IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 20, 2019, accepted November 23, 2019, date of publication December 23, 2019, date of current version January 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2961784

Uncertainty Based Under-Sampling for Learning Naive Bayes Classifiers Under Imbalanced Data Sets

CHRISTOS K. ARIDAS[1], STAMATIS KARLOS[1], VASILEIOS G. KANAS[2], NIKOS FAZAKIS[2], AND SOTIRIS B. KOTSIANTIS[1]

[1]Department of Mathematics, University of Patras, 26504 Patras, Greece
[2]Department of Electrical and Computer Engineering, University of Patras, 26504 Patras, Greece

Corresponding author: Christos K. Aridas (char@upatras.gr)

This research was implemented through the Operational Program Human Resources Development, Education and Lifelong Learning and is co-financed by the European Union (European Social Fund) and Greek national funds.

ABSTRACT In many real world classification tasks, all data classes are not represented equally. This problem, known also as the curse of class imbalanced in data sets, has a potential impact in the training procedure of a classifier by learning a model that will be biased in favor of the majority class. In this work at hand, an under-sampling approach is proposed, which leverages the usage of a Naive Bayes classifier, in order to select the most informative instances from the available training set, based on a random initial selection. The method starts by learning a Naive Bayes classification model on a small stratified initial training set. Afterwards, it iteratively teaches its base model with the instances that the model is most uncertain about and retrains it until some criteria are satisfied. The overall performance of the proposed method has been scrutinized through a rigorous experimental procedure, being tested using six multimodal data sets, as well as another forty-four standard benchmark data sets. The empirical results indicate that the proposed under-sampling method achieves comparable classification performance in contrast to other resampling techniques, regarding several proper metrics and having performed a suitable statistical testing procedure.

INDEX TERMS Active selection, classification, naive bayes, imbalanced data, under-sampling.

I. INTRODUCTION

Supervised learning, and specifically classification, is one of the most widely used Machine learning (ML) and Data Mining (DM) tasks. In real world data sets, the distribution of the class members is not always even. This leads to a situation which is called imbalanced problem [1]–[4], also known as ''the curse of imbalanced data sets'', which is the problem of learning an hypothesis from a class that is underrepresented in a data set. This problem is encountered in multiple domain specific tasks such as bioinformatics, fraud detection and medical diagnosis, among others and has been considered one of the top 10 problems in data mining and pattern recognition [5], [6]. Imbalanced data sets usually influence the learning process of a model, since most of the learning algorithms make the assumption of even distribution among classes or equal miss-classification costs [2]. For this reason, several methods have been proposed to address this problem that arises in this specific type of data sets [7], which should be inserted into the learning chain of automated solutions when this kind of anomaly takes place.

Usually, the methods that have been developed to cope with the imbalanced problem can be categorized in three distinct groups: algorithmic level methods, data level methods and hybrid approaches [7]–[10]. Algorithmic level approaches work either by adjusting the decision threshold or by modifying the optimization function [7], [11], [12], or even by providing miss-classification costs for each class to the learning algorithm [13]–[16]. Data level approaches work mainly in the prepossessing step modifying the training data, oriented towards coping with the skewed classes and providing a more balanced data set to the next stage. These methods can be further categorized in: 1) under-sampling methods [17]–[26], where they try to eliminate instances from the the majority class, 2) over-sampling methods [27]–[34], where they try to replicate or generate instances for the minority class and

The associate editor coordinating the review of this manuscript and approving it for publication was Juan Liu.

3) combination of over- and under-sampling [35], [36]. Lastly, hybrid approaches combine sampling techniques with ensemble methods [8].

The aim of this research work is to propose an under-sampling method that will actively select a subset from a training set, after modifying it appropriately. This subset aims to be more informative from the original without sacrificing the performance of the final classifier. The proposed method has been tested on several multimodal and standard benchmark data sets for imbalanced classification tasks, both exploiting them based on their initial structure and modifying them appropriately, so as to exist a challenging Imbalanced Ratio (IR) over binary problems. From the obtained results, it has been observed that the proposed method tends to produce data sets that outperform other well-known and widely used resampling methods in terms of *AUC* and Balanced Accuracy.

The rest of the paper is organized as follows: In Section II some of the most well-known under-sampling methods are briefly reviewed. In Section III the proposed algorithm is presented and analyzed. In Section IV experimental results obtained using six multimodal and forty-four standard benchmark data sets are exhibited. The paper ends in Section V with a synopsis and concluding remarks.

II. RELATED WORK

Recent research articles indicate that the imbalanced problem is an active research field in ML. Data level methods in contrast to other family of methods have a potential broader applicability since their goal is to fix the skewed distribution of data sets, rather than depend on supervised learning-based modifications. Over-sampling refers to a family of algorithms that contain approaches that replicate or generate a number of artificially created instances for the minority class. Random over-sampling is a non-heuristic method that seeks to balance class distribution using random replication of the minority class instances. Several authors have concluded that random over-sampling can increase the likelihood of occurring over-fitting, since it makes exact copies of the minority class instances [7]. The most well-known over-sampling method is the Synthetic Minority Over-Sampling Technique (*SMOTE*) which generates artificial instances by interpolating existing instances and avoids the risk of over-fitting, unlike random over sampling [27]. Since its publication, a number of extension methods have been proposed, mainly in order to cope with its disadvantages [37]. Such methods fix the original method's weaknesses, such as to handle differently some minority class regions, cope with within class imbalance or to eliminate the replication of noise.

On the other hand, under-sampling methods aim to tackle the imbalanced problem mainly by eliminating majority class instances. They can be further categorized in prototype generation and prototype selection methods. Given a training data set S, prototype generation methods generate a new set S' where $|S'| < |S|$ and $S' \notin S$. Prototype generation methods will reduce the number of instances in the targeted classes but the remaining instances are generated and not selected from the original set [38]. On the contrary to prototype generation algorithms, prototype selection algorithms will select samples from the original set S. Therefore, S' is defined such as $|S'| < |S|$ and $S' \subset S$.

The most naive approach to under-sampling is the random under-sampling [17] which is a non-heuristic method that aims to balance the class distribution through the random elimination of majority class examples. The major drawback of random under-sampling is that this method can discard potentially useful data that could be important for the classification process.

Edited Nearest Neighbors (*ENN*) is one of the first approaches that have been recorded for editing the provided data D, in order to reduce their cardinality, without sacrificing much of the total information that is initially contained, or on ideal scenario, to improve the insights that are underlying into this, as has been proposed by D. Wilson [18]. The proposed process is oriented towards boosting the performance of Nearest Neighbor (*NN*) estimators. Thus, the preprocessing stage that is applied before the fit of *1NN* is the three-nearest neighbor rule (modified rule), which eliminates instances that do not agree with the majority in examined neighbors. Its performance, regarding both accuracy and asymptotic convergence, mainly against Naive Bayes algorithm, seems fair enough. It has to be referred, that the use of larger size of neighbors during the modified rule demands much more data in order to reach to similar results.

A Tomek's [19] link between two instances of different classes x and y is defined such that for any instance z: $d(x, y) < d(x, z)$ and $d(x, y) < d(y, z)$ where $d(.)$ is the distance between the two instances. So, Tomek's links removal is a technique that removes pairs of instances only if they belong to different classes but are each other's nearest neighbors.

One Side Selection [20] is an approach that categorizes all the instances of majority class into four categories: i) noisy, ii) borderline, iii) redundant, iv) safe. Its approach is to find a subset of initially provided training set, which should be consistent regarding *1-NN* classifier: this means that the selected learner predicts accurately all included instances to this subset. The approach here does not try to find the smallest one consistent. Thus, it starts using all minority examples and only one from majority. Then it appends all the miss-classified instances from the rest of the set and applies a discarding method based on Tomek Links, so as to remove borderline and noisy data from the previous step.

One direct improvement of *OSS* is Neighboring Cleaning Rule (*NCL*) [21] that addresses the main drawback of its ancestor: the application of Tomek line criterion that is sensitive to noisy data. Thus, noisy data are removed according to concept of *ENN* method. Firstly, by deleting instances from majority class locally, based on the edited nearest neighbor rule and then by reducing instances from proper classes according to 3-NN, whose decisions are misclassified. The experiments that were executed based on *3-NN* and C4.5 were really encouraging, removing less instances against simple

random selection within classes (*SWC*) and *OSS* over ten data sets from *UCI*.

