Combining *Random Subspace* Approach with SMOTE Oversampling for Imbalanced Data Classification

Pawel Ksieniewicz(✉)

Wrocław University of Science and Technology, Wrocław, Poland
pawel.ksieniewicz@pwr.edu.pl

Abstract. Following work tries to utilize a hybrid approach of combining *Random Subspace* method and SMOTE oversampling to solve a problem of imbalanced data classification. Paper contains a proposition of the ensemble diversified using Random Subspace approach, trained with a set oversampled in the context of each reduced subset of features. Algorithm was evaluated on the basis of the computer experiments carried out on the benchmark datasets and three different base classifiers.

Keywords: Imbalanced classification · SMOTE · Random Subspace · Classifier ensembles

1 Introduction

A major part of the pattern recognition problems presents the task of classification, in which we train a model capable of assigning new, unknown objects to predefined groups on the basis of a knowledge extracted from a set of labeled patterns [5]. Most classical classification algorithms assume an equal percentage of each class and encounters a problem when the proportions between them are strongly disturbed, tending to favor prediction of the more common one. Data about this characteristic is called *imbalanced data* [22]. Most of the real problems, such as diagnosis of diseases, SPAM-detection or fraud recognition, require detection of events far from normal and therefore are not balanced, which makes necessary to modify the pattern recognition models for their needs [10].

In order to eliminate this problem and to construct a model capable of classifying imbalanced data, three approaches are most commonly used [13]. The first of these are methods of data pre-processing, in which we do not modify the learning process, but we introduce changes in the training set itself. The simplest examples are *random undersampling*, in which we take into training set the full minority class and a random subset of the same size from majority class and *random oversampling*, where we use the full majority class and randomly selected objects of the minority class of the same cardinality, regardless of the repetition of the patterns [6].

H. Pérez García et al. (Eds.): HAIS 2019, LNAI 11734, pp. 660–673, 2019.
https://doi.org/10.1007/978-3-030-29859-3_56

More complex solutions of this type are *oversampling* algorithms, which instead of repeating existing minority class patterns, generate new synthetic objects based on the information contained in their distribution. The most common of them are ADASYN [9] and SMOTE [3], developed into a multitude that also takes into account the distribution of the majority class of varieties such as *Borderline*-SMOTE [8], *Safe-Level*-SMOTE [2] or LN-SMOTE [18].

Another approach is to use *inbuilt mechanisms* into the classifier learning process itself. The most commonly used is the one-class classification, insensitive to the distribution of classes of the problem [14] and the cost-sensitive classification that takes into account the loss-function asymmetric in favor of the minority class [10].

The final group of considered approaches are hybrid solutions that combine preprocessing methods with classifier ensembles. The modifications of *Bagging* and *Boosting* are the most popular [20], but there are also methods based on combining a team of classifiers built on the basis of various methods of oversampling or random split undersampling [15].

An important factor that we must take into consideration during the experiments on imbalanced data is also the metric used to assess the quality of constructed models [12]. Typical accuracy, in a strongly imbalanced problem, gives us results being far from the truth, showing, for example, 90% accuracy when wrongly classifying the entire minority class occurring in one in ten samples. In binary classification problems, therefore, the most-used are taking into account the proportions of the measurement classes F-measure or geometric mean score. In multi-class problems, where the above measures can not be calculated, we use the most often the balanced accuracy score.

Following paper attempts to propose a new hybrid method, based on classifier ensembles built in accordance with the *Random Subspace* [26] principle common for multidimensional data and the SMOTE algorithm used for oversampling of objects in the subspaces of each of the member classifiers [24]. Previous studies have already used a combination of these methods, but *oversampling* is performed there before application of *Random Subspace*, which may not properly use the profits achieved by finding a subspace that allows effective determination of the decision boundary [11].

The main contributions of this work are:

- Proposition of the method of joint use of weighted classifier ensemble obtained by the *Random Subspace* method with the use of SMOTE oversampling.
- Implementation of the proposed method in varieties taking into account the separate use of each of its elements.
- Experimental evaluation of the impact of SMOTE, *Random Subspace* and weighing the ensemble fuser on the quality of imbalanced data classification.

