



A Genetic-Based Ensemble Learning Applied to Imbalanced Data Classification

Jakub Klikowski, Paweł Ksieniewicz^(✉), and Michał Woźniak

Department of Systems and Computer Networks,
Wrocław University of Science and Technology, Wrocław, Poland
{jakub.klikowski,pawel.ksieniewicz,michal.wozniak}@pwr.edu.pl

Abstract. Imbalanced data classification is still a focus of intense research, due to its ever-growing presence in the real-life decision tasks. In this article, we focus on a classifier ensemble for imbalanced data classification. The ensemble is formed on the basis of the individual classifiers trained on supervise-selected feature subsets. There are several methods employing this concept to ensure a high diverse ensemble, nevertheless most of them, as *Random Subspace* or *Random Forest*, select attributes for a particular classifier randomly. The main drawback of mentioned methods is not giving the ability to supervise and control this task. In following work, we apply a genetic algorithm to the considered problem. Proposition formulates an original learning criterion, taking into consideration not only the overall classification performance but also ensures that trained ensemble is characterised by high diversity. The experimental study confirmed the high efficiency of the proposed algorithm and its superiority to other ensemble forming method based on random feature selection.

Keywords: Machine learning · Classification · Imbalanced data · Feature selection · Genetic algorithm

1 Introduction

The most of classification data sets do not have exactly the same number of instances for each class. The serious problem appears when the data is significantly imbalanced, when one of the classes quantitatively dominates over other ones. Data imbalance may have a negative effect on the efficiency of standard classification algorithms, thus this problem requires a proper approach and the use of dedicated techniques [10].

Data pre-processing is one of the solutions to this problem. The operation of such methods consists in the prior preparation of data, where algorithm equalizes the number of instances in particular classes. There are three main approaches: creating additional objects in the minority class (*oversampling*), removing objects in the majority class (*undersampling*), or combine both techniques.

Another approach focuses on designing algorithms dedicated to imbalanced data analysis. Appropriate procedures and modifications change the operation of standard classifiers. The introduced changes are aimed at improving the obtained classification quality by increasing the weight of minority data in the imbalanced set. One of such modifications is the feature selection. Lee et al. [12] presented a classifier with feature selection based on fuzzy entropy for pattern classification. The pattern space is partitioned into non-overlapping fuzzy decision regions. It allows to obtain smooth boundaries and achieve better classification performance. Koziarski et al. [9] came up with idea of feature selection based ensemble learning method, which creates subspaces in guided incremental manner according to parameters setting. One of the parameters decides about preference toward feature quality or diversity. Canuto and Nascimento [4] proposed the genetic-based feature selection method to create ensemble classifiers. They use an hybrid and adaptive fitness function consists of the features correlation measure and the ensemble accuracy. Du et al. [6] invented a feature selection based method for imbalanced data with genetic algorithm. It improves the quality of a single model by using geometric mean score as fitness function. Haque et al. [8] proposed an genetic algorithm method for learning heterogeneous ensembles on imbalanced data. This idea is based on the search for possible combinations of different types of classifiers in the ensemble with selecting the best compositions using genetic algorithm with the Matthews correlation coefficient score as the fitness function.

In this work, we present the *Genetic Ensemble Selection* (GES) classifier, being a novel ensemble learning method which trains diverse individuals on the basis of supervised selected features using a genetic algorithm. Base genetic approach is extended by an original regularisation component to the learning criterion, aiming to ensure a high diversity of the ensemble and to protect it against the hazard of overfitting. The experiments conducted on the collection of benchmark datasets prove the validity and a good performance of the proposed method. The solution refers to the method proposed by Ksieniewicz and Woźniak [11] who applied random search to find the best ensemble in the pool using (*balanced accuracy score*) [3]. Following work expands research on this idea by implementing a genetic algorithm to search optimal subspaces of features and modifies the proposed regularisation criterion for alternatives solutions.

In a nutshell, the main contributions of this works state as follows:

- Proposition of a genetic-based classifier ensemble forming algorithm employing feature selection.
- Proposition of an original fitness function consisting of a classification performance metric with the regularisation component.
- Evaluation of the proposed method on the basis of the wide range benchmark datasets.

2 Proposed Method

The aim of the classification algorithm is to assign an observed object to one of the predefined set of labels $\mathcal{M} = \{1, 2, \dots, M\}$. The decision is made on the

basis of an analysis of the selected attributes, which are gathered in the feature vector x

$$x = [x^1, x^2, \dots, x^d]^T \in \mathcal{X}. \quad (1)$$

Let us define a pool of individual classifiers as

$$\Pi = \{\Psi_1, \Psi_2, \dots, \Psi_K\}, \quad (2)$$

where Ψ_k denotes the k -th elementary classifier.

In this work the individuals use subsets of attributes to ensure the high diversity of the ensemble [14], therefore let's propose the following representation of Π as a bit word

$$\Pi = [[b_1^1, b_1^2, \dots, b_1^d] [b_2^1, b_2^2, \dots, b_2^d] \dots [b_K^1, b_K^2, \dots, b_K^d]], \quad (3)$$

where b_i^j denotes if the j th feature is used by the i th classifier.