NearMiss [22] family of under-sampling methods that select instances from the majority class based on their distance to other instances in the same class. Let the positive instances to be the instances that belong to the targeted class to be under-sampled. Negative instances are referring to the instances from the minority class. *NearMiss-1* selects the positive instances for whose the mean distance to the N closest samples of the negative class is the smallest. *NearMiss-2* choose the positive instances whose the mean distance to the N farthest samples of the negative class is the smallest. While *NearMiss-3* is a two step procedure: At first, for each negative sample, their M nearest-neighbors will be kept aside. Then, the positive instances that are selected are those that their mean distance to the N nearest-neighbors is the largest.

Under-sampling methods may lose some useful information or ignore noise in the datasets. Thus, Hou et al. [23] proposed a density-based under-sampling algorithm (*DBU*) to overcome the two aforementioned problems. Similar examples are expected to be close to each other and a noisy example to be far from other examples belonging to the same class. Therefore, similar instances should have a high density while the noisy instances should have a low density. *DBU* uses the local density peaks to represent the whole majority class, so that it can retain the useful information and eliminate the noisy instances automatically.

Under-sampling approaches based on clustering have recently been proposed [17], [24], [25]. In [24] the authors proposed a novel under-sampling approach which is called cluster-based instance selection (*CBIS*), that combines clustering analysis and instance selection. The clustering phase of *CBIS* groups similar instances of the majority class into *subclasses*, while in the instance selection phase the method filters out unrepresentative instances from each of the *subclasses*. Lemaître et al. [17] implemented two under-sampling strategies that use data partition algorithms to preprocess the majority class. Concretely, the number of clusters in the majority class is set to be equal to the number of instances of the minority class. The first strategy uses the cluster centers to represent the majority class, while the second strategy uses the nearest neighbors of the cluster centers. Ofek et al. [25] proposed *Fast-CBUS*, a fast, novel clustering-based under-sampling technique which demonstrates high predictive performance, while its time complexity is bound by the size of the minority class instances. In the training phase, this method clusters the minority examples and selects a similar number of majority examples from each cluster. A specific classification model is then trained for each cluster. An unlabeled example is classified as the majority class, if it does not fit into any of the clusters. Otherwise, cluster-specific classifiers are used to return the example's classification and the results are weighted by the inverse-distance from the clusters.

An active selection method has been employed in 2009 for handling imbalanced data, on contrast with resampling and cost-sensitive methods, trying to build incrementally a trustworthy data set for applying a simple classifier in the field of Biomedical data [26]. This strategy, named as Active Example Selection (*AES*), is applied in order to avoid using instances that are not totally useful for the finally exported classifier, since a large number of instances, mainly stemming from majority class, are redundant and induce time-consuming training stage without any predictive improvement. Hence, a small random subset of provided data is kept and used as the training set with the same population of both initially minority and majority classes. The remaining examples are used as the validation set and the selected base learner produces its class probabilities per instance. The instances for which the worst decisions were formatted are moved iteratively to the training set, independently on their class category. The exploited base learner in this work is NB, because of its simplistic structure that does not depend on many parameters and demands only a few data so as to be tuned. Five different data sets, four from *UCI* and one from real-life used mainly with artificial networks were examined against the approaches that use cost-sensitive, Random Over-sampling, Random Under-sampling and a hybrid of these last two methods, all combined with NB, managing to outperform them as far as several useful metrics are concerned.

III. PROPOSED METHOD

The aim of the work at hand is to provide a method that will cope with the problems that arise in classification when the available training data set is imbalanced. The method will edit the available training data set and will provide a reduced set to the final classifier, with the aim to decrease the training time while increasing the classification performance.

The high level concept of the proposed method is that a small balanced subset of the available training set is picked at random and trains a Naive Bayes classification model. The remaining set gets balanced using *SMOTE* and acts as a pool for the active selection procedure. For a prespecified number of queries, the active selection procedure is performed, in order to teach the Naive Bayes classification model and get the most out of the remaining data pool.

In this research work, we used Naive Bayes in the proposed under-sampling approach, as well as in the final classification, as it is stated in Section IV. Naive Bayes [39], [40] learning algorithm is a family of algorithms that learn a probabilistic model. Based on Bayes' theorem makes the naive assumption of independence among every pair of predictor variables given the value of the target class. Given class variable y and dependent feature vector $x = (x_1 \dots x_n)$ Bayes' theorem states the following relationship

$$P(y \mid x_1, \dots, x_n) = \frac{P(y)P(x_1, \dots x_n \mid y)}{P(x_1, \dots, x_n)} \tag{1}$$

assuming the naive conditional independence that

$$P(x_i|y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i|y), \tag{2}$$

for all i this relationship is simplified to

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y) \prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)} \tag{3}$$

Since $P(x_1, \ldots, x_n)$ is constant given the input, the following classification rule can be used:

$$P(y \mid x_1, \ldots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y)$$

$$\Downarrow$$

$$\hat{y} = \arg\max_{y} P(y) \prod_{i=1}^{n} P(x_i \mid y), \tag{4}$$

where, x_i represents the i-th predictor variable. Using the maximum a posteriori probability we can obtain estimates for $P(y)$ and $P(x_i \mid y)$ from the available training data. Learning algorithms that are based on Bayes' theorem differ mainly by the assumption they make regarding the distribution of the likelihood $P(x_i \mid y)$. When having data sets with numerical features, a common assumption is that the likelihood of the features is assumed to be Gaussian

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right) \tag{5}$$

Maximum likelihood estimation is used to compute σ_y^2 and μ_y.

Apart from its naive assumption of conditional independence, Naive Bayes classification has shown good performances in many complex real-world problems [41]–[47].

The entire algorithmic procedure of the proposed approach is presented in Algorithm 1. Also an example of an under sampled data set, that was generated by the proposed approach, appears in Figure 1. Consider that the training data set is denoted as D and the new data set that will be generated is denoted as S. Initially, a small number of instances m for each class is selected at random. The selected instances are added to the set S (line 8) and are removed from the set D (line 9). The number m (line 7) can be determined by multiplying the total number of minority instances n (line 6) in D by an r parameter. The remaining set D is kept aside and is used as a pool of instances in order to teach the classification model in the active selection procedure. Since D is imbalanced, *SMOTE* (line 10) is applied, with k being the number of nearest neighbors and a new set D' is generated, which will has the same distribution for each class. The detailed algorithm of *SMOTE* is described in Algorithm 2. Afterwards, a classification model is trained using Naive Bayes algorithm on the data set S. For a prespecified number of active selection queries q do the following: 1) use the Naive Bayes classification model to predict class membership probabilities for each instance in D'. Convert the class probabilities to uncertainties (line 15-16) which are defined by $U(x) = 1 - P(\hat{y}|x)$, where x is the instance to be predicted and $\hat{y}$ (line 18) is the most likely prediction. 2) Remove most uncertain instance from the set D' (line 22) and add it to the set S (line 20). 3) The Naive Bayes algorithm is retrained (line 21) using the updated set S. 4) If the prespecified number of queries reaches the limit (line 12), or the score of Balanced Accuracy [48], [49] against D' (line 24) exceeds a certain threshold t (line 25), stop and return the set S.

Algorithm 1 The proposed method

```
1:  data ← Training data set
2:  q ← Number of active selection queries
3:  r ← Ratio of the minority class
4:  t ← Balanced accuracy threshold
5:  k ← Nearest neighbors considered by SMOTE
6:  Begin
7:     n ← getTheNumberOfMinorityInstances(data)
8:     m ← n * r
9:     generatedSet ← getInitTrainSet(data, m)
10:    deleteInstances(data, generatedSet)
11:    dataPool ← SMOTE(data, k)
12:    model ← trainNaiveBayesClassifier(generatedSet)
13:    for 1 to q do
14:       uncertainties ← ∅
15:       for instance ∈ dataPool do
16:          u ← getUncertainty(model, instance)
17:          uncertainties ← uncertainties ∪ {u}
18:       end for
19:       index ← getMaximumUncertainty(uncertainties)
20:       instance ← dataPool[index]
21:       generatedSet ← generatedSet ∪ {instance}
22:       model ← trainNaiveBayesClassifier(generatedSet)

23:       deleteInstance(dataPool, instance)
24:       bacc ← measureBalancedAccuracy(model, dataPool)

25:       if bacc >= t then
26:          break
27:       else
28:          continue
29:       end if
30:    end for
31:    return data
32: End
```

IV. NUMERICAL EXPERIMENTS

In this section, the data set descriptions, the design of the experiments, as well as the results and the statistical analysis of the experiments are presented. Specifically, in Subsection IV-A the data sets that are used in the experiments are presented. In Subsection IV-B the evaluation protocol is described as well as the other resampling techniques that are included in the comparisons. While, In Subsection IV-C the performance of the proposed method is compared against other well-known resampling techniques.