2 Method Design

Random Subspace. The construction of the classifier ensemble gives us two basic difficulties [25]. The first is to provide a diverse pool of classifiers that allow to perform independent, parallel prediction. We can achieve this by using different classification algorithms or various subsets of the training set. The proposed method uses the second approach, where each member of the committee is built on a random subspace of the training set.

SMOTE. Before training each member of the ensemble, minority oversampling is performed using the basic version of the SMOTE algorithm. Training takes place using the base classifier chosen by the experimenter, which must, however, be a probabilistic classifier, or at least have probabilistic interpretation.

Fuser. The second difficulty before the effective construction of the classifier ensemble is the construction of its *fuser* – the function responsible for making decisions on behalf of the ensemble based on the opinions of its members [16]. Among the concepts used, according to names popular in implementations, there are *hard fusers* – based on voting principles and *soft fusers* – based on support accumulation. The use of probabilistic base classifiers allows in this case the use of a method with accumulation of support, additionally enriched by weighing. The weights of member classifiers in the proposed method will be their quality measured with the *f-measure* metric determined for the training data. Utilized F-score is determined as ratio between duplicated product of precision and recall relative to its sum [21].

The full processing scheme of the method proposed in this paper is presented in Fig. 1. The available training set is divided into a random subsets of features using the *Random Subspace* method and minority class in all the subspaces is independently oversampled using the SMOTE method. The structure of classifiers constructed in this way allows for prediction by support accumulation weighted with *f-measure* obtained on a training set.

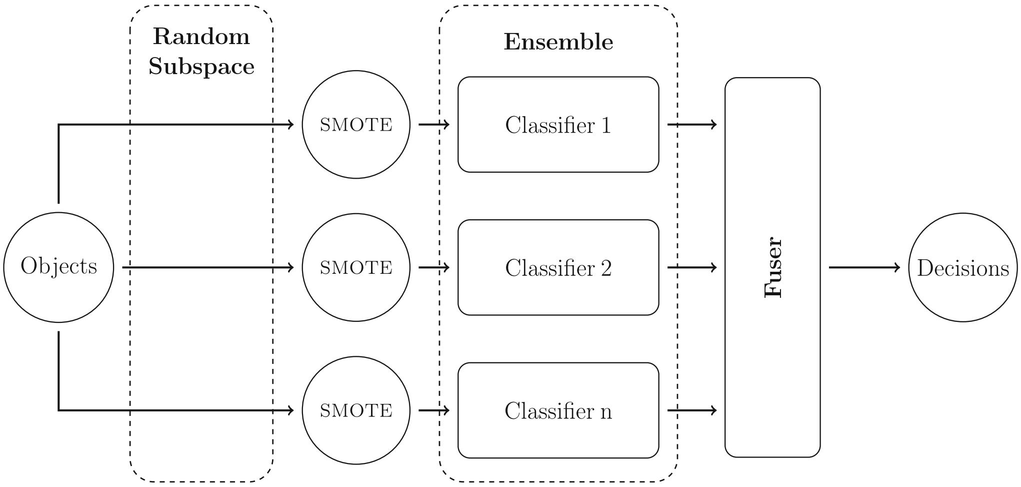

Fig. 1. Random subspace approach to build classifier ensemble

3 Experiments Set-Up

The experimental evaluation was carried out on the basis of 30 datasets with various imbalance ratio, available in the KEEL *data repository* [1]. The application of the *Random Subspace* method was based on arbitrarily set 30 subspaces using three features each [23]. As base classifiers, three standard methods with probabilistic interpretation were adopted:

1. Gaussian Naive Bayes
2. Logistic Regression
3. Support Vector Machine

The method was programmed in accordance with the programming interface of the *scikit-learn* environment [19], so it was possible to use the basic classifiers implemented in it. The used implementation of the SMOTE algorithm was the version available in the *imbalanced-learn* library [17].

The study identified and tested six approaches for each base classifier:

1. Base method.
2. SMOTE oversampling
3. RS – *Random Subspace*
4. RS+SMOTE – Combined *Random Subspace* and SMOTE
5. WRS – *Random Subspace* weighted by *f-score* obtained on a training set
6. WRS+SMOTE – Combined *Random Subspace* and SMOTE weighted by *f-score* obtained on a training set.

