000
	fit_intercept	[True,False]	True	True
	class_weight	[10,28,50,100,249]	-	10,28,50,100,249
Extreme Gradient Boosting	objective	binary:logistic	binary:logistic	binary:logistic
	learning_rate	(0.01, 0.2, “log-uniform")	0.01	0.01
	subsample	Real (0.5, 1)	0.805, 0.970	0.805,0.970
	colsample_bytree	Real (0.5, 1)	0.776,1	0.776,1
	n_estimators	Integer (1, 5000)	1684, 5000	5000
	max_depth	Integer (2, 10)	3, 9	3, 9
	scale_pos_weight	Integer (1,400)	28	28,249
Random Forest	max_features	Integer (1, 20)	20	20, None
	max_depth	Integer (2, 10)	10	10
	n_estimators	Integer (100, 5000)	5000	5000
	min_samples_split	Integer (2, 100)	2	2
	min_samples_leaf	Integer (1, 50)	1	1
	bootstrap	[True, False]	False	False
	class_weight	['balanced', 'balanced_subsample']	balanced_subsample	balanced_subsample, balanced, 10, 28, 100

Table 3: Hyperparameter Tuning Schema

As mentioned earlier, hyperparameter tuning is not very crucial for logistic regression. The random state is fixed for the reproducibility of the results in all three algorithms. The penalty term is determined as Ridge regression. The grids for max_iter and C values are kept as wide as possible to let the Bayesian search determine the best results. Class weight limits are chosen keeping in mind the imbalance ratio of the problem which is around 1:249.

Extreme Gradient Boosting is the second algorithm that needs hyperparameter tuning. The learning rate boundaries are kept small, and the distribution is determined as log-uniform to allow for more fits during the Bayesian search with a smaller learning rate. The ranges for the other hyperparameters for both Extreme Gradient Boosting and Random Forest algorithms are again kept as wide as possible also giving attention to not overfitting and the literature research as previously mentioned.

The hyperparameters scale_pos_weight and class_weight are deployed for the cost-sensitive learning models. These are the parameters that increase the weight associated with the minority class when the model learns from the dataset.

The best parameters determined from the Bayesian search guide the study to choose the final hyperparameter grid. The only additions to the grid apart from the Bayesian search results are for class weights. Additional parameters are also tried for Logistic Regression due to the short runtime of the algorithm. The results obtained by the final grid search are analyzed in detail in the following section of the paper.

VI) Comparative Analysis

Table 4 illustrates the results of the best algorithms on the test set after hyperparameter tuning. The family of the methods, the methods, combined algorithms, and the performance metrics of interest are shown in the table.

The results show that Extreme Gradient Boosting outperforms Logistic Regression and Random Forest significantly in all the class imbalance methods except undersampling. The overall best results of all the algorithms are obtained in combination with cost-sensitive learning. The performance between different methods of oversampling does not seem to differ significantly. Oversampling and hybrid sampling provide similar results. Undersampling shows the worst performance in terms of the highest scores of the learners in each methodology. Oversampling proves to be a better approach when combined with boosting in comparison to

undersampling which aligns with the direction of the literature review. In addition, the high performance of cost-sensitive learning is another result that verifies the expectations from the literature.

FAMILY OF METHODS	METHODS	METHOD NAME	ALGORITHM NAME	PR AUC	AVERAGE PRECISION
DATA-LEVEL METHODS	Undersampling	RUS	LR	0.0541	0.0559
		RUS	XGB	0.0523	0.0536
		RUS	RF	0.0816	0.0824
	Oversampling	SMOTE	LR	0.0333	0.0339
		SMOTE	XGB	0.2649	0.2656
		SMOTE	RF	0.0339	0.0354
		Borderline SMOTE	LR	0.0314	0.0320
		Borderline SMOTE	XGB	0.2688	0.2695
		Borderline SMOTE	RF	0.0649	0.0420
		ADASYN	LR	0.0343	0.0349
		ADASYN	XGB	0.2640	0.2646
		ADASYN	RF	0.0333	0.0347
	Hybrid Sampling	RUS + SMOTE	LR	0.0461	0.0468
		RUS + SMOTE	XGB	0.2570	0.2580
		RUS + SMOTE	RF	0.0417	0.0434
ALGORITHM-LEVEL METHODS	Cost Sensitive Learning	CSL	LR	0.0659	0.0668
		CSL	XGB	**0.3643**	**0.3645**
		CSL	RF	0.1898	0.1903
Baseline Result = Imbalance Ratio (IR)				0.004	0.004

Table 4: Experimental Results

Extreme Gradient Boosting in combination with cost-sensitive learning has the best performance according to the experimental results. In the next section, the study further analyzes this method. The section provides the confusion matrix before and after thresholding and further performance metrics such as the F1 score.