As the optimisation criterion we propose

$$Q(\Pi) = \text{metric}(\Pi) - \alpha * \frac{\text{no} - \text{features}(\Pi)}{d} + \beta * \frac{\text{av} - \text{Hamming}(\Pi)}{d}, \quad (4)$$

where $\text{metric}(\Pi)$ denotes value of the chosen performance metric, $\text{no} - \text{features}(\Pi)$ is the number of features used by all classifiers from Π and $\text{av} - \text{Hamming}(\Pi)$ stands for the average *Hamming distance* between the words represented individuals in Π . Procedure to count the criterion is shown at Algorithm 1.

Algorithm 1. Criterion count

```

1: Input: pool of individual classifiers  $\Pi$ , training set  $\mathcal{T}\mathcal{S}$ 
2: Parameters:  $\alpha$ ,  $\beta$ 
3: Output: value of criterion (eq. 4) for  $\Pi$ 
4:
5:  $\text{counter} \leftarrow 0$ 
6:  $\text{nobits} \leftarrow 0$ 
7:  $\text{word} \leftarrow [00..0]$ 
8: for  $i \leftarrow 1$  to  $K - 1$  do
9:   for  $j \leftarrow i$  to  $K$  do
10:     $\text{counter} \leftarrow \text{counter} + 1$ 
11:     $\text{nobits} \leftarrow \text{nobits} + \text{number of bits of } [b_i^1, b_i^2, \dots, b_i^d] \text{ XOR } [b_j^1, b_j^2, \dots, b_j^d]$ 
12:   end for
13:    $\text{word} \leftarrow \text{wordOR} [b_i^1, b_i^2, \dots, b_i^d]$ 
14: end for
15:  $\text{word} \leftarrow \text{wordOR} [b_K^1, b_K^2, \dots, b_K^d]$ 
16:  $\text{no} - \text{features} \leftarrow \text{number of bits in word} * \frac{1}{d}$ 
17:  $\text{av} - \text{Hamming} \leftarrow \frac{\text{nobits}}{\text{counter} * d}$ 
18:  $\text{BAC} \leftarrow \text{balanced accuracy of } \Pi \text{ calculated on } \mathcal{T}\mathcal{S}$ 
19:  $\text{criterion} \leftarrow \text{BAC} - \alpha * \text{no} - \text{used} - \text{features} + \beta * \text{av} - \text{Hamming} - \text{dist}$ 
20: return  $\text{criterion}$ 

```

We employ the genetic algorithm for finding the optimal ensemble line-up. The pseudocode of GES (*Genetic Ensemble Selection*) procedure is presented in Algorithm 2. Let's shortly describe its main steps.

Algorithm 2. Feature Selection Genetic Ensemble Classifier procedure

```

1: Input:
2: Parameters:  $e$  - number of generations  $p$  - population size  $ms$  - mutation size  $mp$ 
   - mutation probability  $cs$  - crossing size  $cp$  - crossing probability  $n$  - elite size
3: Output:
4:  $population \leftarrow$  create  $p$  random subspaces of features
5: for  $i \leftarrow 0$  to  $e$  do
6:    $mindexes \leftarrow$  select  $ms$  individuals indexes list with  $mp$  probability
7:   for all  $m$  in  $mindexes$  do
8:      $population[m] \leftarrow$  invert random features bits of  $population[m]$ 
9:   end for
10:
11:    $cindexes \leftarrow$  select  $cs$  individuals indexes list with  $cp$  probability
12:   for all  $c$  in  $cindexes$  do
13:      $p1 \leftarrow population[c]$ 
14:      $j \leftarrow$  generates a random number in the range of 0 to  $population$  length
15:      $p2 \leftarrow population[j]$ 
16:      $c1, c2 \leftarrow$  exchange randomly selected features between  $p1$  and  $p2$ 
17:      $population[c] \leftarrow c1$ 
18:      $population[j] \leftarrow c2$ 
19:   end for
20:
21:    $scores \leftarrow$  new list
22:   for  $j \leftarrow 0$  to  $population$  length do
23:      $quality \leftarrow$  calculate quality for  $population[j]$ 
24:     insert  $quality$  into  $scores[j]$ 
25:   end for
26:
27:    $population \leftarrow population$  sort by  $scores$ 
28:    $elite \leftarrow$  select first  $n$  individuals from  $population$ 
29:    $population \leftarrow elite +$  select  $(p - n)$  individuals from  $population$ 
30: end for

```

Initialisation Procedure. Algorithm 2 starts with the randomly generated population of the ensembles $Population = \{II_1, II_2, \dots, II_S\}$. Its size is an arbitrarily chosen by an input parameter. Basically, the larger the population exists, the more comprehensive optimisation is performed.

Evaluate Population Procedure. Individuals in the population are evaluated by criterion 4 calculated on the basis of algorithm (Eq. 1) using samples stored in the training set. Obtained results are used to select the elite (the best evaluated ensemble), being immune to mutation or replacement by crossover.

Genetic Operators. The mutation shall inject some randomness into chromosomes. We used a simple bit mutation (flipping random bits in a binary word). The crossover operator exchanges random member classifiers between two randomly chosen parent individuals to form child chromosomes, using standard two-point crossover operator [2].

Selection and Reproduction. To maintain the diversity of the population, which is essential for ensuring the ability to explore the space of possible solution of the problem effectively, the selection process has to be realised in probabilistic manner i.e., not only are the best chromosomes chosen, but also the probability of selection is straight proportional to their evaluation according to the criterion 4. To meet this assumption, a standard ranking selection procedure was implemented, where $S - 1$. Generated individuals and the elite individual are joined together and form offspring population.