In the course of the experiment stratified 5-fold crossvalidation was used [4], measuring quality of all predictions using the *f-score* method. Tests analyzing statistical dependence were carried out using the Wilcoxon test [7]. The full code of the proposed solution as well as the experiments themselves was made available through the public Git repository[1].

4 Experimental Evaluation

The detailed results of the experiments, including the averaged f-measure for all folds of cross-validation along with the evaluation of the statistical dependence of the analyzed solutions for each of the data sets are presented in Tables 1, 2 and 3 respectively for all three base classifiers.

Table 4 contains the ranks determined by ranked Wilcoxon test for each of the analyzed methods and base classifiers. Based on them, assuming the levels of significance .9 and .95, a summary was prepared, counting the cases of advantage of each solution over the others and stored in Table 5.

As may be observed from the obtained results, which was also in line with expectations, regardless of the underlying classifier, using only SMOTE, without

[1] https://github.com/w4k2/wrssmote.

Table 1. Classification results on *Gaussian Naive Bayes*

Dataset	Base method	SMOTE	RS	RS+SMOTE	WRS	WRS+SMOTE
australian	0.719	0.718	0.755	0.795	**0.859**	**0.863**
	—	—	—	1,2,3	1,2,3,4	1,2,3,4
glass-0-1-2-3-vs-4-5-6	**0.709**	**0.718**	**0.694**	**0.744**	**0.680**	**0.737**
	—	—	—	—	—	—
glass-0-1-4-6-vs-2	**0.219**	**0.231**	**0.292**	**0.270**	**0.266**	**0.226**
	—	—	—	—	—	—
glass-0-1-5-vs-2	**0.218**	**0.161**	0.016	**0.198**	**0.218**	**0.176**
	3	3	—	3	—	3
glass-0-1-6-vs-2	**0.199**	**0.190**	**0.192**	**0.228**	**0.155**	**0.193**
	—	—	—	—	—	—
glass-0-1-6-vs-5	**0.760**	**0.760**	**0.636**	**0.557**	**0.717**	**0.648**
	—	—	—	—	—	—
glass-0-4-vs-5	**0.960**	**0.893**	**0.727**	**0.727**	**0.893**	**0.893**
	—	—	—	—	—	—
glass-0-6-vs-5	**0.893**	**0.893**	**0.733**	**0.670**	**0.796**	**0.664**
	—	—	—	—	—	—
glass0	**0.642**	**0.639**	**0.633**	**0.644**	**0.644**	**0.649**
	—	—	—	—	—	—
glass1	**0.604**	**0.638**	**0.504**	**0.626**	**0.603**	**0.626**
	—	—	—	—	—	—
glass2	**0.189**	**0.193**	**0.177**	**0.181**	**0.202**	**0.203**
	—	—	—	—	—	—
glass4	**0.237**	**0.509**	**0.200**	**0.580**	**0.200**	**0.567**
	—	—	—	—	—	—
glass5	**0.768**	**0.768**	**0.591**	**0.498**	**0.659**	**0.590**
	—	—	—	—	—	—
glass6	**0.772**	**0.786**	**0.815**	**0.826**	**0.800**	**0.811**
	—	—	—	—	—	—
heart	**0.802**	**0.808**	**0.797**	**0.800**	**0.827**	**0.808**
	—	—	—	—	—	—
hepatitis	0.719	**0.851**	**0.649**	0.588	**0.917**	**0.893**
	—	—	—	—	1,4	4

Table 1. (*continued*)