VII) Further Analysis of the Best Performing Model

Extreme Gradient Boosting along with cost-sensitive learning has shown the best performance according to the experiments as discussed in Section VI. Figure 2 illustrates the precision-recall curve for this model. The area under the curve below the precision-recall curve is 0.3643.

The analysis of ROC AUC is straightforward such as a score below 0.5 yields worse results than a random classifier. Therefore, a large variety of papers deploy ROC AUC as a performance metric. However, PR AUC does not have the same attribute due to not considering the true negatives in the sample. Saito and Rehmsmeier (2015) propose that the baseline result in the analysis of PR AUC should be the imbalance ratio. This approach is implemented in this paper and the baseline results in Table 4 illustrate the imbalance ratio between classes which is 0.004. Therefore, the proposed model works around 90 times better compared to the baseline results.

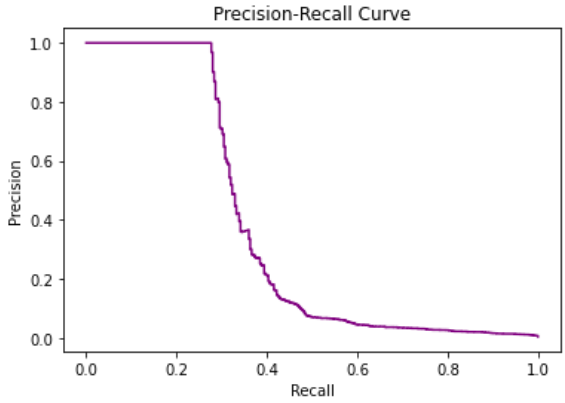

Figure 2: Precision-Recall Curve for the Best Performing Model

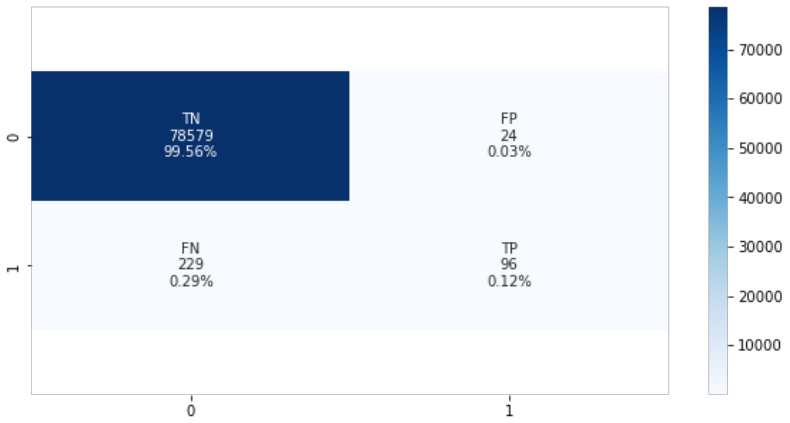

Figure 3: Confusion matrix of the best-performing model before thresholding

Figure 3 illustrates the confusion matrix for this model before thresholding. The threshold is kept at 0.5 which is the default value. The resulting performance metrics are as follows: Precision: 0.80, Recall: 0.30, and F1 score: 0.431

In addition, further thresholding is applied to the best model to create the confusion matrix in Figure 4. The thresholding is conducted such that the maximum F1 score in the test set would be achieved. The threshold is determined as 0,774. The resulting performance metrics are as follows: Precision: 0.97, Recall: 0.28, and F1 score: 0.434.