3 Experimental Evaluation

Conducting appropriate experiments and analysis of the results should allow us to verify the idea and check the quality of the classification for the proposed method. This section describes the set-up of the conducted experiments, presents their results and analysis. During the experiments we would like to verify the following research hypotheses:

- Does the proposed approach improve the classification quality?
- Do the adjusted regularisation methods have a positive effect on the operation of the classifier?
- Is the random subspace an adequate solution for the feature selection for this classifier proposal?

3.1 Set-Up

All experiments were carried out using benchmark binary-class imbalanced data sets, which are posted and publicly available on the KEEL repository [1]. Selected data sets have a different imbalance ratios (from balanced problems to 1:33 proportion) and the various number of features. The tests were carried out on the basis of 5-fold stratified cross validation. This approach allows us a reliable check of the proposed method. The *Friedman ranking test* [5] was used for assessing the ranks of classifiers over all examined benchmarks. Checks whether the assigned rank differs significantly from assigning to each classifier an average rank. In the case if the hypothesis of ranking equality is rejected, the *Shaffer post hoc test* [7] is used to find out which of the tested methods are distinctive among an $N \times N$ comparison. The post hoc procedure was based on the significance level $\alpha = 0.05$.

The implementation was done using according to guidelines of *scikit-learn* library API [13]. *Logistic Regression* and *Gaussian Naive Bayes* were used as the base classifiers and the tests were carried out for the comparative analysis of different approaches of GES building classifier ensembles on them:

- GES—base *Genetic Ensemble Selection* classifier without regularization,
- GES-A—*Genetic Ensemble Selection* with *alpha*-regularization,
- GES-B—*Genetic Ensemble Selection* with both *alpha*- and *beta*-regularization.

Each comparison was supplemented with a bare-base method trained on the whole set of the features and a *Random Subspace* (RS) approach. All 10 approaches have been tested and evaluated using three metrics *F-score*, *Geometric Mean Score* and *Balanced accuracy score* [3], where metric used for evaluation is always used also as a part of optimization criterion *Q*. The entire implementation and code allowing to repeat all the experiments has been placed in a publicly available repository¹.

3.2 Results

The first experiment was aiming to identify the best hyperparameters for the genetic algorithm implemented within the GES. Parameters that have been tested include the probability of crossing individuals and the probability of mutation.

Conducted research allowed to indicate the best parameters value in the range from 0% to 10% for crossing and in the range from 0% to 2% for mutation. Figure 1 shows the visualisation of the obtained learning curves for two exemplary datasets. The intensity of the green tells us about the average value of the learning curve in the course. The height of the dot on the right and the result above each square is the quality on the test set of the already learned model. In addition, the intensity of the red of this dot tells whether it is among the worst or the best results. After analysis of all the results, the optimal parameters were selected as 2.5% for crossover and 1% for mutation. These hyperparameter settings were later used to carry out further experiments.

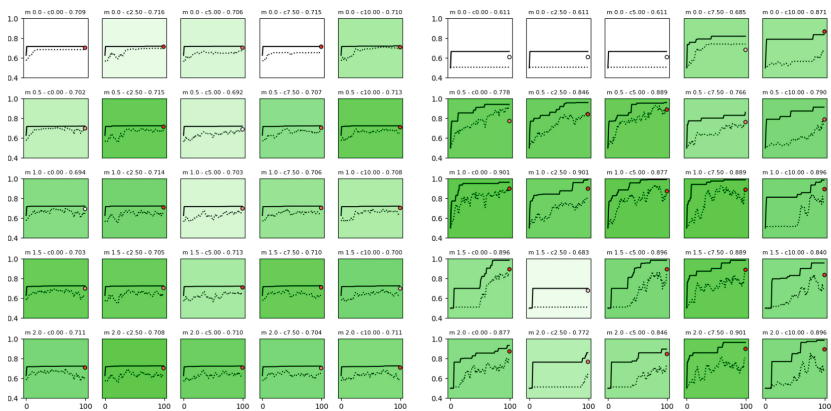


Fig. 1. Crossing and mutation probability influence on BAC.

¹ <https://github.com/w4k2/genetic-ensemble-selection>.

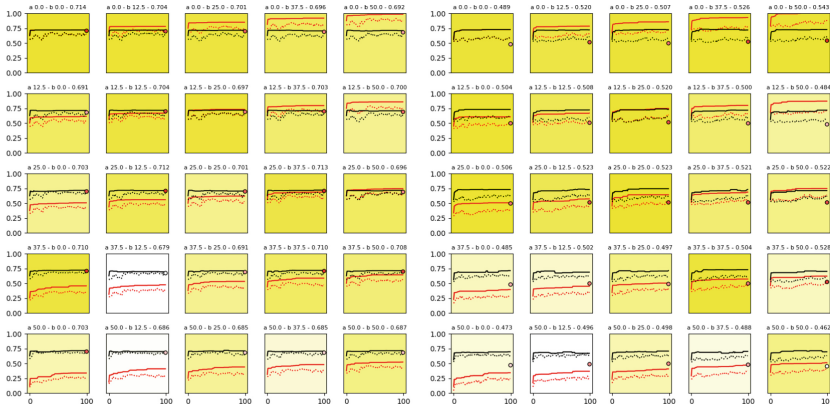


Fig. 2. Example of alpha and beta influence on BAC.