Dataset	Base method	SMOTE	RS	RS+SMOTE	WRS	WRS+SMOTE
page-blocks-1-3-vs-4	**0.493**	**0.540**	**0.470**	**0.498**	**0.435**	**0.471**
	—	—	—	—	—	—
pima	**0.621**	**0.665**	0.523	**0.657**	0.597	**0.664**
	3	3,5	—	3,5	3	3,5
shuttle-c0-vs-c4	0.980	0.980	**0.975**	**0.981**	**0.996**	1.000
	—	—	—	—	—	1,2
vowel0	**0.709**	**0.592**	0.057	**0.568**	**0.426**	**0.569**
	3	3	—	3	3	3
wisconsin	**0.943**	**0.944**	**0.959**	**0.957**	**0.959**	**0.959**
	—	—	—	—	—	—
yeast-0-2-5-6-vs-3-7-8-9	**0.262**	**0.488**	0.023	**0.377**	0.113	**0.471**
	3	3,5	—	3	—	3,5
yeast-0-2-5-7-9-vs-3-6-8	0.201	0.173	**0.619**	0.195	**0.706**	**0.650**
	—	—	1,2,4	—	1,2,4	1,2,4
yeast-0-3-5-9-vs-7-8	**0.269**	**0.216**	**0.244**	**0.186**	**0.151**	**0.080**
	—	6	—	—	—	—
yeast-0-5-6-7-9-vs-4	0.175	0.180	**0.386**	0.193	**0.429**	**0.445**
	—	—	1,2	—	—	1,2,4
yeast-2-vs-4	0.297	0.270	**0.574**	0.312	**0.743**	**0.671**
	—	—	—	—	1,2,4	1,2,4
yeast-2-vs-8	0.254	0.175	**0.683**	0.109	**0.658**	**0.658**
	—	—	1,2,4	—	2,4	2,4
yeast1	0.457	0.458	**0.557**	0.484	0.489	**0.557**
	—	—	1,2,4,5	1,2	1,2	1,2,4,5
yeast3	0.236	0.252	**0.642**	0.243	**0.608**	**0.686**
	—	—	1,2,4	—	1,2,4	1,2,4
yeast5	0.154	0.192	0.181	0.107	**0.582**	**0.485**
	4	4	4	—	1,2,3,4	1,2,3,4

using Random Subspace, positively influences the quality of classification against the training of the classifier just with the original data set. At the same time, using only *Random Subspace* leads to very poor results, even worse than the base method.

Table 2. Classification results on *Logistic Regression*

Dataset	Base method	SMOTE	RS	RS+SMOTE	WRS	WRS+SMOTE
australian	**0.842**	**0.846**	**0.841**	**0.859**	**0.859**	**0.853**
	—	—	—	—	—	—
glass-0-1-2-3-vs-4-5-6	**0.775**	**0.807**	**0.547**	**0.747**	**0.583**	**0.720**
	—	—	—	—	—	—
glass-0-1-4-6-vs-2	**0.000**	**0.258**	**0.000**	**0.290**	**0.000**	**0.286**
	—	1,3,5	—	—	—	—
glass-0-1-5-vs-2	0.000	**0.184**	0.000	**0.172**	0.000	**0.181**
	—	1,3,5	—	1,3,5	—	1,3,5
glass-0-1-6-vs-2	**0.000**	**0.176**	**0.000**	**0.196**	**0.000**	**0.282**
	—	1,3,5	—	1,3,5	—	—
glass-0-1-6-vs-5	**0.149**	**0.451**	0.000	**0.519**	0.000	**0.519**
	—	3,5	—	3,5	—	3,5
glass-0-4-vs-5	**0.367**	**0.720**	0.000	**0.829**	0.162	**0.829**
	—	3,5	—	3,5	—	3,5
glass-0-6-vs-5	**0.347**	**0.584**	0.000	**0.638**	0.000	**0.657**
	3,5	3,5	—	3,5	—	3,5
glass0	**0.516**	**0.678**	0.237	**0.675**	0.369	**0.678**
	3	3,5	—	3,5	—	3,5
glass1	0.243	**0.568**	0.000	**0.553**	0.133	**0.538**
	3,5	1,3,5	—	1,3,5	3	1,3,5
glass2	0.000	**0.167**	0.000	**0.202**	0.000	**0.195**
	—	1,3,5	—	1,3,5	—	1,3,5
glass4	0.167	**0.578**	0.000	**0.591**	**0.200**	**0.566**
	—	1,3	—	1,3	—	1,3
glass5	**0.149**	**0.510**	0.000	**0.465**	0.000	**0.428**
	—	3,5	—	3,5	—	3,5
glass6	**0.759**	**0.832**	**0.570**	**0.825**	**0.742**	**0.825**
	—	—	—	—	—	—

Table 2. (*continued*)

