Actively Balanced Bagging for Imbalanced Data

Jerzy Błaszczyński$^{(\boxtimes)}$ and Jerzy Stefanowski

Institute of Computing Science, Poznań University of Technology,
Piotrowo 2, 60-965 Poznań, Poland
{jerzy.blaszczynski,jerzy.stefanowski}@cs.put.poznan.pl

Abstract. Under-sampling extensions of bagging are currently the most accurate ensembles specialized for class imbalanced data. Nevertheless, since improvements of recognition of the minority class, in this type of ensembles, are usually associated with a decrease of recognition of majority classes, we introduce a new, two phase, ensemble called Actively Balanced Bagging. The proposal is to first learn a bagging classifier and then iteratively improve it by updating its bootstraps with a limited number learning examples. The examples are selected according to an active learning strategy, which takes into account: decision margin of votes, example class distribution in the training set and/or in its neighbourhood, and prediction errors of component classifiers. Experiments with synthetic and real-world data confirm usefulness of this proposal.

Keywords: Class imbalance · Active learning · Bagging · Ensembles of classifiers · Neighbourhood Balanced Bagging

1 Introduction

The problem of learning from class-imbalanced data has been a topic of intensive research in recent years, motivated by an observation that standard learning methods are inherently biased to focus on the majority classes and fail to sufficiently recognize the minority class which is often of particular interest in the given application. For reviews of new methods see, e.g., [6,10].

Ensembles are among the most effective methods for improving recognition of the minority class. Most of them extend strategies from bagging or boosting. They either employ pre-processing methods before learning component classifiers or embed a cost-sensitive framework in the ensemble learning process; see review in [9,10]. Experimental comparisons [3,11,12] have shown that extensions of bagging work better than those of boosting and other solutions.

The basic idea behind extending bagging for imbalanced data is to modify the distribution of examples from minority and majority classes in bootstraps [9]. This may be achieved in many ways, which usually balance the numbers of examples from both classes. Experimental studies show that under-sampling, i.e., reduction of examples from the majority class, performs better than over-sampling (i.e., multiplication of examples from the minority class) [3,4,11]. Improvements are observed even for simplest extensions, like Exactly

Balanced Bagging (EBBag) [7,9]. Other studies [3,11] demonstrate that Roughly Balanced Bagging (RBBag), which applies a more sophisticated random under-sampling, achieves the best G-mean among compared ensembles. We have proposed Neighbourhood Balanced Bagging (NBBag) being competitive to RBBag. NBBag modifies bootstrap sampling by weighting examples [4]. The weight of an minority example depends on quantity of examples in the whole training set, and in the example neighbourhood, which belong to the majority class.

However, these improvements may come with a decrease of recognition majority class examples [4]. We claim that there is still a possibility to better learn the trade-off between performance in both classes. This leads us to a research question how to achieve it. To best of our knowledge the current research focuses on various modifications of bootstrap samples in under-sampling bagging extensions. In this paper, we want to consider another hypothesis: given an already good technique of constructing a bagging extension, is it possible to perform additional steps of updating its bootstraps by selecting a limited number of learning examples, which are important to improve performance in both classes.

This hypothesis directs our attention toward *active learning methods* [15]. Recall that, in active learning, a learning method may achieve a better classification performance when it is allowed to choose examples on which it learns. Usually it is performed on partially labeled examples, i.e., after the initial step of full supervised learning, the active method should select a limited number of examples to be labeled and further used to update the classifier. However, the active learning could also be used to filter examples in the complete labeled examples [2]. In this way active strategies have been already applied to imbalanced learning - although these attempts are still quite limited (see Sect. 2).

We introduce a new approach, called Active Balancing Bagging (ABBag), to extend bagging for imbalanced data by an active learning modification of bootstrap samples of previously constructed under-sampling bagging extension. In this proposal, after the initial training, the ensemble (here we consider two algorithms: EBBag and NBBag) is re-trained on bootstraps enlarged by batches, i.e., small portions of learning examples, selected according to active selection strategies. Batch querying, although being very practical by nature, is rarely used by existing active learning approaches [5]. According to our best knowledge no similar extension has been considered in the literature so far.

The paper is organized as follows. The next section summarizes related works on bagging extensions proposed for imbalanced data and active learning strategies. The following Sect. 3, describes the proposed active balancing bagging extension. Experimental validation of the proposed methods is carried out in Sect. 4. The paper is concluded in Sect. 5.