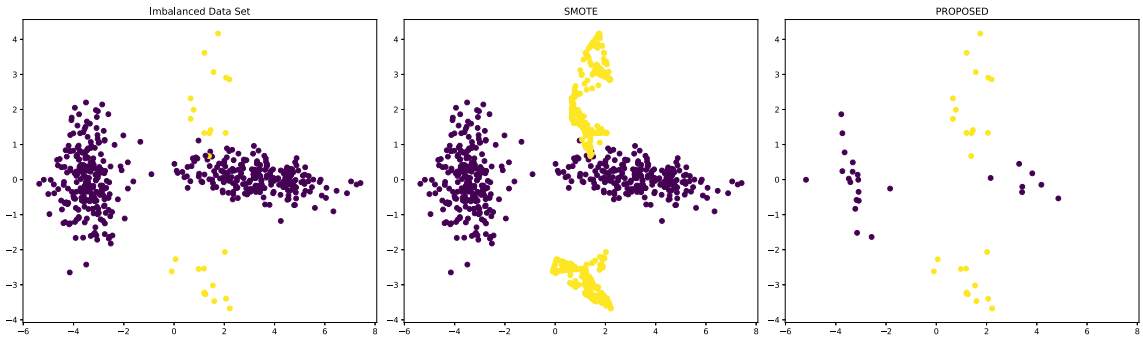

FIGURE 1. Resampling on an artificial data set.

Algorithm 2 SMOTE

data ← Training data set
k ← Number of nearest neighbors
Begin
  Compute: *nMajority* and *nMinority*. The number of majority and minority instances respectively in *data*
  $n \leftarrow nMajority - nMinority$
  for 1 **to** n **do**
    Choose at random a minority instance $\vec{a}$
    Choose at random a neighboring instance $\vec{b}$ among its k nearest minority class neighbors
    $\vec{x} \leftarrow \vec{a} + \lambda \times (\vec{b} - \vec{a})$, where λ is a random number in the range [0, 1]
    $data \leftarrow data \cup \{\vec{x}\}$
  end for
  return *data*
End

A. DATA SETS DESCRIPTION

In order to evaluate the performance of the proposed method, a number of experiments have been conducted, using two different sets of data sets. In Tables 1 and 2 the most important characteristics per each data set are exhibited: name, number of instances, number of attributes, its imbalanced ratio (*IR*), as well as the number of different classes.

The first set of data sets contains six multimodal data sets that were collected from independent/different data sources. Arabic Natural Audio Data set (anad) is a recently formatted problem related with sentimental analysis based on video signals from Arabic talk shows. There, the signals that were analyzed concerned talks between human and artificial anchors that were later provided to human listeners so as to categorize the extracted emotion. Three different classes had been defined as a possible outcome of emotion: happy, angry and surprised. During the pre-process stage, the most representative reactions were removed (e.g. laughs) and then the total amount of signals was split into chunks, whose duration was equal to 1 second. The feature engineering stage contains exploitance of low-level descriptors that stem from sound signals, applying numerous mathematical functions over them [50].

TABLE 1. Multimodal data sets description.

Data set	#Examples	#Attributes	IR
36speakers-man-vs-all	89014	25	41.1
36speakers-woman-vs-all	89014	25	38.4
anad-angry-vs-surprised	878	844	5.4
anad-happy-vs-surprised	642	844	3.7
wesad-eda-baseline-vs-amusement	18506	45	31.0
wesad-eda-baseline-vs-stress	18953	45	17.5
voice	1743	20	10.0

Gender Recognition by Voice Data set (voice) [1] constitutes a well-known task that was created so as to discriminate through speech signals the speakers' gender: male or female. Some preprocessing actions have been made using suitable R packages, following the specifications of human voice frequency bandwidth (citations of R packages). Different sources were used for collecting the finally included speech samples, while some widely accepted acoustic features were regarded, after having postulated some assumptions for correct validation of the formatted data set.

Characterizing Individual Speakers data set (36speakers) [51] was formatted under a flexible design that has gathered information and samples from two distinct recording sessions and various speaking styles, in order to examine which factors allow the recognition of individual speakers from a closed set, aiming at gaining information from recorded signals that may occur during realistic scenarios and combine them appropriately. More technical information could be mined from the original paper. The general concept here is the discrimination among 36 different speakers, based on their recordings that were made through reading phrases and/or sentences under a comfortable rate, simulating a default speaking situation. We kept only the training set that was accumulated by speech signals based on 24 separate sentences that read from 36 speakers, whose origin affects their utterance (20 male and 16 female speakers from Ireland, U.K and U.S.). The features that were used are Mel-frequency cepstral coefficients (MFCCs).

[1] https://data.world/ml-research/gender-recognition-by-voice

TABLE 2. Standard benchmark data sets description.

Data set	#Examples	#Attributes	IR
abalone19	4174	10	129.44
dermatology-6	358	34	16.90
ecoli-0-1-4-6_vs_5	280	6	13.00
ecoli-0-1-4-7_vs_2-3-5-6	336	7	10.59
ecoli-0-1-4-7_vs_5-6	332	6	12.28
ecoli-0-2-3-4_vs_5	202	7	9.10
ecoli-0-3-4-6_vs_5	205	7	9.25
ecoli-0-3-4-7_vs_5-6	257	7	9.28
ecoli-0-3-4_vs_5	200	7	9.00
ecoli-0-4-6_vs_5	203	6	9.15
ecoli-0-6-7_vs_5	220	6	10.00
ecoli1	336	7	3.36
ecoli2	336	7	5.46
ecoli3	336	7	8.60
ecoli4	336	7	15.80
glass-0-1-4-6_vs_2	205	9	11.06
glass0	214	9	2.06
glass1	214	9	1.82
glass4	214	9	15.46
glass6	214	9	6.38
haberman	306	3	2.78
iris0	150	4	2.00
new-thyroid1	215	5	5.14
newthyroid2	215	5	5.14
page-blocks-1-3_vs_4	472	10	15.86
page-blocks0	5472	10	8.79
pima	768	8	1.87
poker-8_vs_6	1477	10	85.88
segment0	2308	19	6.02
vehicle0	846	18	3.25
vehicle1	846	18	2.90
vehicle2	846	18	2.88
vehicle3	846	18	2.99
winequality-red-4	1599	11	29.17
winequality-red-8_vs_6-7	855	11	46.50
wisconsin	683	9	1.86
yeast-0-2-5-6_vs_3-7-8-9	1004	8	9.14
yeast-0-2-5-7-9_vs_3-6-8	1004	8	9.14
yeast-0-3-5-9_vs_7-8	506	8	9.12
yeast-1-4-5-8_vs_7	693	8	22.10
yeast-1_vs_7	459	7	14.30
yeast-2_vs_4	514	8	9.08
yeast-2_vs_8	482	8	23.10
yeast5	1484	8	32.73

TABLE 3. Parameters of the compared methods.

Method	Parameters
ENN	n_neighbors=3
NCR	n_neighbors=3, threshold_cleaning=0.5
OSS	n_neighbors=1
RENN	n_neighbors=3, max_iter=100
SMOTE	n_neighbors=5
PROPOSED	q=200, r=0.5, t=1.0, k=5

Wearable Stress and Affect Detection data set (wesad) [52] was recently composed favoring the combination of both motion and physiological features into a common framework. At total, 15 different sensor modalities were measured, leading to 3 different states: stress, amusement and neutral. During the experimental phase, data from 12 subjects were obtained, including both male and female persons with specific age and health condition criteria, along with the ground-truth information submitting the participants into proper questionnaires. As it concerns the measurement phase, they attached two different devices to each individual: one wrist worn and one chest worn. Different window shift and window length parameters were applied depending on the modality that was tackled and its specific properties. More details could be found in the original demonstration.

While the second set of data sets contains forty-four standard benchmark data sets for imbalanced classification that were obtained from the KEEL-dataset repository [53].