The next experiment was to find the best regularization parameters α and β . Parameters were tested in the range from 0 to 1 with a step of 0.125. Figure 2 shows the visualisations of the results obtained on exemplary datasets. The intensity of the yellow tells us about the average value of the learning curve in the course. The height of the dot on the right and the result above each square is the quality on the test set of the already learned model. In addition, the intensity of the red of this dot tells whether it is among the worst or the best results. The analysis of the obtained results allows to find best parameters.

The main experiment was aimed at conducting research on data sets after optimising the parameters of the proposed method. Tables 1, 2 and 3 shows the obtained results. Each table has been divided into two parts, for both employed base classifiers. The ranking tests outcome is presented in Table 4 for Logistic Regression as a base classifier and Table 5 for Gaussian Naive Bayes.

3.3 Analysis

The scores obtained show that the proposed GES method has a certain potential. Ranking tests indicate that GES and its variants have much better performance than the random subspace or base classifier method. GES is able to choose the most favourable ensemble of subspaced classifiers, which gives a significant advantage to RS. On the other side, learning using this method is computationally complex process and it can not be applied to problems where the classifier’s learning speed is an important factor. In response to hypotheses of experiments:

- Results shows that features selection with the genetic algorithm approach can improve imbalanced data classification quality.
- Regularisation methods have no significant impact on the classifier’s operation. This is indicated by good performance and lack of statistical difference between base GES and its variants—GES-A and GES-B where regularisation is reduced to one metric or totally excluded.

Table 1. Overview of the results of the main experiment for F -score.

dataset	Logistic Regression					Gaussian Naive Bayes				
	LR 1	RS 2	GES 3	GES-A 4	GES-B 5	GNB a	RS b	GES c	GES-A d	GES-B e
<i>australian</i>	0.762	0.728	0.808	0.794	0.800	0.719	0.732	0.834	0.852	0.844
	-	-	2	2	2	-	-	a,b	a,b	a,b
<i>glass-0-1-2-3-vs-4-5-6</i>	0.775	0.385	0.800	0.770	0.695	0.709	0.612	0.719	0.656	0.728
	2	-	2	2	2	-	-	-	-	-
<i>glass-0-1-4-6-vs-2</i>	0.000	0.215	0.147	0.214	0.267	0.219	0.096	0.252	0.267	0.242
	-	1	1	1	1	-	-	-	-	-
<i>glass-0-1-5-vs-2</i>	0.000	0.179	0.197	0.194	0.205	0.218	0.014	0.288	0.340	0.271
	-	1	1	-	1	b	-	b	b	-
<i>glass-0-1-6-vs-2</i>	0.000	0.199	0.181	0.172	0.168	0.199	0.088	0.244	0.220	0.273
	-	1	1	1	1	-	-	-	-	-