2   Related Works

Besides working on new algorithms, some researchers also studied the characteristics of imbalanced data and sources of difficulties for learning classifiers. It has been noticed that the global imbalance ratio is not the only or even not

the most important factor which makes learning difficult. Other data difficulty factors such as class overlapping, small disjuncts or lack of representativeness significantly deteriorate the quality of an induced model even on exactly balanced data [13,14]. They also inspire development of new methods, see e.g. [4].

Several extensions of bagging approach for imbalanced data have been proposed [4,9,16]. In the simplest one, called Exactly Balanced Bagging (EBBag), while constructing bootstrap sample, the entire minority class is copied and combined with the randomly chosen subset of the majority class to exactly balance cardinalities of examples in both classes. The first bagging extension which uses knowledge of data difficulty factors is Neighbourhood Balanced Bagging (NBBag) [4]. NBBag focuses bootstrap sampling toward difficult minority examples by using certain type of weights. The weight of a minority example depends on the analysis of class labels among its k nearest neighbours. A minority example is considered the more unsafe, the more it has majority examples in its neighbourhood. Thus, this part of the weight reflects a local balancing factor. Moreover, this weight is also aggregated with a global balancing factor, which takes into account the imbalance ratio between classes. Hence, the formula for minority example weight is the following: $w = 0.5 \times \left(\frac{(N'_-)^\psi}{k} + 1 \right)$ where N'_- is the number of majority examples among k nearest neighbours of the example and ψ is a scaling factor. Setting $\psi = 1$ causes a linear amplification of the example weight with an increase of unsafeness and setting ψ to values greater then 1 effects in an exponential amplification. Each majority example is assigned a constant weight $w = 0.5 \times \frac{N_+}{N_-}$.

Active learning approaches for imbalanced data are rather limited. The active strategy proposed by Ertkin et al. [8] relies on iterations of selecting single uncertain example and rebuilding the model. However, selecting more that one instance at time, and rebuilding model on a batch, often can greatly reduce the computation time [5]. This is especially true when the cost of constructing classifier is higher than for online SVM [8]. A simple active extension of bagging has been proposed for unlabeled data in imbalanced medical problems [16].

3   Active Selection of Examples in Balanced Bagging

In this section we describe Actively Balanced Bagging (ABBag), which is composed of two phases. The first phase consists in learning an ensemble classifier by one of approaches to construct under-sampling extensions of bagging. Although one can choose any extension, we will further consider one simple, yet effective one: Exactly Balanced Bagging (EBBag) [9], and better performing, yet more complex one: Neighbourhood Balanced Bagging NBBag [4]. For more information on constructing these ensemble refer to Sect. 2. Here, we focus on an explanation of the second phase, which is specific to ABBag, and which we call *active selection of examples*. It consists in: (1) an iterative modification of bootstrap samples, constructed in the first phase, by adding selected examples from the training set; (2) re-training of component classifiers on modified bootstraps. The examples in (1) are added to bootstraps in small portions called batches.

The proposed active selection of examples can be seen as a variant of Query-by-committee (QBC) approach [1]. In QBC, one constructs an ensemble of diverse component classifiers and then ranks all classified examples with respect to a selected committee disagreement measure (a decision margin). Originally, this margin is defined as a difference between number of votes for a class, or probabilities of a class, between the two most supported classes. Note that QBC has been already successfully applied in active learning of ensemble of diverse component classifiers in [5]. Ensembles constructed in the first phase of ABBag are attractive to combine with QBC, because, as we have previously shown in [4], they promote component classifiers diversity. However, QBC does not take into account global (i.e., concerning the whole training set) properties of examples distribution, and in result, it can focus on selecting outliers and sparse regions [5].

Recall that single example selection is a typical strategy in active learning [2, 15]. In ABBag we promote to select, in each iteration, a small batch instead of one example. It is motivated by a potential reduction of computation time as well as increasing diversity of examples in the batch. As it was observed in [5], a greedy selection of example with respect to a single criterion, e.g., typical for active strategy: highest utility/uncertainty measure [2,15], does not provide desired diversity. In our view, giving chance to also randomly select some slightly sub-optimal examples besides the best ones may result in higher diversity of new bootstraps and increased diversity of re-trained component classifiers.

We address the above mentioned issues twofold. First, and foremost, the proposed active selection of examples considers multiple factors. Among them, we consider: uncertainty of example prediction by component classifiers, and a prediction error of a single component classifier. In addition, we consider factors specific to imbalanced data, which reflect more global (i.e., concerning the whole training set) and/or local (i.e., concerning example neighbourhood) class distribution of examples. Second, we use a specific variant of the rejection sampling to enforce diversity within the batch through randomization.