B. DESIGN OF EXPERIMENTS

All data sets have been partitioned using the five-fold stratified cross-validation procedure. Stratification performed in order to maintain the class distribution across the train and the test folds. This procedure divides the instances in five stratified folds. Each tested method has been trained using four folds (training partition) and the fold left out (testing partition) has been used for evaluation of the final method. Then the average across all tested folds has been computed for the performance metric used. Area Under the ROC Curve (*AUC*) [49], [54] and Balanced Accuracy have been used as evaluation metrics. *AUC* supplies a scalar value to determine how well a classifier compensates its true positive (TP_{rate}) and false positive rates(FP_{rate}). An approximation [55] of this measure is given by

$$AUC = \frac{1 + TP_{rate} - FP_{rate}}{2} \tag{6}$$

Similarly, another common metric to measure classification performance under imbalanced data sets is Balanced Accuracy. Balanced Accuracy (*BACC*) is formulated as

$$BACC = \frac{\frac{TP}{P} + \frac{TN}{N}}{2} \tag{7}$$

where TP is the number of true positives, P is the number of positive examples, TN is the number of true negatives, N is the number of negative examples.

The proposed method is compared against six other under-sampling methods, without a resampling method

TABLE 4. Classification performance using AUC for multimodal data sets.

Data set	ENN	NCR	NONE	OSS	RENN	RUS	SMOTE	TL	PROPOSED
36speakers-man-vs-all	0.8789	0.8788	0.8783	0.8784	0.8790	0.8764	0.8789	0.8784	**0.8799**
36speakers-woman-vs-all	0.8722	0.8724	0.8705	0.8705	0.8725	0.8667	**0.8775**	0.8706	0.8768
anad-angry-vs-surprised	0.8270	0.8282	0.8231	0.8265	0.7990	0.8181	0.8085	0.8291	**0.8438**
anad-happy-vs-surprised	0.7872	0.7813	0.8049	0.8021	0.8057	0.8167	0.8052	0.8039	**0.8244**
voice	0.9223	0.9271	0.9402	0.9410	0.9138	0.9522	0.9507	0.9387	**0.9616**
wesad-eda-baseline-vs-amusement	0.5854	0.5871	0.5799	**0.5906**	0.5898	0.5657	0.5809	0.5811	0.5693
wesad-eda-baseline-vs-stress	0.7034	0.7041	0.7042	0.7041	0.7031	**0.7084**	0.7059	0.7041	0.7031

TABLE 5. Classification performance using balanced accuracy for multimodal data sets.

Data set	ENN	NCR	NONE	OSS	RENN	RUS	SMOTE	TL	PROPOSED
36speakers-man-vs-all	0.6337	0.6341	0.6328	0.6326	0.6343	0.7834	0.7864	0.6328	**0.7911**
36speakers-woman-vs-all	0.6017	0.6017	0.5998	0.6001	0.6017	0.7933	0.7971	0.5998	**0.8018**
anad-angry-vs-surprised	0.7561	0.7686	0.7673	0.7660	0.6974	0.7823	0.7303	0.7724	**0.7869**
anad-happy-vs-surprised	0.7658	0.7598	0.7876	0.7757	0.8014	0.8173	0.7971	0.7826	**0.8219**
voice	0.7708	0.7787	0.8073	0.8044	0.7738	0.8562	0.8494	0.8063	**0.9389**
wesad-eda-baseline-vs-amusement	0.5035	0.5035	0.5035	0.5035	0.5035	0.5287	0.5154	0.5035	**0.5848**
wesad-eda-baseline-vs-stress	0.5749	0.5745	0.5740	0.5740	0.5753	**0.6509**	0.6345	0.5740	0.6375

TABLE 6. Classification performance using AUC for standard data sets.

Data set	ENN	NCR	NONE	OSS	RENN	RUS	SMOTE	TL	PROPOSED
abalone19	0.6949	0.6945	0.6936	0.6952	0.695	**0.7075**	0.6794	0.6938	0.659
dermatology-6	0.966	0.966	0.966	0.8792	0.966	0.9289	0.9749	0.966	**0.9881**
ecoli-0-1-4-6_vs_5	0.8582	0.8582	0.8582	0.8534	0.8582	0.8606	0.862	0.8582	**0.8803**
ecoli-0-1-4-7_vs_2-3-5-6	0.9217	0.9244	0.9261	0.926	0.9146	0.9081	0.9272	0.925	**0.96**
ecoli-0-1-4-7_vs_5-6	0.9484	0.9484	0.949	0.945	0.9484	0.826	0.9433	0.949	**0.9575**
ecoli-0-2-3-4_vs_5	0.8561	0.8575	0.8616	0.8686	0.8561	0.8164	0.8739	0.863	**0.9**
ecoli-0-3-4-6_vs_5	0.8486	0.85	0.85	0.8527	0.8486	0.7662	0.8541	0.8486	**0.9216**
ecoli-0-3-4-7_vs_5-6	0.9254	0.9254	0.9237	0.9193	0.9254	0.928	0.9059	0.9254	**0.9413**
ecoli-0-3-4_vs_5	0.8354	0.8438	0.8438	0.8604	0.8354	0.75	0.8604	0.8438	**0.8944**
ecoli-0-4-6_vs_5	0.8456	0.8456	0.8456	0.8527	0.8456	0.8483	0.8566	0.8456	**0.8715**
ecoli-0-6-7_vs_5	0.8738	0.8788	0.8838	0.8713	0.8738	0.8375	0.8825	0.8825	**0.915**
ecoli1	0.85	0.8523	0.8468	0.8389	0.8534	0.8718	0.8421	0.8479	**0.8854**
ecoli2	0.9281	0.9285	0.9278	0.9277	0.9271	0.9211	**0.9417**	0.9278	0.9406
ecoli3	0.8903	0.8922	0.9063	0.9196	0.8727	**0.9226**	0.9082	0.9068	0.894
ecoli4	0.9155	0.9155	0.9202	0.9203	0.9155	0.9155	0.9337	0.9171	**0.9504**
glass-0-1-4-6_vs_2	0.685	0.6904	0.6948	0.6798	0.6985	0.6179	0.703	0.6961	**0.7201**
glass0	0.7713	0.8004	0.8024	0.8097	0.7551	0.7999	0.7925	0.8043	**0.826**
glass1	0.6626	0.6619	0.6596	0.6687	0.7015	**0.7015**	0.6365	0.6564	0.66
glass4	0.7233	0.7258	0.7275	0.7142	0.7225	0.76	0.7517	0.7275	**0.8297**
glass6	0.8537	0.8535	0.8537	0.8252	**0.8879**	0.855	0.8268	0.8564	0.864
haberman	0.6695	**0.6763**	0.6448	0.6578	0.6709	0.611	0.6448	0.6572	0.6562
iris0	**1.0**	**1.0**	**1.0**	0.9	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**
new-thyroid1	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**
newthyroid2	**1.0**	**1.0**	0.9992	**1.0**	**1.0**	0.9976	0.9952	0.9992	0.9992
page-blocks-1-3_vs_4	0.9019	0.9083	0.9087	0.9076	0.9019	0.9017	0.9087	0.9087	**0.9921**
page-blocks0	0.9361	0.9351	0.931	0.9275	0.937	0.9348	0.9322	0.9317	**0.9542**
pima	0.7989	0.809	**0.8171**	0.8145	0.7892	0.8135	0.8152	0.8134	0.8148
poker-8_vs_6	0.4366	0.4394	0.4376	0.4268	0.4368	0.4555	0.5003	0.4372	**0.5314**
segment0	0.9819	0.9819	0.9819	0.9822	0.9819	0.9815	0.9803	0.9819	**0.9873**
vehicle0	0.8055	0.8048	0.8114	0.8232	0.8013	0.8094	0.8196	0.8113	**0.9036**
vehicle1	0.7093	0.7129	0.713	0.717	0.7081	0.7187	0.7166	0.7133	**0.7397**
vehicle2	0.8611	0.8586	0.8579	0.85	0.8565	0.8335	0.8493	0.8571	**0.9202**
vehicle3	0.6996	0.7014	0.6991	0.6977	0.7004	0.6974	0.6994	0.6984	**0.7621**
winequality-red-4	0.6587	0.653	0.6482	0.6509	0.6603	0.626	0.6531	0.6498	**0.6941**
winequality-red-8_vs_6-7	0.7125	0.7169	0.7105	0.6693	**0.7205**	0.6736	0.6507	0.7132	0.6946
wisconsin	0.9745	0.9771	0.9833	**0.9933**	0.9737	0.9831	0.9834	0.9815	0.9779
yeast-0-2-5-6_vs_3-7-8-9	0.7613	0.762	0.7594	0.7623	0.7641	0.7416	0.7554	0.7599	**0.8162**
yeast-0-2-5-7-9_vs_3-6-8	0.916	0.9155	0.9132	0.8875	0.9162	0.9074	0.8157	0.915	**0.9322**
yeast-0-3-5-9_vs_7-8	0.6945	0.6899	0.6985	0.7121	0.6796	0.7125	**0.7312**	0.6978	0.721
yeast-1-4-5-8_vs_7	0.6553	0.6565	0.6576	0.6583	0.6482	0.6197	0.6579	0.6578	**0.6586**
yeast-1_vs_7	0.8023	0.8	0.8	0.7996	**0.805**	0.7922	0.7763	0.8	0.8005
yeast-2_vs_4	0.9004	0.9006	0.8985	0.8984	0.9017	**0.9211**	0.8843	0.8987	0.9204
yeast-2_vs_8	0.8353	0.8364	0.8359	0.8277	0.8348	**0.8504**	0.7934	0.8359	0.8378
yeast5	0.9865	0.9864	0.9861	0.9859	0.9865	0.9764	0.9816	0.9862	**0.9886**