<i>glass-0-1-6-vs-5</i>	0.149	0.000	0.115	0.015	0.000	0.760	0.180	0.760	0.648	0.648
	-	-	-	-	-	b	-	b	b	b
<i>glass-0-4-vs-5</i>	0.507	0.040	0.711	0.333	0.360	0.960	0.268	0.773	0.960	0.727
	2	-	2	-	-	b	-	b	b	b
<i>glass-0-6-vs-5</i>	0.333	0.000	0.144	0.180	0.160	0.893	0.349	0.893	0.867	0.960
	-	-	-	-	-	b	-	b	b	b
<i>glass0</i>	0.516	0.000	0.699	0.701	0.660	0.642	0.627	0.628	0.625	0.632
	2	-	2	1,2	2	-	-	-	-	-
<i>glass1</i>	0.243	0.000	0.583	0.524	0.563	0.604	0.490	0.634	0.626	0.630
	2	-	1,2	1,2	1,2	-	-	-	-	-
<i>glass2</i>	0.000	0.165	0.152	0.068	0.158	0.189	0.168	0.216	0.189	0.190
	-	1	-	-	1	-	-	-	-	-
<i>glass4</i>	0.167	0.200	0.441	0.428	0.541	0.237	0.000	0.257	0.300	0.257
	-	-	-	-	-	b	-	-	b	-
<i>glass5</i>	0.149	0.200	0.333	0.200	0.348	0.768	0.140	0.634	0.457	0.768
	-	-	-	-	-	b	-	b	-	b
<i>glass6</i>	0.759	0.238	0.811	0.832	0.778	0.772	0.809	0.775	0.738	0.800
	2	-	2	2	2	-	-	-	-	-
<i>heart</i>	0.822	0.207	0.796	0.787	0.775	0.802	0.769	0.816	0.812	0.837
	2	-	2	2	2	-	-	-	-	b
<i>hepatitis</i>	0.919	0.912	0.912	0.903	0.899	0.719	0.615	0.889	0.840	0.795
	-	-	-	-	-	-	-	b	-	-
<i>page-blocks-1-3-vs-4</i>	0.533	0.421	0.548	0.550	0.568	0.493	0.347	0.419	0.419	0.419
	-	-	-	-	-	-	-	-	-	-
<i>pima</i>	0.629	0.000	0.634	0.646	0.635	0.621	0.501	0.600	0.621	0.616
	2	-	2	2	2	b	-	b	b	b
<i>shuttle-c0-vs-c4</i>	0.996	0.031	0.996	0.996	0.996	0.980	0.968	0.988	0.984	0.984
	-	-	2	2	2	-	-	-	-	-
<i>shuttle-c2-vs-c4</i>	1.000	0.960	0.960	1.000	0.960	0.813	0.800	0.800	0.800	0.867
	-	-	-	-	-	-	-	-	-	-
<i>vowel0</i>	0.540	0.113	0.637	0.553	0.573	0.709	0.717	0.736	0.716	0.726
	2	-	2	2	2	-	-	-	-	-
<i>wisconsin</i>	0.949	0.048	0.909	0.891	0.868	0.943	0.941	0.943	0.937	0.933
	2	-	2	2	2	-	-	-	-	-
<i>yeast-0-2-5-6-vs-3-7-8-9</i>	0.056	0.000	0.067	0.024	0.065	0.262	0.026	0.410	0.384	0.296
	2	-	-	-	-	b	-	b	b	b
<i>yeast-0-2-5-7-9-vs-3-6-8</i>	0.315	0.000	0.683	0.347	0.055	0.201	0.174	0.752	0.677	0.748
	2,5	-	2,5	-	-	b	-	a,b	a,b	a,b
<i>yeast-0-3-5-9-vs-7-8</i>	0.036	0.000	0.114	0.114	0.000	0.269	0.148	0.377	0.269	0.345
	-	-	-	-	-	-	-	b	-	-
<i>yeast-0-5-6-7-9-vs-4</i>	0.000	0.000	0.000	0.067	0.000	0.175	0.067	0.521	0.480	0.531
	-	-	-	-	-	b	-	a,b	a,b	a,b
<i>yeast-2-vs-4</i>	0.236	0.000	0.186	0.151	0.359	0.297	0.508	0.770	0.713	0.774
	2	-	-	-	2	-	-	a	a	a
<i>yeast-2-vs-8</i>	0.000	0.000	0.000	0.000	0.000	0.254	0.249	0.658	0.658	0.658
	-	-	-	-	-	-	-	-	-	-
<i>yeast1</i>	0.329	0.374	0.567	0.563	0.571	0.457	0.502	0.529	0.519	0.549
	-	-	1,2	1,2	1,2	-	-	a	a	a
<i>yeast3</i>	0.153	0.225	0.756	0.729	0.758	0.236	0.548	0.778	0.778	0.767
	-	-	1,2	1,2	1,2	-	a	a,b	a,b	a,b
<i>yeast5</i>	0.000	0.000	0.000	0.000	0.000	0.154	0.321	0.586	0.664	0.669
	-	-	-	-	-	-	a	a,b	a,b	a,b
Ranking	3.2742	4.1452	2.2258	2.7097	2.6452	3.4194	4.5161	2.0806	2.8226	2.1613