The pseudo-code of the algorithm for learning ABBag ensemble is presented as Algorithm 1. It starts with training set LS, and m_{bag} bootstrap samples $\boldsymbol{S}$, and it results, in the first phase (lines 1–3), in an under-sampling extension of bagging. It makes use of initial balancing weights $\boldsymbol{w}$, which are calculated in accordance with the under-sampling bagging extension. These initial *balancing weights* $\boldsymbol{w}$ allow us to direct sampling toward more difficult to learn examples in comparison to uniform sampling typical for bagging. In case of EBBag, $\boldsymbol{w}$ reflects the global imbalance of example in the training set. For NBBag, $\boldsymbol{w}$ expresses both global and local imbalance of example in the training set [4].

In the second phase, the active selection of examples is performed between lines 4, and 10. All bootstraps from $\boldsymbol{S}$ are iteratively (m_{al} times) enlarged by adding batches, and new component classifiers are re-learned. In each iteration, new weights $\boldsymbol{w'}$ of examples are calculated according to weights update method um (which is described in the next paragraph), and then they are sorted (lines 6–7). Eachbootstrap is enhanced by n_{al} examples selected ran-

---

Algorithm 1. Actively Balanced Bagging Algorithm

Input : LS training set; TS testing set; CLA component classifier learning algorithm; m_{bag} number of bootstrap samples; S bootstrap samples; w initial weights of examples in LS; um weights update method; m_{al} number of active learning iterations; n_{al} size of active learning batch

Output: C ensemble classifier

1 *Learning phase*;
2 for $i := 1$ *to* m_{bag} **do**
3 $\quad$ $C_i :=$ CLA (S_i) {generate a component classifier} ;

4 for $l := 1$ *to* m_{al} **do**
5 $\quad$ **for** $i := 1$ *to* m_{bag} **do**
6 $\quad\quad$ $w' :=$ updateWeights(w, C, um) {update weights used in sampling} ;
7 $\quad\quad$ sort all x with respect to $w'(x)$, so that $w'(x_1) \geq w'(x_2) \geq \ldots \geq w'(x_n)$;
8 $\quad\quad$ $S_i' :=$ random sample from $x_1, x_2, \ldots, x_{n_{al}}$ according to w' {rejection sampling from top n_{al} x sorted according to w'; $\alpha = w'(x_{n_{al}})$ } ;
9 $\quad\quad$ $S_i := S_i \cup S_i'$;
10 $\quad\quad$ $C_i :=$ CLA (S_i) {re-train a new component classifier} ;

11 *Classification phase*;
12 foreach x *in* TS **do**
13 $\quad$ $C(x) :=$ majority vote of $C_i(x)$, where $i = 1, \ldots, m$ {the prediction for example x is a combination of predictions of component classifiers C_i} ;

---

domly with rejection sampling according to $\alpha = w'(x_{n_{al}}) + \epsilon$, i.e., n_{al} random examples with weights w' higher than α are selected (lines 8–9). Parameter ϵ introduces an additional (after α) level of randomness into the sampling. Finally, after each bootstrap sample is enlarged, new component classifier C_i is trained resulting in new ensemble classifier C (line 10).

We consider four different weights update methods. The simplest method, called *margin* (m), is substituting the initial weights of examples with a decision margin between component classifiers in C. For a given example it is defined as: $m = 1 - \left| \frac{V_{maj} - V_{min}}{m_{bag}} \right|$, where V_{maj} is number of votes for majority class, V_{min} is number of votes for minority class, and m_{bag} is the number of component classifiers. Since the margin may not be directly reflecting the characteristic of imbalanced data (indeed under-sampling somehow should reduce bias of the classifiers) we consider aggregating it with additional factors. As a result, three variants of weights update methods are proposed. In the first variant, called, *margin and weight* (mw), new weight w' is a product of m and initial balancing weight w. Additionally we reduce the influence of w in subsequent iterations of active example selection, as active selection iteration index l is increasing. The reason for this reduction of influence is that we expect that margin m is improving with subsequent iterations, and thus initial weights w are becoming less important. More precisely, $mw = m \times w^{\left(\frac{m_{al} - l}{m_{al}} \right)}$.