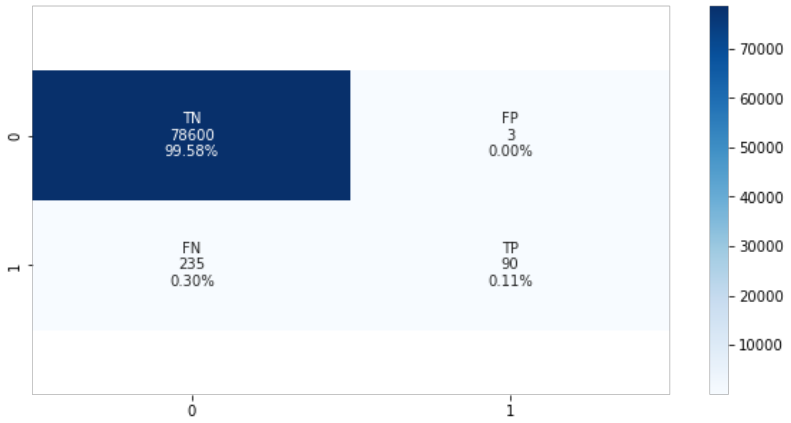

Figure 4: Confusion Matrix for the Best Performing Model after Thresholding

The performance metrics before and after the thresholding proves that the thresholding significantly improves the precision score which also results in a higher F1 score. The decline in the recall is insignificant in comparison to the other metrics.

VIII) Limitations and Future Work

The performance of every model is highly dependent on the features. This study utilizes every session-level feature that is available to improve the predictive performance. However, this performance may further be increased via tracking of additional features on the website which needs further engineering from the company.

The experiments are conducted on the company laptop due to data security concerns. Therefore, the computational power is highly insufficient to create large grids to check many different hyperparameter settings. One approach to tackle the problem is to deploy a two-step hyperparameter tuning with the inclusion of Bayesian search. However, this still could not be applied to every single model due to time constraints. Therefore, the fine-tuning of the models is an area open to progress.

This study focuses on choosing the best models depending on the literature. Some algorithms such as CatBoost and LightGBM are left out of the scope of this project and XGBoost is chosen as the representative algorithm due to similar logic and performance expectations. However, these models can also be deployed in combination with the methods against class imbalance to see the respective performances.

Moreover, the growth in the realm of machine learning and deep learning is exponential. Although a well-balanced set of methods and algorithms depending on the literature have been applied to tackle the problem of severe class imbalance, the area is still open to progress. The new methods bring always new opportunities which can be future work in this realm.

IX) Conclusions

Class imbalance is frequently encountered in machine learning tasks. This study focuses on the severe class imbalance in the e-commerce industry. The analysis shows that cost-sensitive learning with the combination of boosting performs significantly better in comparison to the other methods. In addition, the choice of evaluation metrics and the threshold becomes crucial depending on the problem and the importance of the classes. In this scenario, analysis of PR space proves to be efficient due to the high importance of minority examples. Moreover, further thresholding is also important to assign probability outputs to classes to maximize the performance of the confusion matrix and the related metrics such as the F1 score.

X) Link for the GitHub Repository

The code related to the study can be found in the following GitHub repository:

https://github.com/berkaykocak-cloud/Thesis

References

Abouelenien, M., Yuan, X., Giritharan, B., Liu, J., & Tang, S. (2013). Cluster-based sampling and ensemble for bleeding detection in capsule endoscopy videos. *American Journal of Science and Engineering*, *2*(1), 24-32.

Akbani, R., Kwek, S., & Japkowicz, N. (2004, September). Applying support vector machines to imbalanced datasets. In *European conference on machine learning* (pp. 39-50). Springer, Berlin, Heidelberg.

Anand, A., Pugalenthi, G., Fogel, G. B., & Suganthan, P. N. (2010). An approach for classification of highly imbalanced data using weighting and undersampling. Amino acids, 39(5), 1385-1391.

Bach, M., Werner, A., Żywiec, J., & Pluskiewicz, W. (2017). The study of under-and over-sampling methods' utility in analysis of highly imbalanced data on osteoporosis. *Information Sciences*, *384*, 174-190.

Bahad, P., & Saxena, P. (2020). Study of adaboost and gradient boosting algorithms for predictive analytics. In *International Conference on Intelligent Computing and Smart Communication 2019* (pp. 235-244). Springer, Singapore.

Batuwita, R., & Palade, V. (2009, December). A new performance measure for class imbalance learning. application to bioinformatics problems. In *2009 International Conference on Machine Learning and Applications* (pp. 545-550). IEEE.

Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*(2), 123-140.

Buda, M., Maki, A., & Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks*, *106*, 249-259.