Table 2. Overview of the results of the main experiment for *Gmean* score.

dataset	Logistic Regression					Gaussian Naive Bayes				
	LR 1	RS 2	GES 3	GES-A 4	GES-B 5	GNB a	RS b	GES c	GES-A d	GES-B e
<i>australian</i>	0.788	0.711	0.812	0.827	0.815	0.752	0.763	0.884	0.865	0.864
	2	-	2	2	2	-	-	a,b	a,b	a,b
<i>glass-0-1-2-3-vs-4-5-6</i>	0.833	0.000	0.891	0.878	0.926	0.792	0.685	0.789	0.780	0.775
	2	-	2	2	2	-	-	-	-	-
<i>glass-0-1-4-6-vs-2</i>	0.000	0.338	0.519	0.471	0.481	0.495	0.282	0.603	0.614	0.619
	-	-	1	1	1	-	-	-	-	-
<i>glass-0-1-5-vs-2</i>	0.000	0.000	0.591	0.592	0.596	0.510	0.051	0.611	0.559	0.638
	-	-	1,2	1,2	1,2	b	-	b	b	b
<i>glass-0-1-6-vs-2</i>	0.000	0.286	0.421	0.458	0.537	0.481	0.258	0.494	0.578	0.463
	-	-	1	1	1	-	-	-	-	-
<i>glass-0-1-6-vs-5</i>	0.259	0.000	0.555	0.411	0.417	0.930	0.691	0.788	0.935	0.735
	-	-	2	-	2	b	-	-	b	-
<i>glass-0-4-vs-5</i>	0.624	0.000	0.893	0.394	0.337	0.994	0.584	0.822	0.994	0.994
	2	-	2,5	-	-	b	-	-	b	b
<i>glass-0-6-vs-5</i>	0.461	0.000	0.362	0.318	0.435	0.936	0.715	0.936	0.936	0.936
	2	-	-	-	2	-	-	-	-	-
<i>glass0</i>	0.619	0.000	0.739	0.749	0.757	0.632	0.605	0.624	0.633	0.624
	2	-	2	2	2	-	-	-	-	-
<i>glass1</i>	0.377	0.000	0.564	0.589	0.514	0.543	0.529	0.553	0.505	0.548
	2	-	1,2	1,2	2	-	-	-	-	-
<i>glass2</i>	0.000	0.385	0.397	0.395	0.536	0.568	0.459	0.562	0.529	0.514
	-	1	1	1	1	-	-	-	-	-
<i>glass4</i>	0.228	0.231	0.811	0.595	0.663	0.341	0.000	0.341	0.342	0.523
	-	-	1,2	-	-	b	-	b	b	b
<i>glass5</i>	0.274	0.200	0.639	0.677	0.768	0.929	0.672	0.788	0.744	0.885
	-	-	-	-	1,2	b	-	-	-	b
<i>glass6</i>	0.855	0.000	0.879	0.898	0.896	0.862	0.856	0.864	0.890	0.844
	2	-	2	2	2	-	-	-	-	-
<i>heart</i>	0.841	0.306	0.828	0.804	0.836	0.822	0.792	0.825	0.820	0.838
	2	-	2	2	2	-	-	-	-	-
<i>hepatitis</i>	0.634	0.844	0.628	0.748	0.737	0.678	0.623	0.778	0.797	0.718
	-	-	-	-	-	-	-	-	b	-
<i>page-blocks-1-3-vs-4</i>	0.688	0.643	0.921	0.902	0.908	0.653	0.543	0.581	0.547	0.548
	-	-	-	-	-	-	-	-	-	-
<i>pima</i>	0.703	0.000	0.722	0.718	0.703	0.701	0.595	0.699	0.685	0.695
	2	-	2	2	2	b	-	b	b	b
<i>shuttle-c0-vs-c4</i>	0.996	0.081	0.996	0.996	0.996	0.991	0.994	0.992	0.996	0.996
	2	-	2	2	2	-	-	-	-	-
<i>shuttle-c2-vs-c4</i>	1.000	0.996	0.996	0.996	0.996	0.983	0.800	0.800	0.937	0.937
	-	-	-	-	-	-	-	-	-	-
<i>vowel0</i>	0.719	0.161	0.703	0.695	0.758	0.873	0.814	0.829	0.821	0.803
	2	-	2	2	2	-	-	-	-	-
<i>wisconsin</i>	0.959	0.120	0.885	0.907	0.888	0.962	0.958	0.954	0.965	0.957
	2	-	2	2	2	-	-	-	-	-
<i>yeast-0-2-5-6-vs-3-7-8-9</i>	0.135	0.000	0.127	0.063	0.125	0.364	0.013	0.456	0.439	0.363
	2	-	-	-	-	b	-	b	b	b
<i>yeast-0-2-5-7-9-vs-3-6-8</i>	0.399	0.000	0.700	0.108	0.108	0.393	0.172	0.905	0.885	0.846
	2	-	2,4,5	-	-	b	-	a,b	a,b	a,b
<i>yeast-0-3-5-9-vs-7-8</i>	0.063	0.000	0.126	0.089	0.000	0.346	0.138	0.661	0.577	0.602
	-	-	-	-	-	-	-	a,b	a,b	a,b
<i>yeast-0-5-6-7-9-vs-4</i>	0.000	0.000	0.000	0.000	0.000	0.182	0.112	0.754	0.727	0.751
	-	-	-	-	-	-	-	a,b	a,b	a,b
<i>yeast-2-vs-4</i>	0.332	0.000	0.216	0.000	0.354	0.432	0.611	0.850	0.882	0.864
	2,4	-	-	-	-	-	-	a	a,b	a
<i>yeast-2-vs-8</i>	0.000	0.000	0.000	0.000	0.000	0.564	0.513	0.745	0.723	0.749
	-	-	-	-	-	-	-	-	-	-
<i>yeast1</i>	0.456	0.501	0.693	0.692	0.692	0.206	0.621	0.669	0.670	0.679
	-	-	1,2	1,2	1,2	-	a	a	a	a
<i>yeast3</i>	0.282	0.384	0.897	0.896	0.907	0.446	0.680	0.894	0.912	0.905
	-	-	1,2	1,2	1,2	-	a	a,b	a,b	a,b
<i>yeast5</i>	0.000	0.000	0.000	0.000	0.000	0.811	0.927	0.941	0.966	0.964
	-	-	-	-	-	-	a	a	a	a
Ranking	3.2742	4.3871	2.3548	2.7258	2.2581	3.0323	4.5968	2.4516	2.3065	2.6129

Table 3. Overview of the results of the main experiment for *Balanced accuracy score*