(*NONE*) and against plain *SMOTE* which is part of the proposed approach. Specifically, the proposed method is compared with *Edited Nearest Neighbors* (*ENN*), *Neighbourhood Cleaning Rule* (*NCR*), *One Side Selection* (*OSS*), *Repeated Edited Nearest Neighbours* (*RENN*) [56], *Random Under Sampling* (*RUS*), *Synthetic Minority Over-sampling Technique* (*SMOTE*), *Tomek Links* (*TL*). As final classifier, the Naive Bayes classifier is used. In Table 3 the parameters for each compared method are presented which are the defaults in *imbalanced-learn*c [17] package[2]. *SMOTE* and *RUS* configured to achieve 1:1 ratio. All experiments were conducted in Python using the available implementations and utilities of *scikit-learn* [57], *imbalanced-learn* [17] and *modAL* [58]. Also, an implementation of the proposed approach is provided[3].

[2]TL does not take any parameters
[3]https://gitlab.com/chkoar/uunb

TABLE 7. Classification performance using balanced accuracy for standard data sets.

Data set	ENN	NCR	NONE	OSS	RENN	RUS	SMOTE	TL	PROPOSED
abalone19	0.5015	0.502	0.5083	0.5173	0.5167	0.678	**0.7012**	0.5074	0.6153
dermatology-6	0.9587	0.9587	0.9587	0.863	0.9587	0.9186	0.966	0.9587	**0.9793**
ecoli-0-1-4-6_vs_5	0.7962	0.7962	0.8712	0.7981	0.7962	0.5865	**0.8962**	0.7962	0.8
ecoli-0-1-4-7_vs_2-3-5-6	0.6467	0.6467	0.6467	0.6467	0.6467	0.5317	**0.6801**	0.6467	0.6467
ecoli-0-1-4-7_vs_5-6	0.6167	0.6167	0.7685	0.7967	0.6167	0.603	**0.882**	0.7767	0.6167
ecoli-0-2-3-4_vs_5	0.5333	0.5625	0.5573	0.533	0.5333	0.5194	0.6777	0.5573	**0.6778**
ecoli-0-3-4-6_vs_5	0.5331	0.5547	0.5547	0.5331	0.5331	0.5426	0.6872	0.5547	**0.6872**
ecoli-0-3-4-7_vs_5-6	0.6135	0.5157	0.5794	0.6972	0.6135	**0.787**	0.6421	0.5794	0.6135
ecoli-0-3-4_vs_5	0.5333	0.5611	0.5556	0.525	0.5333	0.4833	**0.6722**	0.5583	0.6694
ecoli-0-4-6_vs_5	0.789	0.7917	0.7917	0.7919	0.789	0.778	**0.7974**	0.7917	0.7656
ecoli-0-6-7_vs_5	0.815	0.805	0.8125	0.79	0.815	0.5875	**0.8475**	0.81	0.7475
ecoli1	0.7156	0.719	0.7027	0.695	0.7337	**0.8319**	0.7519	0.7046	0.617
ecoli2	0.5976	0.5727	0.5744	0.5569	0.5976	**0.8571**	0.6682	0.5744	0.5797
ecoli3	0.8041	0.7914	0.8046	0.8139	0.5908	0.7198	**0.8633**	0.8029	0.6955
ecoli4	0.8767	0.8767	0.8767	0.8751	0.8529	0.8638	**0.8862**	0.8767	0.8726
glass-0-1-4-6_vs_2	0.5811	0.5731	0.5614	0.564	0.5785	0.4896	0.5442	0.5864	**0.6196**
glass0	0.6984	0.6984	0.7014	0.705	0.6915	**0.7053**	0.6871	0.6979	0.6765
glass1	0.6318	0.6477	0.6339	0.6594	0.6446	0.6632	0.6501	**0.6697**	0.6612
glass4	0.6333	0.5908	0.5075	0.5742	0.7633	0.7767	0.7483	0.5075	**0.8067**
glass6	0.873	0.8595	0.8757	0.8703	**0.9063**	0.8761	0.8757	0.8757	0.8896
haberman	0.6435	0.631	0.5645	0.6057	**0.6558**	0.6032	0.618	0.6057	0.6334
iris0	**1.0**	**1.0**	**1.0**	0.9	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**
new-thyroid1	0.9833	0.9806	0.9833	**0.9972**	0.9833	0.9778	0.975	0.9833	0.9833
newthyroid2	0.9778	0.9778	0.9663	0.9774	0.9778	0.9667	0.9778	0.9663	**0.9917**
page-blocks-1-3_vs_4	0.7538	0.764	0.764	0.7651	0.7938	0.7835	0.8017	0.764	**0.9686**
page-blocks0	0.7525	0.7532	0.7044	0.6949	0.7747	0.7303	0.7446	0.7105	**0.8044**
pima	0.7167	0.7259	0.713	0.7195	0.7118	**0.7392**	0.7342	0.7183	0.7381
poker-8_vs_6	0.5	0.5	0.5	0.5	0.5	0.5063	**0.5775**	0.5	0.563
segment0	0.8954	0.8954	0.8962	0.8969	0.8952	0.8843	0.8916	0.8962	**0.9386**
vehicle0	0.7333	0.7383	0.7198	0.7337	0.7404	0.7496	0.758	0.7198	**0.8288**
vehicle1	0.6717	0.6756	0.6805	0.678	0.6763	0.6779	0.6793	0.6733	**0.6839**
vehicle2	0.7458	0.7313	0.7271	0.7245	0.7527	0.753	0.7621	0.724	**0.8864**
vehicle3	0.6717	0.6685	0.6758	0.6749	0.6748	0.6668	0.6692	0.6734	**0.744**
winequality-red-4	0.5421	0.5421	0.5175	0.5178	0.5499	0.6313	0.6241	0.5172	**0.6488**
winequality-red-8_vs_6-7	0.569	0.5678	0.5488	0.5494	0.5666	**0.6843**	0.6351	0.5476	0.6259
wisconsin	0.9598	0.962	0.9617	0.9446	0.9564	0.9625	**0.9627**	0.9614	0.9601
yeast-0-2-5-6_vs_3-7-8-9	0.5337	0.5348	0.5493	0.5493	0.5287	0.5008	**0.6632**	0.5443	0.6341
yeast-0-2-5-7-9_vs_3-6-8	**0.8863**	0.8485	0.5452	0.6137	0.8858	0.8732	0.4967	0.6225	0.816
yeast-0-3-5-9_vs_7-8	0.6023	0.5345	0.5652	0.5663	0.587	0.5556	0.5417	0.5663	**0.6957**
yeast-1-4-5-8_vs_7	0.5354	0.5354	0.5346	0.5316	0.5362	**0.5467**	0.5209	0.5346	0.5241
yeast-1_vs_7	0.5588	0.5588	0.5577	0.5565	0.5577	**0.6473**	0.5821	0.5577	0.6171
yeast-2_vs_4	0.7382	0.6589	0.5777	0.7349	0.7349	0.6963	0.5987	0.5788	**0.7682**
yeast-2_vs_8	**0.7739**	**0.7739**	0.6825	0.7728	**0.7739**	0.7685	0.6329	**0.7739**	0.7395
yeast5	0.8049	0.8045	0.8045	0.7753	0.8052	0.8028	0.858	0.8049	**0.9003**

C. RESULTS

In Tables 4,6 and Tables 5,7 the cross-validated results for each method are presented in terms of *AUC* and Balanced Accuracy, respectively. According to Demšar [59] non-parametric tests should be preferred instead of parametric ones in the context of machine learning problems, since they do not assume normal distributions or homogeneity of variance, especially when the number of the test cases is low. Thus, in order to validate the significance of the results, the Friedman test [59], which is a rank-based non-parametric test for comparing several machine learning algorithms on multiple data sets, has been used, having as control method the proposed approach. The null hypothesis of the test states that all the algorithms perform equivalently and therefore their ranks should be equal. The average rankings as well as the results of the Friedman test are presented in Tables 8 and 9.