Burez, J., & Van den Poel, D. (2009). Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, *36*(3), 4626-4636.

Castro, C. L., & Braga, A. P. (2013). Novel cost-sensitive approach to improve the multilayer perceptron performance on imbalanced data. *IEEE transactions on neural networks and learning systems*, *24*(6), 888-899.

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

Chawla, N. V., Lazarevic, A., Hall, L. O., & Bowyer, K. W. (2003, September). SMOTEBoost: Improving prediction of the minority class in boosting. In *European conference on principles of data mining and knowledge discovery* (pp. 107-119). Springer, Berlin, Heidelberg.

Dai, H. L. (2015). Class imbalance learning via a fuzzy total margin based support vector machine. *Applied Soft Computing*, *31*, 172-184.

Denil, M. , & Trappenberg, T. (2010). Overlap versus Imbalance. In Canadian Conference on Advances in Artificial Intelligence.

Dewancker, I., McCourt, M., & Clark, S. (2015). Bayesian optimization primer. *URL https://app. sigopt. com/static/pdf/SigOpt_ Bayesian_Optimization_Primer. pdf*.

Dhote, S., Vichoray, C., Pais, R., Baskar, S., & Mohamed Shakeel, P. (2020). Hybrid geometric sampling and AdaBoost based deep learning approach for data imbalance in E-commerce. *Electronic Commerce Research*, *20*(2), 259-274.

Freund, Y., & Schapire, R. E. (1996, July). Experiments with a new boosting algorithm. In *icml* (Vol. 96, pp. 148-156).

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2011). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *42*(4), 463-484.

García, V., Sánchez, J. S., & Mollineda, R. A. (2012). On the effectiveness of preprocessing methods when dealing with different levels of class imbalance. *Knowledge-Based Systems*, *25*(1), 13-21.

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert systems with applications*, *73*, 220-239.

Han, H., Wang, W. Y., & Mao, B. H. (2005, August). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing* (pp. 878-887). Springer, Berlin, Heidelberg.

Hao, M., Wang, Y., & Bryant, S. H. (2014). An efficient algorithm coupled with synthetic minority over-sampling technique to classify imbalanced PubChem BioAssay data. *Analytica chimica acta*, *806*, 117-127.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008, June). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)* (pp. 1322-1328). IEEE.

He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, *21*(9), 1263-1284.

Huang, C. Y., & Dai, H. L. (2021). Learning from class-imbalanced data: review of data driven methods and algorithm driven methods. *Data Science in Finance and Economics*, *1*(1), 21-36.

Japkowicz, N. (2000, June). The class imbalance problem: Significance and strategies. In *Proc. of the Int'l Conf. on Artificial Intelligence* (Vol. 56, pp. 111-117).

Japkowicz, N., & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent data analysis*, *6*(5), 429-449.

Jiang, L., Xie, Y., Wen, X., & Ren, T. (2022). Modeling highly imbalanced crash severity data by ensemble methods and global sensitivity analysis. *Journal of Transportation Safety & Security*, *14*(4), 562-584.

Kaur, H., Pannu, H. S., & Malhi, A. K. (2019). A systematic review on imbalanced data challenges in machine learning: Applications and solutions. *ACM Computing Surveys (CSUR)*, *52*(4), 1-36.

Khalilia, M., Chakraborty, S., & Popescu, M. (2011). Predicting disease risks from highly imbalanced data using random forest. *BMC medical informatics and decision making*, *11*(1), 1-13.

Krawczyk, B., & Schaefer, G. (2013, May). An improved ensemble approach for imbalanced classification problems. In *2013 IEEE 8th international symposium on applied computational intelligence and informatics (SACI)* (pp. 423-426). IEEE.

Krawczyk, B., Woźniak, M., & Schaefer, G. (2014). Cost-sensitive decision tree ensembles for effective imbalanced classification. *Applied Soft Computing*, *14*, 554-562.

Kubat, M., Holte, R., & Matwin, S. (1997, April). Learning when negative examples abound. In *European conference on machine learning* (pp. 146-153). Springer, Berlin, Heidelberg.

Kubat, M., & Matwin, S. (1997, July). Addressing the curse of imbalanced training sets: one-sided selection. In *Icml* (Vol. 97, No. 1, p. 179).