Recall that both considered so far weights update methods produce bootstrap samples which, in the same iteration l, differ only according to randomization introduced by rejection sampling, i.e., weights $\boldsymbol{w'}$ are the same for each i. That is why, we consider yet another modification of m and mw, which makes $\boldsymbol{w'}$, and, consequently, each bootstrap dependent on performance of the corresponding component classifier. These two new update methods: *margin and component error* (mce), and *margin, weight and component error* ($mwce$) are defined, respectfully, as follows. $mce = m + 1_e \times w$, and $mwce = mw + 1_e \times w$. In this notation, 1_e is an indicator function defined so that $1_e = 1$ when a component classifier is making a prediction error on the example, and $1_e = 0$ otherwise.

4   Experiments

We consider two aims of the experiments. First, we check whether the predictive performance of Actively Balanced Bagging is improved in comparison to known well performing under-sampling extensions of bagging. To examine this part more generally we have decided to choose two efficient extensions based on different principles: Exactly Balanced Bagging (EBBag) [7] and Neighbourhood Balanced Bagging (NBBag) [4]. Our second aim is to compare different variants of active selection methods, which result in different versions of ABBag.

In order to examine both aims we use several synthetic and real-world imbalanced data sets representing various difficulty factors in the distribution of the minority class. All considered synthetic data sets, i.e., `flower`, and `paw`, were constructed, as described in [17]. They have a controlled level of difficulty, i.e., decomposition of the minority class into sub-concepts, overlapping area, presence of rare examples and outliers. The percentage of minority examples representing safe, borderline, rare and outlier examples are given in the suffix of the data set name. The examples from the minority class were generated randomly inside predefined spheres and the majority class examples were randomly distributed in an area surrounding them. Following similar motivations and analysis from works on difficulty imbalanced data [4,14], we selected UCI data sets, and additional `scrotal-pain` data set[1]. Due to page limits we redirect the reader to the description and characteristics of these data sets in [4]. Note that some of data sets contain multiple majority classes, which were aggregated into one class.

The experiments were carried out in two variants with respect to the first phase of the active selection. In the first variant, standard EBBag or under-sampling NBBag was considered. In the other variant, the size of each of the classes in bootstrap samples was reduced to 50% of the size of the minority class in the training set. Active selection parameters, used in the second phase, m_{al}, and n_{al} were chosen in a way, which enables the bootstrap samples constructed in ABBag to excess the size of standard under-sampling bootstrap by a factor not higher than two. The size of ensembles m_{bag}, in accordance with previous experiments [4], was always fixed to 50 J4.8 decision trees. NBBag parameters were

---

[1] We are grateful to prof. W. Michalowski and the MET Research Group from the University of Ottawa for allowing us to use `scrotal-pain` data set.

set the same as in [4]. All presented results are estimated by a stratified 10-fold cross-validation repeated three times to improve reproducibility. The number of repetitions is the same for all considered data sets for better comparability.

Due to page limits, we present results of two main experiments, which are good representatives of tendencies observed in other analyzed settings.[2] In Tables 1 and 2, we present values of G-mean measure, since we want to find a good trade-off between recognition in both classes [4]. More precisely, we show the best G-mean value, which can be achieved with a limited number of active

Table 1. G-mean of actively balanced 50% under-sampling EBBag

Data set	EBBag	m-EBBag	mce-EBBag	mw-EBBag	$mwce$-EBBag
abalone	79.486	79.486	79.486	79.486	79.574
breast-cancer	57.144	59.817	58.355	59.793	59.827
car	96.513	97.312	98.202	97.208	98.192
cleveland	70.818	74.823	72.347	75.027	71.940
cmc	64.203	65.229	64.900	65.435	64.786
ecoli	87.836	88.398	88.667	88.287	88.639
flower5-3d-10-20-35-35	0.000	53.445	52.992	52.768	53.809
flower5-3d-100-0-0-0	92.315	93.272	94.094	93.402	94.398
flower5-3d-30-40-15-15	77.248	77.867	78.591	77.780	78.543
flower5-3d-30-70-0-0	91.105	91.947	92.971	91.876	93.039
flower5-3d-50-50-0-0	91.966	92.311	93.414	92.381	93.558
haberman	62.908	65.551	65.423	65.909	65.378
hepatitis	78.561	79.302	80.169	79.427	81.044
paw3-3d-10-20-35-35	0.000	51.245	51.047	51.984	50.788
paw3-3d-100-0-0-0	90.857	93.004	94.193	92.926	94.312
paw3-3d-30-40-15-15	74.872	76.707	78.429	77.971	78.091
paw3-3d-30-70-0-0	88.545	91.159	91.535	90.827	91.438
paw3-3d-50-50-0-0	91.424	91.427	92.249	91.427	92.232
scrotal-pain	72.838	73.915	73.581	73.572	73.344
solar-flare	82.048	82.834	82.771	82.464	82.674
transfusion	66.812	66.847	66.812	67.607	66.812
vehicle	95.506	96.275	96.663	96.409	96.573
yeast	82.658	85.225	85.305	84.702	85.084
average rank	4.891	2.739	2.109	3.000	2.261