Dataset	Base method	SMOTE	RS	RS+SMOTE	WRS	WRS+SMOTE
heart	**0.829** 4,6	**0.817** —	**0.804** —	0.794 —	**0.829** —	0.794 —
hepatitis	**0.919** —	**0.875** —	**0.912** —	**0.904** —	**0.912** —	**0.904** —
page-blocks-1-3-vs-4	**0.560** —	**0.683** —	**0.421** —	**0.537** —	**0.442** —	**0.501** —
pima	0.618 3,5	**0.684** 1,3,5	0.432 —	**0.653** 3,5	0.560 3	**0.665** 3,5
shuttle-c0-vs-c4	**0.996** —	1.000 —	**0.996** —	**0.996** —	**0.996** —	**0.996** —
vowel0	**0.579** 3	**0.652** 3	0.087 —	**0.637** 3	**0.550** 3	**0.628** 3
wisconsin	**0.947** —	**0.956** —	**0.950** —	**0.959** —	**0.945** —	**0.959** —
yeast-0-2-5-6-vs-3-7-8-9	0.019 —	**0.461** 1,3,5	0.000 —	**0.423** 1,3,5	0.000 —	**0.420** 1,3,5
yeast-0-2-5-7-9-vs-3-6-8	0.236 3,5	**0.587** 1,3,5	0.000 —	**0.563** 3,5	0.000 —	**0.575** 1,3,5
yeast-0-3-5-9-vs-7-8	0.067 —	**0.275** 1,3,5	0.000 —	**0.244** 1,3,5	0.067 —	**0.261** 1,3,5
yeast-0-5-6-7-9-vs-4	0.000 —	**0.468** 1,3,5	0.000 —	**0.466** 1,3,5	0.000 —	**0.488** 1,3,5
yeast-2-vs-4	0.170 —	**0.690** 1,3,5	0.000 —	**0.664** 1,3,5	0.103 —	**0.680** 1,3,5
yeast-2-vs-8	0.080 —	**0.546** 1,3,5	0.000 —	**0.480** 1,3,5	0.080 —	**0.658** 1,3,5
yeast1	0.329 3	**0.584** 1,3,5	0.062 —	**0.565** 1,3,5	0.241 3	**0.593** 1,3,5
yeast3	0.109 3	**0.671** 1,3,5	0.000 —	**0.681** 1,3,5	0.056 —	**0.686** 1,3,5
yeast5	0.000 —	**0.480** 1,3,5	0.000 —	**0.461** 1,3,5	0.000 —	**0.462** 1,3,5

Table 3. Classification results on *Support Vector Machines*

Dataset	Base method	SMOTE	RS	RS+SMOTE	WRS	WRS+SMOTE
australian	0.000	0.247	0.693	**0.795**	0.675	**0.813**
	—	—	1,2	1,2,3,5	1,2	1,2,3,5
glass-0-1-2-3-vs-4-5-6	**0.741**	**0.864**	**0.674**	**0.835**	**0.666**	**0.842**
	—	—	—	—	—	—
glass-0-1-4-6-vs-2	0.000	**0.296**	0.000	**0.366**	0.000	**0.375**
	—	1,3,5	—	1,3,5	—	1,3,5
glass-0-1-5-vs-2	0.000	**0.264**	0.000	**0.377**	0.000	**0.389**
	—	1,3,5	—	1,3,5	—	1,3,5
glass-0-1-6-vs-2	0.000	**0.286**	0.000	**0.364**	0.000	**0.380**
	—	1,3,5	—	1,3,5	—	1,3,5
glass-0-1-6-vs-5	0.000	**0.544**	0.000	**0.492**	0.000	**0.466**
	—	1,3,5	—	1,3,5	—	1,3,5
glass-0-4-vs-5	**0.431**	**0.762**	0.031	**0.629**	**0.693**	**0.629**
	3	3	—	3	3	3
glass-0-6-vs-5	**0.467**	**0.800**	0.000	**0.700**	0.000	**0.700**
	3,5	3,5	—	3,5	—	3,5
glass0	0.460	**0.699**	0.198	**0.731**	0.455	**0.738**
	3	1,3,5	—	1,3,5	3	1,3,5
glass1	**0.578**	**0.603**	0.184	**0.596**	**0.618**	**0.588**
	3	3	—	3	3	3
glass2	0.000	**0.247**	0.000	**0.287**	0.000	**0.309**
	—	1,3,5	—	1,3,5	—	1,3,5
glass4	**0.671**	**0.772**	0.000	**0.762**	**0.593**	**0.781**
	3	3	—	3	3	3
glass5	0.000	**0.403**	0.000	**0.537**	0.000	**0.472**
	—	1,3,5	—	1,3,5	—	1,3,5
glass6	**0.824**	**0.809**	**0.827**	**0.847**	**0.827**	**0.847**
	—	—	—	—	—	—
heart	0.000	0.063	**0.764**	**0.802**	0.657	**0.714**
	—	1	1,2,5	1,2,5	1,2	1,2
hepatitis	**0.912**	**0.912**	**0.912**	**0.927**	**0.912**	**0.906**
	—	—	—	—	—	—