TABLE 8. Friedman tests and average rankings for multimodal data sets.

Method	AUC	Balanced Accuracy
PROPOSED	3.37143	2.08571
SMOTE	4.88571	3.72857
OSS	4.91429	6.27143
TL	5.08571	5.91429
NCR	5.14286	6
ENN	5.17143	6.57143
NONE	5.34286	5.84286
RENN	5.35714	6.1
RUS	5.72857	2.48571
Statistic	2.10985	23.1004
p-value	0.03509	$<10^{-6}$

According to the average ranks the proposed approach tends to generate data sets that will produce better classifiers. The p-values of all Friedman tests indicate that the null hypothesis has to be rejected. So, there is at least

TABLE 9. Friedman tests and average rankings for standard data sets.

Method	AUC	Balanced Accuracy
PROPOSED	3.702	3.825
SMOTE	4.968	4.03
OSS	5.214	5.482
TL	4.948	5.291
NCR	5.039	5.35
ENN	5.339	5.32
NONE	4.998	5.452
RENN	5.275	5.173
RUS	5.518	5.077
Statistic	8.311	11.95
p-value	$<10^{-6}$	$<10^{-6}$

TABLE 10. Post hoc analysis for multimodal data sets with Holm's procedure using proposed as control method for AUC.

Comparison	Statistic	Adj.p-value	Result
PROSED VS RUS	3.60060	0.00254	H_0 rejected
PROSED VS RENN	3.03323	0.01694	H_0 rejected
PROSED VS NONE	3.01141	0.01694	H_0 rejected
PROSED VS ENN	2.74955	0.02984	H_0 rejected
PROSED VS NCR	2.70590	0.02984	H_0 rejected
PROSED VS TL	2.61861	0.02984	H_0 rejected
PROSED VS OSS	2.35675	0.03687	H_0 rejected
PROSED VS SMOTE	2.31311	0.03687	H_0 rejected

TABLE 11. Post hoc analysis for multimodal data sets with Holm's procedure using proposed as control method for balanced accuracy.

Comparison	Statistic	Adj. p-value	Result
PROPOSED VS ENN	6.85204	$<10^{-6}$	H_0 rejected
PROPOSED VS OSS	6.39378	$<10^{-6}$	H_0 rejected
PROPOSED VS RENN	6.13192	$<10^{-6}$	H_0 rejected
PROPOSED VS NCR	5.97917	$<10^{-6}$	H_0 rejected
PROPOSED VS TL	5.84824	$<10^{-6}$	H_0 rejected
PROPOSED VS NONE	5.73913	$<10^{-6}$	H_0 rejected
PROPOSED VS SMOTE	2.50951	0.02418	H_0 rejected
PROPOSED VS RUS	0.61101	0.54119	H_0 not rejected

TABLE 12. Post hoc analysis for standard data sets with Holm's procedure using proposed as control method for AUC.

Comparison	Statistic	Adj. p-value	Result
PROPOSED VS RUS	6.95440	$<10^{-6}$	H_0 is rejected
PROPOSED VS ENN	6.26680	$<10^{-6}$	H_0 is rejected
PROPOSED VS RENN	6.02309	$<10^{-6}$	H_0 is rejected
PROPOSED VS OSS	5.78808	$<10^{-6}$	H_0 is rejected
PROPOSED VS NCR	5.11788	$<10^{-6}$	H_0 is rejected
PROPOSED VS NONE	4.96121	$<10^{-6}$	H_0 is rejected
PROPOSED VS SMOTE	4.84806	$<10^{-6}$	H_0 is rejected
PROPOSED VS TL	4.76973	$<10^{-6}$	H_0 is rejected

one method that performs significantly different from the proposed method. With the intention of investigating the aforementioned, the post hoc procedure that is proposed by Holm [60] is used. The p-values in Tables 10 and 11 nominate that the proposed method is the best performing method in terms of *AUC*, while in terms of Balanced Accuracy archives similar performance to *RUS* for the multimodal data sets. It must be noted that even in that case, the proposed approach produces smaller training set that *RUS*. Similarly, for the standard benchmark data sets, Tables 12 and 13 indicate

TABLE 13. Post hoc analysis for standard data sets with Holm's procedure using proposed as control method for balanced accuracy.

Comparison	Statistic	Adj. p-value	Result
PROPOSED VS OSS	6.34513	$<10^{-6}$	H_0 is rejected
PROPOSED VS NONE	6.23198	$<10^{-6}$	H_0 is rejected
PROPOSED VS NCR	5.84031	$<10^{-6}$	H_0 is rejected
PROPOSED VS ENN	5.72715	$<10^{-6}$	H_0 is rejected
PROPOSED VS TL	5.61400	$<10^{-6}$	H_0 is rejected
PROPOSED VS RENN	5.16140	$<10^{-6}$	H_0 is rejected
PROPOSED VS RUS	4.79584	$<10^{-6}$	H_0 is rejected
PROPOSED VS SMOTE	0.78335	0.43342	H_0 is not rejected

that the proposed approach produces data sets that make the classifiers perform significantly better, compared to the other tested data level methods, in terms of *AUC* while in terms of Balanced Accuracy the proposed approach archives similar performance to *SMOTE*.

V. CONCLUSION AND FUTURE WORK

In this research work, a data level method that copes with imbalanced data sets in classification tasks has been presented. Experiments on several multimodal and standard benchmark data sets show that the proposed method can handle imbalanced data sets and significantly improve the classification performance of Naive Bayes classifier in contrast to other approaches in terms of *AUC* and Balanced Accuracy. Although Naive Bayes algorithm could be seen as a weak probabilistic learner, its assets matched with the context of this work: i) it has been proven that in real cases Naive Bayes handles imbalanced data sets with favoring manner, being robust enough even to more degenerate situations of imbalanced problems [61], ii) its performance based on its structure and its assumptions export decisions without spending much temporal or spatial resources.

This kind of approaches could also be generalized and integrated into other similar tasks, such as the concept of Active Learning, where only a small initial labeled set exists and a larger pool of unlabeled data is available for mining its instances appropriately. By using our active example selection method, we could select unlabeled instances that satisfy more dedicated criteria, like trade-offs between discriminative and representative properties [62], [63]. Moreover, different learning models should also be studied, as has been done in the literature, concerning the base classifier into the core of the scheme that tackles with the stage of balancing the instances and discriminate their efficacy on more detailed scientific domains [64] or even try to generalize also our method on extremely imbalanced cases [65]. Imbalanced problems are also found in the case of Online Learning, where more realistic scenarios take place. A well-known field of interest is related with client credit assessment, where the assumption of even distribution is not accurate, since concept drift is the usual phenomenon that puts obstacles on ML approaches [66]. Although ensembles of learners usually achieve great accuracy, Naive Bayes and/or algorithms that are generated from similar approaches

and belong to Bayesian learning family, have not been rigorously reviewed [67], [68].

REFERENCES

[1] R. C. Prati, G. E. Batista, and M. C. Monard, "Data mining with imbalanced class distributions: Concepts and methods," in *Proc. IICAI*, 2009.

[2] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009, doi: 10.1109/tkde.2008.239.

[3] A. Fernández, S. del Río, N. V. Chawla, and F. Herrera, "An insight into imbalanced big data classification: Outcomes and challenges," *Complex Intell. Syst.*, vol. 3, no. 2, pp. 105–120, Mar. 2017.

[4] B. Krawczyk, "Learning from imbalanced data: Open challenges and future directions," *Prog. Artif. Intell.*, vol. 5, no. 4, pp. 221–232, Apr. 2016.

[5] Q. Yang and X. Wu, "10 challenging problems in data mining research," *Int. J. Inf. Technol. Decis. Making*, vol. 5, no. 4, pp. 597–604, Dec. 2006, doi: 10.1142/s0219622006002258.

[6] M. Rastgoo, G. Lemaitre, J. Massich, O. Morel, F. Marzani, R. Garcia, and F. Meriaudeau, "Tackling the problem of data imbalancing for melanoma classification," in *Proc. Int. Joint Conf. Biomed. Eng. Syst. Technol. (BIOSTEC)*, 2016, pp. 32–39.