Kubat, M., Holte, R. C., & Matwin, S. (1998). Machine learning for the detection of oil spills in satellite radar images. *Machine learning*, *30*(2), 195-215.

Kumar, P., Bhatnagar, R., Gaur, K., & Bhatnagar, A. (2021, March). Classification of imbalanced data: review of methods and applications. In *IOP Conference Series: Materials Science and Engineering* (Vol. 1099, No. 1, p. 012077). IOP Publishing.

Li, J., Fong, S., Mohammed, S., & Fiaidhi, J. (2016). Improving the classification performance of biological imbalanced datasets by swarm optimization algorithms. *The Journal of Supercomputing*, *72*(10), 3708-3728.

Li, K., Kong, X., Lu, Z., Wenyin, L., & Yin, J. (2014). Boosting weighted ELM for imbalanced learning. *Neurocomputing*, *128*, 15-21.

Liu, X. Y., Wu, J., & Zhou, Z. H. (2008). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *39*(2), 539-550.

Liu, X. Y., & Zhou, Z. H. (2013). Ensemble methods for class imbalance learning. *Imbalanced Learning: Foundations, Algorithms, and Applications*, 61-82.

Longadge, R., & Dongre, S. (2013). Class imbalance problem in data mining review. *arXiv preprint arXiv:1305.1707*.

López, V., Del Río, S., Benítez, J. M., & Herrera, F. (2015). Cost-sensitive linguistic fuzzy rule based classification systems under the MapReduce framework for imbalanced big data. *Fuzzy Sets and Systems*, *258*, 5-38.

López, V., Fernández, A., García, S., Palade, V., & Herrera, F. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information sciences*, *250*, 113-141.

López, V., Fernández, A., Moreno-Torres, J. G., & Herrera, F. (2012). Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics. *Expert Systems with Applications*, *39*(7), 6585-6608.

Loyola-González, O., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A., & García-Borroto, M. (2016). Study of the impact of resampling methods for contrast pattern based classifiers in imbalanced databases. *Neurocomputing*, *175*, 935-947.

Menardi, G., & Torelli, N. (2014). Training and assessing classification rules with imbalanced data. *Data mining and knowledge discovery*, *28*(1), 92-122.

Osuna, E., Freund, R., & Girosit, F. (1997, June). Training support vector machines: an application to face detection. In *Proceedings of IEEE computer society conference on computer vision and pattern recognition* (pp. 130-136). IEEE.

Park, C., Kim, D., Oh, J., & Yu, H. (2015). Predicting user purchase in e-commerce by comprehensive feature engineering and decision boundary focused under-sampling. In *Proceedings of the 2015 International ACM Recommender Systems Challenge* (pp. 1-4).

Patel, H., Singh Rajput, D., Thippa Reddy, G., Iwendi, C., Kashif Bashir, A., & Jo, O. (2020). A review on classification of imbalanced data for wireless sensor networks. *International Journal of Distributed Sensor Networks*, *16*(4), 1550147720916404.

Pazzani, M., Merz, C., Murphy, P., Ali, K., Hume, T., & Brunk, C. (1994). Reducing misclassification costs. In *Machine Learning Proceedings 1994* (pp. 217-225). Morgan Kaufmann.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research, 12, 2825–2830.*

Qian, Y., Liang, Y., Li, M., Feng, G., & Shi, X. (2014). A resampling ensemble algorithm for classification of imbalance problems. *Neurocomputing*, *143*, 57-67.

Ramyachitra, D., & Manikandan, P. (2014). Imbalanced dataset classification and solutions: a review. *International Journal of Computing and Business Research (IJCBR)*, *5*(4), 1-29.

Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, *10*(3), e0118432.

Schapire, R. E. (1990). The strength of weak learnability. *Machine learning*, *5*(2), 197-227.

Seiffert, C., Khoshgoftaar, T. M., Van Hulse, J., & Napolitano, A. (2009). RUSBoost: A hybrid approach to alleviating class imbalance. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *40*(1), 185-197.

Singh, A., & Purohit, A. (2015). A survey on methods for solving data imbalance problem for classification. *International Journal of Computer Applications*, *127*(15), 37-41.

Solberg, A. S., & Solberg, R. (1996, May). A large-scale evaluation of features for automatic detection of oil spills in ERS SAR images. In *IGARSS'96. 1996 International Geoscience and Remote Sensing Symposium* (Vol. 3, pp. 1484-1486). IEEE.