Table 3. (*continued*)

Dataset	Base method	SMOTE	RS	RS+SMOTE	WRS	WRS+SMOTE
page-blocks-1-3-vs-4	0.000	0.037	0.114	**0.478**	0.057	**0.398**
	—	—	—	1,2,3,5	—	1,2,3,5
pima	0.000	0.015	0.000	**0.400**	0.000	**0.368**
	—	—	—	1,2,3,5	—	1,2,3,5
shuttle-c0-vs-c4	0.190	0.606	**0.987**	**0.983**	**0.987**	**0.987**
	—	1	1,2	1,2	1,2	1,2
vowel0	**0.547**	**0.566**	0.208	**0.706**	**0.600**	**0.678**
	—	3	—	3	3	3
wisconsin	**0.933**	**0.936**	**0.961**	**0.968**	**0.961**	**0.968**
	—	—	—	—	—	—
yeast-0-2-5-6-vs-3-7-8-9	0.000	**0.493**	0.000	**0.420**	0.118	**0.402**
	—	1,3,5	—	1,3,5	—	1,3,5
yeast-0-2-5-7-9-vs-3-6-8	0.000	**0.652**	0.053	**0.559**	0.167	**0.579**
	—	1,3,5	—	1,3,5	—	1,3,5
yeast-0-3-5-9-vs-7-8	0.000	**0.198**	0.000	**0.239**	**0.114**	**0.241**
	—	1,3	—	1,3	—	1,3
yeast-0-5-6-7-9-vs-4	0.000	**0.472**	0.000	**0.495**	0.000	**0.497**
	—	1,3,5	—	1,3,5	—	1,3,5
yeast-2-vs-4	0.000	**0.697**	**0.463**	**0.664**	**0.589**	**0.682**
	—	1	1	1	1	1
yeast-2-vs-8	**0.613**	**0.658**	0.000	**0.487**	**0.658**	**0.658**
	3	3	—	3	3	3
yeast1	0.073	**0.573**	0.097	**0.574**	0.345	**0.584**
	—	1,3,5	—	1,3,5	1,3	1,3,5
yeast3	0.000	0.681	0.000	**0.721**	**0.759**	**0.701**
	—	1,3	—	1,3	1,2,3	1,3
yeast5	0.000	**0.457**	0.000	**0.522**	0.000	**0.523**
	—	1,3,5	—	1,3,5	—	1,3,5

Enhancing the Random Subspace method with weighing, which – thanks to the f-measure – is realized in the context of the imbalanced problem, makes it to be more effective than the base method, although it still does not affect the quality of classification as positively as SMOTE.

Table 4. Ranks computed by the Wilcoxon test

	Base method	SMOTE	RS	RS+SMOTE	WRS	WRS+SMOTE
GNB	-	191.0	258.0	245.0	196.0	132.0
SMOTE	274.0	-	273.0	279.0	194.0	137.0
RS	207.0	192.0	-	240.0	102.5	75.0
RS+SMOTE	220.0	186.0	195.0	-	153.0	100.0
WRS	269.0	241.0	362.5	312.0	-	203.5
WRS+SMOTE	333.0	298.0	360.0	335.0	231.5	-
LR	-	10.0	416.5	18.0	389.0	22.0
SMOTE	455.0	-	460.0	283.0	453.0	235.0
RS	18.5	5.0	-	4.0	52.0	4.0
RS+SMOTE	417.0	182.0	431.0	-	429.0	187.0
WRS	46.0	12.0	383.0	6.0	-	7.0
WRS+SMOTE	413.0	230.0	431.0	248.0	428.0	-
SVM	-	2.0	213.0	7.0	101.5	1.0
SMOTE	433.0	-	371.0	137.0	348.5	114.0
RS	222.0	64.0	-	1.0	114.0	5.0
RS+SMOTE	458.0	328.0	464.0	-	431.0	234.0
WRS	363.5	116.5	321.0	34.0	-	26.5
WRS+SMOTE	464.0	321.0	430.0	231.0	438.5	-