[7] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Handling imbalanced datasets: A review," *GESTS Int. Trans. Comput. Sci. Eng.*, vol. 30, no. 1, pp. 25–36, 2006.

[8] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 4, pp. 463–484, Jul. 2012, doi: 10.1109/tsmcc.2011.2161285.

[9] D. Tomar and S. Agarwal, "Prediction of defective software modules using class imbalance learning," *Appl. Comput. Int. Soft Comput.*, vol. 2016, pp. 7658207-1–7658207-12, 2016, doi: 10.1155/2016/7658207.

[10] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, and G. Bing, "Learning from class-imbalanced data: Review of methods and applications," *Expert Syst. Appl.*, vol. 73, pp. 220–239, May 2017, doi: 10.1016/j.eswa.2016.12.035.

[11] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," in *Proc. ECML*, 2004.

[12] W. Zong, G.-B. Huang, and L. Chen, "Weighted extreme learning machine for imbalance learning," *Neurocomputing*, vol. 101, pp. 229–242, Feb. 2013.

[13] P. Domingos, "MetaCost," in *Proc. 5th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 1999.

[14] V. López, A. Fernández, J. G. Moreno-Torres, and F. Herrera, "Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics," *Expert Syst. Appl.*, vol. 39, no. 7, pp. 6585–6608, Jun. 2012.

[15] X.-Y. Liu and Z.-H. Zhou, "The influence of class imbalance on cost-sensitive learning: An empirical study," in *Proc. 6th Int. Conf. Data Mining (ICDM)*, Washington, DC, USA, 2006, pp. 970–974.

[16] Y. Sun, "Cost-sensitive boosting for classification of imbalanced data," Ph.D. dissertation, Waterloo, ON, Canada, 2007.

[17] G. Lemaître, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning," *J. Mach. Learn. Res.*, vol. 18, no. 17, pp. 1–5, 2017. [Online]. Available: http://jmlr.org/papers/v18/16-365.html

[18] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-2, no. 3, pp. 408–421, Jul. 1972, doi: 10.1109/tsmc.1972.4309137.

[19] I. Tomek, "Two modifications of CNN," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, no. 11, pp. 769–772, Nov. 1976, doi: 10.1109/tsmc.1976.4309452.

[20] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: One-sided selection," in *Proc. ICML*, 1997.

[21] J. Laurikkala, "Improving identification of difficult small classes by balancing class distribution," in *Artificial Intelligence in Medicine*. Berlin, Germany: Springer, 2001, pp. 63–66.

[22] J. Zhang and I. Mani, "KNN approach to unbalanced data distributions: A case study involving information extraction," in *Proc. Workshop Learn. Imbalanced Datasets (ICML)*, 2003.

[23] Y. Hou, B. Li, L. Li, and J. Liu, "A density-based under-sampling algorithm for imbalance classification," *J. Phys., Conf. Ser.*, vol. 1302, Aug. 2019, Art. no. 022064, doi: 10.1088/1742-6596/1302/2/022064.

[24] C.-F. Tsai, W.-C. Lin, Y.-H. Hu, and G.-T. Yao, "Under-sampling class imbalanced datasets by combining clustering analysis and instance selection," *Inf. Sci.*, vol. 477, pp. 47–54, Mar. 2019, doi: 10.1016/j.ins.2018.10.029.

[25] N. Ofek, L. Rokach, R. Stern, and A. Shabtai, "Fast-CBUS: A fast clustering-based undersampling method for addressing the class imbalance problem," *Neurocomputing*, vol. 243, pp. 88–102, Jun. 2017, doi: 10.1016/j.neucom.2017.03.011.

[26] M. S. Lee, J.-K. Rhee, B.-H. Kim, and B.-T. Zhang, "AESNB: Active example selection with Naïve Bayes classifier for learning from imbalanced biomedical data," in *Proc. 9th IEEE Int. Conf. Bioinf. BioEng.*, Jun. 2009, pp. 15–21.

[27] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[28] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "Safe-level-SMOTE: Safe-level-synthetic minority over-sampling TEchnique for handling the class imbalanced problem," in *Advances in Knowledge Discovery and Data Mining*. Berlin, Germany: Springer, 2009, pp. 475–482.

[29] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning," in *Advances in Intelligent Computing* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2005, pp. 878–887.

[30] S. Hu, Y. Liang, L. Ma, and Y. He, "MSMOTE: Improving classification performance when training data is imbalanced," in *Proc. IEEE 2nd Int. Workshop Comput. Sci. Eng.*, Oct. 2009, pp. 13–17.

[31] F. Koto, "SMOTE-out, SMOTE-cosine, and selected-SMOTE: An enhancement strategy to handle imbalance in data level," in *Proc. IEEE Int. Conf. Adv. Comput. Sci. Inf. Syst.*, Oct. 2014, pp. 280–284.

[32] J. A. Sáez, J. Luengo, J. Stefanowski, and F. Herrera, "SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering," *Inf. Sci.*, vol. 291, pp. 184–203, Jan. 2015, doi: 10.1016/j.ins.2014.08.051.

[33] S. Sharifirad, A. Nazari, and M. Ghatee, "Modified SMOTE using mutual information and different sorts of entropies," 2018, *arXiv:1803.11002*. [Online]. Available: https://arxiv.org/abs/1803.11002

[34] J. Stefanowski and S. Wilk, "Selective pre-processing of imbalanced data for improving classification performance," in *Data Warehousing and Knowledge Discovery*. Berlin, Germany: Springer, 2008, pp. 283–292, doi: 10.1007/978-3-540-85836-2_27.

[35] G. E. Batista, R. C. Prati, and M. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explor. Newslett.*, vol. 6, no. 1, pp. 20–29, Jun. 2004, doi: 10.1145/1007730.1007735.

[36] E. Ramentol, Y. Caballero, R. Bello, and F. Herrera, "SMOTE-RSB: A hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using SMOTE and rough sets theory," *Knowl. Inf. Syst.*, vol. 33, no. 2, pp. 245–265, Dec. 2011, doi: 10.1007/s10115-011-0465-6.

[37] A. Fernández, S. Garcia, F. Herrera, and N. V. Chawla, "SMOTE for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary," *J. Artif. Intell. Res.*, vol. 61, pp. 863–905, Apr. 2018, doi: 10.1613/jair.1.11192.

[38] A. Onan, "Consensus clustering-based undersampling approach to imbalanced learning," *Sci. Program.*, vol. 2019, pp. 1–14, Mar. 2019, doi: 10.1155/2019/5901087.

[39] I. Rish, "An empirical study of the Naïve Bayes classifier," in *Proc. Work Empir Methods Artif. Intell. (IJCAI)*, vol. 3, Jan. 2001.

[40] H. Zhang, "The optimality of Naïve Bayes," in *Proc. 17th Int. Florida Artif. Intell. Res. Soc. Conf.*, V. Barr and Z. Markov, Eds. Miami Beach, FL, USA: AAAI Press, 2004.

[41] M. J. Sánchez-Franco, A. Navarro-García, and F. J. Rondán-Cataluña, "A Naïve bayes strategy for classifying customer satisfaction: A study based on online reviews of hospitality services," *J. Bus. Res.*, vol. 101, pp. 499–506, Aug. 2019, doi: 10.1016/j.jbusres.2018.12.051.

[42] N. Sebe, M. Lew, I. Cohen, A. Garg, and T. Huang, "Emotion recognition using a cauchy Naïve Bayes classifier," in *Proc. Object Recognit. Supported User Interaction Service Robots*, 2002, doi: 10.1109/icpr.2002.1044578.

[43] S. Xu, "Bayesian Naïve Bayes classifiers to text classification," *J. Inf. Sci.*, vol. 44, no. 1, pp. 48–59, Nov. 2016, doi: 10.1177/0165551516677946.

[44] A. Fadlil, I. Riadi, and S. Aji, "A novel ddos attack detection based on Gaussian Naïve Bayes," *Bull. Elect. Eng. Informat.*, vol. 6, no. 2, pp. 140–148, 2017.