dataset	Logistic Regression					Gaussian Naive Bayes				
	LR 1	RS 2	GES 3	GES-A 4	GES-B 5	GNB a	RS b	GES c	GES-A d	GES-B e
<i>australian</i>	0.791	0.727	0.812	0.823	0.818	0.770	0.781	0.869	0.863	0.853
	2	-	2	2	2	-	-	a,b	a,b	a,b
<i>glass-0-1-2-3-vs-4-5-6</i>	0.852	0.500	0.904	0.920	0.911	0.822	0.770	0.851	0.810	0.835
	2	-	2	2	2	-	-	-	-	-
<i>glass-0-1-4-6-vs-2</i>	0.500	0.611	0.544	0.619	0.642	0.561	0.477	0.665	0.643	0.640
	-	-	-	-	1	-	-	-	-	-
<i>glass-0-1-5-vs-2</i>	0.500	0.500	0.551	0.537	0.560	0.549	0.446	0.686	0.681	0.627
	-	-	-	-	-	-	-	b	b	-
<i>glass-0-1-6-vs-2</i>	0.500	0.583	0.607	0.636	0.563	0.547	0.448	0.588	0.657	0.583
	-	-	1	-	-	-	-	-	b	-
<i>glass-0-1-6-vs-5</i>	0.584	0.500	0.591	0.650	0.603	0.939	0.746	0.889	0.884	0.893
	-	-	2	-	2	b	-	-	-	-
<i>glass-0-4-vs-5</i>	0.750	0.500	0.894	0.706	0.832	0.994	0.679	0.944	0.994	0.994
	2	-	2	-	2	b	-	b	b	b
<i>glass-0-6-vs-5</i>	0.679	0.500	0.614	0.643	0.520	0.945	0.770	0.995	0.945	0.940
	-	-	-	-	-	-	-	b	-	-
<i>glass0</i>	0.657	0.500	0.763	0.776	0.769	0.701	0.684	0.705	0.701	0.705
	2	-	2	2	2	-	-	-	-	-
<i>glass1</i>	0.478	0.493	0.583	0.603	0.571	0.634	0.574	0.574	0.583	0.581
	-	-	-	2	2	-	-	-	-	-
<i>glass2</i>	0.500	0.565	0.567	0.544	0.534	0.604	0.551	0.672	0.632	0.682
	-	-	-	-	-	-	-	-	-	-
<i>glass4</i>	0.494	0.567	0.839	0.829	0.874	0.508	0.435	0.515	0.512	0.477
	-	-	1,2	1,2	1,2	-	-	-	-	-
<i>glass5</i>	0.594	0.600	0.790	0.793	0.752	0.938	0.729	0.888	0.949	0.949
	-	-	-	-	-	b	-	-	b	b
<i>glass6</i>	0.876	0.500	0.926	0.879	0.912	0.876	0.878	0.878	0.895	0.903
	2	-	2	2	2	-	-	-	-	-
<i>heart</i>	0.843	0.552	0.810	0.828	0.815	0.823	0.795	0.804	0.815	0.823
	2	-	2	2	2	-	-	-	-	-
<i>hepatitis</i>	0.745	0.849	0.745	0.760	0.759	0.722	0.689	0.810	0.796	0.853
	-	-	-	-	-	-	-	-	-	a,b
<i>page-blocks-1-3-vs-4</i>	0.790	0.748	0.932	0.904	0.909	0.762	0.692	0.714	0.697	0.695
	-	-	-	-	-	-	-	-	-	-
<i>pima</i>	0.722	0.500	0.725	0.727	0.723	0.713	0.653	0.706	0.708	0.707
	2	-	2	2	2	b	-	-	b	b
<i>shuttle-c0-vs-c4</i>	0.996	0.508	0.996	0.996	0.996	0.991	0.994	0.992	0.996	0.996
	2	-	2	2	2	-	-	-	-	-
<i>shuttle-c2-vs-c4</i>	1.000	0.996	0.996	0.946	0.996	0.984	0.900	0.900	0.946	0.946
	-	-	-	-	-	-	-	-	-	-
<i>vowel0</i>	0.772	0.533	0.767	0.765	0.799	0.881	0.830	0.835	0.874	0.829
	2	-	2	2	2	-	-	-	-	-
<i>wisconsin</i>	0.960	0.513	0.919	0.915	0.907	0.962	0.958	0.955	0.965	0.959
	2	-	2	2	2	-	-	-	-	-
<i>yeast-0-2-5-6-vs-3-7-8-9</i>	0.513	0.500	0.534	0.503	0.526	0.549	0.472	0.605	0.529	0.604
	-	-	-	-	-	-	-	-	-	-
<i>yeast-0-2-5-7-9-vs-3-6-8</i>	0.602	0.500	0.547	0.509	0.562	0.560	0.486	0.900	0.856	0.851
	2,4	-	2,4	-	-	b	-	a,b	a,b	a,b
<i>yeast-0-3-5-9-vs-7-8</i>	0.508	0.500	0.539	0.538	0.538	0.573	0.513	0.604	0.631	0.645
	-	-	-	-	-	-	-	-	b	b
<i>yeast-0-5-6-7-9-vs-4</i>	0.500	0.500	0.500	0.500	0.507	0.499	0.501	0.775	0.775	0.777
	-	-	-	-	-	-	-	a,b	a,b	a,b
<i>yeast-2-vs-4</i>	0.570	0.500	0.530	0.560	0.550	0.598	0.697	0.869	0.897	0.886
	2	-	-	-	-	-	-	a	a,b	a,b
<i>yeast-2-vs-8</i>	0.500	0.500	0.500	0.500	0.500	0.667	0.644	0.773	0.773	0.773
	-	-	-	-	-	-	-	-	-	-
<i>yeast1</i>	0.585	0.609	0.690	0.703	0.688	0.518	0.656	0.665	0.660	0.661
	-	-	1	1	1	-	a	a	a	a
<i>yeast3</i>	0.542	0.575	0.911	0.897	0.908	0.597	0.729	0.905	0.889	0.903
	-	-	1,2	1,2	1,2	-	a	a,b	a,b	a,b
<i>yeast5</i>	0.500	0.500	0.500	0.500	0.500	0.829	0.930	0.956	0.956	0.933
	-	-	-	-	-	-	a	a	a	a
Ranking	3.4516	4.2258	2.4355	2.4355	2.4516	3.2742	4.5968	2.4194	2.3387	2.371

Table 4. Shaffer test ($\alpha = 0.05$) for comparison between the proposed methods (GMean). Shaffer’s procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.005 . Symbol ‘ \times ’ stands for classifiers without significant differences, ‘ \checkmark ’ for situation in which the method on the left is superior.