The extension of the basic form of *Random Subspace* with SMOTE leads to slightly better results than using only SMOTE, which suggests that the positive influence on the quality of classification by both methods may be independent and it may be complemented. This is confirmed by the use of the full proposal, based on the weighted Random Subspace from SMOTE, whose quality is definitely the best in the competition and except one case (dataset *heart* with the Linear Regression) there are no situations where it is not among the best approaches in the considered pool.

The *Random Subspace* method is particularly popular in the case of multidimensional data, being a solution to the significant problem of the curse of dimensionality. Lowering the number of features analyzed by each model, using the same cardinality of patterns, reduces the decision space, and thus compacts the samples in space. On the other hand, the role of SMOTE is to equalize the density of the occurrence of patterns in the problem by compacting the objects of the minority class. Using both methods together, controlling the influence of each subspace on the final prediction of the ensemble by weighing it with a

Table 5. Summary of the Wilcoxon test. ●= the method in the row improves the method of the column. ○= the method in the column improves the method of the row. Upper diagonal of level significance $\alpha = 0.9$, Lower diagonal level of significance $\alpha = 0.95$

	Base method	SMOTE	RS	RS+SMOTE	WRS	WRS+SMOTE
GNB	-					○
SMOTE		-				○
RS			-		○	○
RS+SMOTE				-		○
WRS			●		-	
WRS+SMOTE	●		●	●		-
LR	-	○	●	○	●	○
SMOTE	●	-	●		●	
RS	○	○	-	○	○	○
RS+SMOTE	●		●	-	●	
WRS	○	○	●	○	-	○
WRS+SMOTE	●		●		●	-
SVM	-	○		○	○	○
SMOTE	●	-	●	○	●	○
RS		○	-	○	○	○
RS+SMOTE	●	●	●	-	●	
WRS	●	○	●	○	-	○
WRS+SMOTE	●	●	●		●	-
$\alpha = .9$	2	6	0	7	4	11
$\alpha = .95$	2	6	0	7	4	10

measure adequate to the imbalanced problem may lead to close to the optimal placement of training patterns in the classification space, and thus lead to a model with a high discriminatory ability, as shown by carried out experiments.

5 Summary

This paper proposes the use of a classifier ensemble diversified using the *Random Subspace* approach and trained on sets oversampled independently in each subspace using the SMOTE algorithm. The experiments performed for its needs are testing the quality of such solution with relation to both concepts present in it, using 30 imbalanced data sets and three probabilistic base classifiers.

As shown by the results of experiments, it is a promising approach, able to effectively use the advantages of both methods leading to a better solution than each of them individually. Research shows that the *Random Subspace* method, although in itself, does not allow to improve the prediction of imbalanced data, in the weighted option may positively affect the achieved quality of classification. Combining it with the creation of synthetic patterns in the subspace areas of the problem gives an effective solution with high usefulness in the processing of this type of problems.

Acknowledgements. This work was supported by the Polish National Science Center under the grant no. UMO- 2015/19/B/ST6/01597 and by the statutory fund of the Faculty of Electronics, Wroclaw University of Science and Technology.