[45] S. E. Pandarakone, S. Gunasekaran, Y. Mizuno, and H. Nakamura, "Application of Naïve Bayes classifier theorem in detecting induction motor bearing failure," in *Proc. 13th Int. Conf. Elect. Mach. (ICEM)*, Sep. 2018, doi: 10.1109/icelmach.2018.8506836.
[46] U. Widodo Wijayanto and R. Sarno, "An experimental study of supervised sentiment analysis using Gaussian Naïve Bayes," in *Proc. Int. Seminar Appl. Technol. Inf. Commun.*, Sep. 2018, pp. 476–481.
[47] B. M. Gayathri and C. P. Sumathi, "An automated technique using Gaussian Naïve Bayes classifier to classify breast cancer," *Int. J. Comput. Appl.*, vol. 148, no. 6, pp. 16–21, Aug. 2016. [Online]. Available: http://www.ijcaonline.org/archives/volume148/number6/25761-2016911146
[48] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, "The balanced accuracy and its posterior distribution," in *Proc. IEEE 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 3121–3124, doi: 10.1109/icpr.2010.764.
[49] J. D. Kelleher, *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*. Cambridge, MA, USA: MIT Press, 2015.
[50] S. Klaylat. (2018). *Arabic Natural Audio Dataset*. [Online]. Available: https://data.mendeley.com/datasets/xm232yxf7t/1
[51] M. Grimaldi and F. Cummins, "Speaker identification using instantaneous frequencies," *IEEE Trans. Audio, Speech, Language Process.*, vol. 16, no. 6, pp. 1097–1111, Aug. 2008, doi: 10.1109/tasl.2008.2001109.
[52] P. Schmidt, A. Reiss, R. Duerichen, C. Marberger, and K. Van Laerhoven, "Introducing WESAD, a multimodal dataset for wearable stress and affect detection," in *Proc. Int. Conf. Multimodal Interact. (ICMI)*, 2018, doi: 10.1145/3242969.3242985.
[53] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *J. Multiple-Valued Logic Soft Comput.*, vol. 17, nos. 2–3, pp. 255–287, 2011.
[54] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006, doi: 10.1016/j.patrec.2005.10.010.
[55] V. López, A. Fernández, S. García, V. Palade, and F. Herrera, "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics," *Inf. Sci.*, vol. 250, pp. 113–141, Nov. 2013, doi: 10.1016/j.ins.2013.07.007.
[56] I. Tomek, "An experiment with the edited nearest-neighbor rule," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, no. 6, pp. 448–452, Jun. 1976, doi: 10.1109/tsmc.1976.4309523.
[57] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011. [Online]. Available: http://www.jmlr.org/papers/v12/pedregosa11a.html
[58] T. Danka and P. Horvath, "modAL: A modular active learning framework for Python," 2018, *arXiv:1805.00979*. [Online]. Available: https://arxiv.org/abs/1805.00979
[59] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1248547.1248548
[60] S. Holm, "A simple sequentially rejective multiple test procedure," *Scandin. J. Statist.*, vol. 6, no. 2, pp. 65–70, 1979. [Online]. Available: http://www.jstor.org/stable/4615733
[61] A. Somasundaram and S. Reddy, "Parallel and incremental credit card fraud detection model to handle concept drift and data imbalance," *Neural Comput. Appl.*, vol. 31, pp. 3–14, Jul. 2019, doi: 10.1007/s00521-018-3633-8.
[62] S.-J. Huang, R. Jin, and Z.-H. Zhou, "Active learning by querying informative and representative examples," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 1936–1949, Oct. 2014, doi: 10.1109/tpami.2014.2307881.
[63] B. Du, Z. Wang, L. Zhang, L. Zhang, W. Liu, J. Shen, and D. Tao, "Exploring representativeness and informativeness for active learning," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 14–26, Jan. 2017, doi: 10.1109/tcyb.2015.2496974.
[64] J. Bektas, T. Ibrikci, and I. T. Ozcan, "Classification of real imbalanced cardiovascular data using feature selection and sampling methods: A case study with neural networks and logistic regression," *Int. J. Artif. Intell. Tools*, vol. 26, no. 6, Dec. 2017, Art. no. 1750019, doi: 10.1142/s0218213017500191.
[65] D. N. Sotiropoulos and G. A. Tsihrintzis, "Artificial immune system-based classification in extremely imbalanced classification problems," *Int. J. Artif. Intell. Tools*, vol. 26, no. 3, Jan. 2017, Art. no. 1750009, doi: 10.1142/s0218213017500099.
[66] H. Zhang and Q. Liu, "Online learning method for drift and imbalance problem in client credit assessment," *Symmetry*, vol. 11, no. 7, p. 890, Jul. 2019, doi: 10.3390/sym11070890.
[67] L. Jiang, H. Zhang, and Z. Cai, "A novel Bayes model: Hidden Naïve Bayes," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 10, pp. 1361–1371, Oct. 2009, doi: 10.1109/tkde.2008.234.
[68] M. J. Flores and J. A. Gámez, and A. M. Martínez, "Domains of competence of the semi-Naïve Bayesian network classifiers," *Inf. Sci.*, vol. 260, pp. 120–148, Mar. 2014, doi: 10.1016/j.ins.2013.10.007.

CHRISTOS K. ARIDAS received the B.S. degree in statistics from the Athens University of Economics and Business, Athens, Greece, in 2011, and the M.S. degree in information systems from Hellenic Open University, Patras, Greece, in 2015. He is currently pursuing the Ph.D. degree in machine learning with the University of Patras, Patras. From 2016 to 2017, he was a Teaching Assistant with the Department of Mathematics, University of Patras. He is also a Machine Learning Engineer of Code4Thought P.C., where he mainly develops natural language processing applications. He is the author/coauthor of several articles published in international conferences and journals. His current research interests include machine learning, data mining, natural language processing, and metaheuristics.

STAMATIS KARLOS was born in Tripolis, Greece, in 1988. He received the Diploma degree from the Department of Electrical and Computer Engineering, University of Patras, in 2011, and the master's degree in mathematics and artificial intelligence from the Department of Mathematics University of Patras, in 2015. He is currently pursuing the Ph.D. degree with the fields of mathematics and computer science, Department of Mathematics, University of Patras, related with partial supervised algorithms applied on several domains of machine learning, mainly through semi-supervised learning and active learning approaches. Since 2015, he has been working as an Adjunct Assistant Professor with the Western Greece University of Applied Sciences and a Teacher Assistant with the Department of Mathematics for undergraduate courses. He is also a Data Scientist in companies related with Digital and Computational Advertising or Maritime Analytics has been held for more than a year. He is the author/coauthor of more than 30 research works, published on international conferences and journals. His awards and honors include three fellowships, oriented towards supporting his studies as post-graduate student (State Scholarships Foundation-IKY). His research interests during Ph.D. studies (H.F.R.I.) and his innovative research interests under related team proposals funded by both European Union's and national funds. He also received the Best Student Paper Award in IISA conference, supported by IEEE, in 2016 and 2019, and the corresponding award during PCI conference of 2016.

VASILEIOS G. KANAS was born in Patras, Greece, in 1987. He received the Ph.D. degree in electrical and computer engineering from the University of Patras, Patras, Greece, in 2017, with a focus on brain signal and image analysis using machine learning methods. He has participated as a Researcher in several EU RD projects as a data analyst, performing data mining and data modeling. He has also been involved in digital image and signal processing. Since 2015, he has been a Senior Software Engineer of Yodiwo AE, designing and developing software and applying machine learning algorithms. He is also involved in developing tools for Yodiwo's IoT Cloud Platform and dealing with all UIs, including mobile apps, web apps, and dashboards. His awards and honors include three fellowships (funded by both European Union's and national funds) to support his Ph.D. studies and research interests in several laboratories, such as the Neurocomputing and Neurorobotics Research Group, Universidad Complutense de Madridand, Madrid, Spain, the Center for Research and Applications of Nonlinear Systems, University of Patras, the National Center For Scientific Research ''Demokritos'', Athens, Greece, the Multidimensional Data Analysis and Knowledge Management Laboratory, University of Patras, and so on. He received the Best Student Paper Award in IISA conference, supported by IEEE, in 2019.

NIKOS FAZAKIS was born in Patras, Greece, in 1988. He received the Diploma degree in electrical and computer engineering (ECE) and the master's degree in business administration from the University of Patras, where he is currently pursuing the Ph.D. degree in electrical engineering. From 2015 to 2019, he was a Researcher with the Wired Communications Laboratory (WCL), Department of ECE, University of Patras, where he participated in numerous European and National research programs. He has a variety of publications in the fields of machine learning and data mining. He was honored with a number of best paper awards in various conferences in the field of artificial intelligence.

SOTIRIS B. KOTSIANTIS received the master's and Ph.D. degrees in computer science from the University of Patras, Greece. He is currently an Assistant Professor with the Department of Mathematics, University of Patras, Greece. He is also a Mathematician. He has a lot of publications with numerous citations. His research interests are in the field of data mining, machine learning, and learning analytics.