Algorithms	BAC		F-score		GMean	
	<i>p</i> – values		<i>p</i> – values		<i>p</i> – values	
RS vs. GES	0.000008	\checkmark	0.000002	\checkmark	0	\checkmark
RS vs. GES-A	0.000008	\checkmark	0.000351	\checkmark	0.000035	\checkmark
RS vs. GES-B	0.00001	\checkmark	0.000188	\checkmark	0	\checkmark
LR vs. GES	0.011402	\checkmark	0.009042	\checkmark	0.022069	\checkmark
LR vs. GES-A	0.011402	\checkmark	0.159833	\times	0.172104	\times
LR vs. GES-B	0.012775	\checkmark	0.117284	\times	0.011402	\checkmark
GES vs. GES-B	0.967965	\times	0.2964	\times	0.809582	\times
GES-A vs. GES-B	0.967965	\times	0.872374	\times	0.244153	\times
GES vs. GES-A	1	\times	0.228269	\times	0.355641	\times

Table 5. Shaffer test ($\alpha = 0.05$) for comparison between the proposed methods (GMean). Shaffer’s procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.005 . Symbol ‘ \times ’ stands for classifiers without significant differences, ‘ \checkmark ’ for situation in which the method on the left is superior.

Algorithms	BAC		F-score		GMean	
	<i>p</i> – values		<i>p</i> – values		<i>p</i> – values	
RS vs. GES	0	\checkmark	0	\checkmark	0	\checkmark
RS vs. GES-A	0	\checkmark	0.000025	\checkmark	0	\checkmark
RS vs. GES-B	0	\checkmark	0	\checkmark	0.000001	\checkmark
GNB vs. GES-A	0.019841	\checkmark	0.000858	\checkmark	0.148235	\times
GNB vs. GES-B	0.024512	\checkmark	0.137291	\times	0.070724	\times
GNB vs. GES	0.033293	\checkmark	0.001733	\checkmark	0.2964	\times
GES vs. GES-A	0.840851	\times	0.064689	\times	0.717764	\times
GES vs. GES-B	0.904101	\times	0.840851	\times	0.687971	\times
GES-A vs. GES-B	0.935981	\times	0.09964	\times	0.445429	\times

- Classification based only on *Random Subspace* gives quite poor performance. Employment of *Random Subspace* to select features along with the optimised selection allows a significant improvement compared to the classification on the full feature space.

4 Conclusions and Future Directions

Presented work focuses on an innovative ensemble method (GES) dedicated for imbalanced data classification. It is based on the genetic algorithm aiming to optimise selection of feature combinations. Conducted experimental evaluation shows the superiority of GES compared to both the basic random selection of features as well as to the base classifiers used to build ensembles learned on a full feature space.

An important aspect of analysis was the inclusion of regularisation component in fitness function, aiming to ensure the diversity of the classifiers committee. Experiments have shown that introducing it does not give statistically significant improvement in results, so in order to further develop of this method, a different regularisation proposal should be developed. New regularisation should give a greater impact on the classifier operation and better performance of the classification. The proposed method can be considered a useful and effective tool in the classification of data with varying degrees of imbalance.

Acknowledgments. This work was supported by the Polish National Science Centre under the grant No. 2017/27/B/ST6/01325 as well as by the statutory funds of the Department of Systems and Computer Networks, Faculty of Electronics, Wrocław 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. Back, T., Fogel, D., Michalewicz, Z.: *Handbook of Evolutionary Computation*. Oxford University Press, New York (1997)
3. Branco, P., Torgo, L., Ribeiro, R.P.: Relevance-based evaluation metrics for multi-class imbalanced domains. In: Kim, J., Shim, K., Cao, L., Lee, J.-G., Lin, X., Moon, Y.-S. (eds.) *PAKDD 2017. LNCS (LNAI)*, vol. 10234, pp. 698–710. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57454-7_54
4. Canuto, A.M., Nascimento, D.S.: A genetic-based approach to features selection for ensembles using a hybrid and adaptive fitness function. In: *The 2012 international joint conference on neural networks (IJCNN)*, pp. 1–8. IEEE (2012)
5. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.* **7**, 1–30 (2006)
6. Du, L., Xu, Y., Jin, L.: Feature selection for imbalanced datasets based on improved genetic algorithm. In: *Decision Making and Soft Computing: Proceedings of the 11th International FLINS Conference*, pp. 119–124. World Scientific (2014)
7. García, S., Fernández, A., Luengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power. *Inf. Sci.* **180**(10), 2044–2064 (2010)
8. Haque, M.N., Noman, N., Berretta, R., Moscato, P.: Heterogeneous ensemble combination search using genetic algorithm for class imbalanced data classification. *PloS One* **11**(1), e0146116 (2016)

9. Koziarski, M., Krawczyk, B., Woźniak, M.: The deterministic subspace method for constructing classifier ensembles. *Pattern Anal. Appl.* **20**(4), 981–990 (2017)
10. Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. *Prog. Artif. Intell.* **5**(4), 221–232 (2016). <https://doi.org/10.1007/s13748-016-0094-0>
11. Ksieniewicz, P., Woźniak, M.: Imbalanced data classification based on feature selection techniques. In: Yin, H., Camacho, D., Novais, P., Tallón-Ballesteros, A.J. (eds.) *IDEAL 2018. LNCS*, vol. 11315, pp. 296–303. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03496-2_33
12. Lee, H.M., Chen, C.M., Chen, J.M., Jou, Y.L.: An efficient fuzzy classifier with feature selection based on fuzzy entropy. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **31**(3), 426–432 (2001)
13. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)
14. Wozniak, M., Graña, M., Corchado, E.: A survey of multiple classifier systems as hybrid systems. *Inf. Fusion* **16**, 3–17 (2014)