Somasundaram, A., & Reddy, U. S. (2016, September). Data imbalance: effects and solutions for classification of large and highly imbalanced data. In *international conference on research in engineering, computers and technology (ICRECT 2016)* (pp. 1-16).

Sun, Y., Kamel, M. S., Wong, A. K., & Wang, Y. (2007). Cost-sensitive boosting for classification of imbalanced data. *Pattern recognition*, *40*(12), 3358-3378.

Tahir, M. A., Kittler, J., Mikolajczyk, K., & Yan, F. (2009, June). A multiple expert approach to the class imbalance problem using inverse random under sampling. In *International workshop on multiple classifier systems* (pp. 82-91). Springer, Berlin, Heidelberg.

Tang, Y., Zhang, Y. Q., Chawla, N. V., & Krasser, S. (2008). SVMs modeling for highly imbalanced classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *39*(1), 281-288.

Tanha, J., Abdi, Y., Samadi, N., Razzaghi, N., & Asadpour, M. (2020). Boosting methods for multi-class imbalanced data classification: an experimental review. *Journal of Big Data*, *7*(1), 1-47.

Tao, D., Tang, X., Li, X., & Wu, X. (2006). Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval. *IEEE transactions on pattern analysis and machine intelligence*, *28*(7), 1088-1099.

Tsai, C. F., & Lin, W. C. (2021). Feature selection and ensemble learning techniques in one-class classifiers: an empirical study of two-class imbalanced datasets. *IEEE Access*, *9*, 13717-13726.

Verikas, A., Gelzinis, A., & Bacauskiene, M. (2011). Mining data with random forests: A survey and results of new tests. *Pattern recognition*, *44*(2), 330-349.

Veropoulos, K., Campbell, C., & Cristianini, N. (1999, July). Controlling the sensitivity of support vector machines. In *Proceedings of the international joint conference on AI* (Vol. 55, p. 60).

Wang, L., Han, M., Li, X., Zhang, N., & Cheng, H. (2021). Review of classification methods on unbalanced data sets. *IEEE Access*, *9*, 64606-64628.

Wang, S., & Yao, X. (2009, March). Diversity analysis on imbalanced data sets by using ensemble models. In *2009 IEEE symposium on computational intelligence and data mining* (pp. 324-331). IEEE.

Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2017). Ensemble learning. *Witten IH, Author, Data Mining*, *4*, 479-501.

Wu, X., & Meng, S. (2016, June). E-commerce customer churn prediction based on improved SMOTE and AdaBoost. In *2016 13th International conference on service systems and service management (ICSSSM)* (pp. 1-5). IEEE.

Yang, H., & King, I. (2009, December). Ensemble learning for imbalanced e-commerce transaction anomaly classification. In *International Conference on Neural Information Processing* (pp. 866-874). Springer, Berlin, Heidelberg.

Ye, Z. F., Wen, Y. M., & Lu, B. L. (2019). A review of imbalanced classification. *J Intell Syst*, *4*, 148-156.

Yijing, L., Haixiang, G., Xiao, L., Yanan, L., & Jinling, L. (2016). Adapted ensemble classification algorithm based on multiple classifier system and feature selection for classifying multi-class imbalanced data. *Knowledge-Based Systems*, *94*, 88-104.

Yu, H., Sun, C., Yang, X., Yang, W., Shen, J., & Qi, Y. (2016). ODOC-ELM: Optimal decision outputs compensation-based extreme learning machine for classifying imbalanced data. *Knowledge-Based Systems*, *92*, 55-70.

Zhang, H., & Li, M. (2014). RWO-Sampling: A random walk over-sampling approach to imbalanced data classification. *Information Fusion*, *20*, 99-116.

Zong, W., Huang, G. B., & Chen, Y. (2013). Weighted extreme learning machine for imbalance learning. *Neurocomputing*, *101*, 229-242.

Declaration of Academic Honesty

I, Berkay Kocak, hereby declare that I have not previously submitted the present work for other examinations. I wrote this work independently. All sources, including sources from the Internet, that I have reproduced in either an unaltered or modified form (particularly sources for texts, graphs, tables, and images), have been acknowledged by me as such.

I understand that violations of these principles will result in proceedings regarding deception or attempted deception.

Berkay Kocak
Berlin, 24.11.2022