References

1. Alcalá-Fdez, J., et al.: Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. J. Mult.-Valued Log. Soft Comput. **17**, 255–287 (2011)
2. Bunkhumpornpat, C., Sinapiromsaran, K., Lursinsap, C.: Safe-Level-SMOTE: safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS (LNAI), vol. 5476, pp. 475–482. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-01307-2_43
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
4. Diamantidis, N., Karlis, D., Giakoumakis, E.A.: Unsupervised stratification of cross-validation for accuracy estimation. Artif. Intell. **116**(1–2), 1–16 (2000)
5. Dietterich, T.G.: Ensemble methods in machine learning. In: Kittler, J., Roli, F. (eds.) MCS 2000. LNCS, vol. 1857, pp. 1–15. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45014-9_1
6. García, S., Herrera, F.: Evolutionary undersampling for classification with imbalanced datasets: proposals and taxonomy. Evol. Comput. **17**(3), 275–306 (2009)
7. Gehan, E.A.: A generalized Wilcoxon test for comparing arbitrarily singly-censored samples. Biometrika **52**(1–2), 203–224 (1965)
8. Han, H., Wang, W.-Y., Mao, B.-H.: Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In: Huang, D.-S., Zhang, X.-P., Huang, G.-B. (eds.) ICIC 2005. LNCS, vol. 3644, pp. 878–887. Springer, Heidelberg (2005). https://doi.org/10.1007/11538059_91
9. He, H., Bai, Y., Garcia, E.A., Li, S.: ADASYN: adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence). IEEE, June 2008
10. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Trans. Knowl. Data Eng. **9**, 1263–1284 (2008)
11. Huang, H.Y., Lin, Y.J., Chen, Y.S., Lu, H.Y.: Imbalanced data classification using random subspace method and SMOTE. In: The 6th International Conference on Soft Computing and Intelligent Systems, and the 13th International Symposium on Advanced Intelligence Systems. IEEE, November 2012

12. Jeni, L.A., Cohn, J.F., De La Torre, F.: Facing imbalanced data-recommendations for the use of performance metrics. In: 2013 Humaine Association Conference on Affective Computing and Intelligent Interaction, pp. 245–251. IEEE (2013)
13. Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. Prog. Artif. Intell. **5**(4), 221–232 (2016)
14. Krawczyk, B., Woźniak, M., Herrera, F.: On the usefulness of one-class classifier ensembles for decomposition of multi-class problems. Pattern Recogn. **48**(12), 3969–3982 (2015)
15. Ksieniewicz, P.: Undersampled majority class ensemble for highly imbalanced binary classification. In: Torgo, L., Matwin, S., Japkowicz, N., Krawczyk, B., Moniz, N., Branco, P. (eds.) Proceedings of the Second International Workshop on Learning with Imbalanced Domains: Theory and Applications. Proceedings of Machine Learning Research, PMLR, ECML-PKDD, Dublin, Ireland, vol. 94, pp. 82–94, 10 September 2018
16. Kuncheva, L.I.: Combining Pattern Classifiers: Methods and Algorithms. Wiley, Hoboken (2004)
17. Lemaître, G., Nogueira, F., Aridas, C.K.: Imbalanced-learn: a Python toolbox to tackle the curse of imbalanced datasets in machine learning. J. Mach. Learn. Res. **18**(1), 559–563 (2017)
18. Maciejewski, T., Stefanowski, J.: Local neighbourhood extension of SMOTE for mining imbalanced data. In: 2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM). IEEE, April 2011
19. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**(Oct), 2825–2830 (2011)
20. Quinlan, J.R., et al.: Bagging, boosting, and C4. 5. In: AAAI/IAAI, vol. 1, pp. 725–730 (1996)
21. Sasaki, Y., et al.: The truth of the F-measure. Teach Tutor Mater **1**(5), 1–5 (2007)
22. Sun, Y., Wong, A.K., Kamel, M.S.: Classification of imbalanced data: a review. Int. J. Pattern Recogn. Artif. Intell. **23**(04), 687–719 (2009)
23. Topolski, M.: Multidimensional MCA correspondence model supporting intelligent transport management. Arch. Transp. Syst. Telemat. **11**, 52–56 (2018)
24. Topolski, M.: Algorithm of multidimensional analysis of main features of PCA with blurry observation of facility features detection of carcinoma cells multiple myeloma. In: Burduk, R., Kurzynski, M., Wozniak, M. (eds.) CORES 2019. AISC, vol. 977, pp. 286–294. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-19738-4_29
25. Wozniak, M.: Hybrid Classifiers: Methods of Data, Knowledge, and Classifier Combination. Studies in Computational Intelligence, vol. 519. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40997-4
26. Yu, G., Zhang, G., Domeniconi, C., Yu, Z., You, J.: Semi-supervised classification based on random subspace dimensionality reduction. Pattern Recogn. **45**(3), 1119–1135 (2012)