---

[2] We have published detailed results for specific values of parameters, on-line, in the appendix, http://www.cs.put.poznan.pl/jblaszczynski/ISMIS17/resABBag.pdf.

Table 2. G-mean of actively balanced under-sampling NBBag

Data set	NBBag	m-NBBag	mce-NBBag	mw-NBBag	$mwce$-NBBag
abalone	78.714	79.426	79.585	79.621	80.066
breast-cancer	58.691	62.037	62.130	62.223	62.201
car	96.200	97.094	97.414	97.126	97.724
cleveland	73.004	73.004	73.718	74.197	75.318
cmc	65.128	65.184	65.128	65.184	65.128
ecoli	88.581	88.867	88.581	88.867	88.581
flower5-3d-10-20-35-35	0.000	52.425	52.685	52.601	51.340
flower5-3d-100-0-0-0	92.373	93.244	93.185	93.048	93.998
flower5-3d-30-40-15-15	76.914	78.520	78.282	77.998	78.412
flower5-3d-30-70-0-0	91.120	92.508	93.019	92.044	92.997
flower5-3d-50-50-0-0	92.003	93.154	92.870	92.726	93.321
haberman	64.128	65.404	65.126	66.098	66.199
hepatitis	78.017	79.323	79.648	79.080	79.551
paw3-3d-10-20-35-35	0.000	52.294	51.545	50.043	50.825
paw-3d-100-0-0-0	90.122	92.915	93.420	93.101	94.346
paw-3d-30-40-15-15	63.966	76.500	76.423	76.987	77.921
paw-3d-30-70-0-0	87.208	90.707	90.438	90.296	90.730
paw-3d-50-50-0-0	91.317	91.603	91.803	91.527	91.973
scrotal-pain	73.205	74.836	76.003	75.515	75.336
solar-flare	83.435	83.738	83.929	83.435	83.605
transfusion	65.226	66.612	65.391	66.612	65.375
vehicle	95.339	96.102	96.922	96.491	97.437
yeast	84.226	85.368	85.444	85.359	84.605
average rank	4.870	2.739	2.435	2.870	2.087

learning iterations (we considered $m_{al} \leq 10$) for relatively small batches. The size of batch in experiments was set as percentage of the size minority class ($n_{al} = \{5\%, 10\%\}$). We show two variants of the first phase of the active selection. Each of them is appropriate for the analyzed type of ABBag. Table 1, presents results of active balancing of 50% under-sampling EBBag. While in Table 2, we present results of active balancing of standard under-sampling NBBag. The last row of Tables 1 and 2 contains average ranks calculated as in the Friedman test – the lower average rank, the better the classifier.

The first, general conclusion resulting from our experiments is that ABBag performs better than under-sampling extensions of bagging, both: EBBag, and NBBag. Let us treat EBBag, and NBBag as baselines in Tables 1 and 2,

respectively. The observed improvements of G-mean are statistically significant regardless of the version of ABBag. More precisely, following Friedman statistical test, each actively balanced EBBag has lower average rank than the baseline EBBag, and, similarly, each actively balanced NBBag has lower average rank than the baseline NBBag. Moreover, Friedman tests results in p-values $\ll$ 0.00001. According to Nemenyi post-hoc test, critical difference CD between average ranks is around 1.272. CD is thus higher than the difference between average ranks of each actively balanced EBBag and the baseline EBBag. An analogous observation holds for each actively balanced NBBag and the baseline NBBag. Some more detailed observations concerning results presented in Tables 1 and 2 may be given. The most striking difference in performance between baseline bagging extensions and ABBag is visible for synthetic data sets with the most difficult to learn distribution of safe, borderline, rare and outlier examples (i.e., data sets with suffix `10-20-35-35`). In that case, both baseline EBBag and NBBag are not able to learn the minority class at all. ABBag shows significantly better performance for these data sets, regardless of the weights update strategy.

Moving to the next research question on the role of the proposed active modifications, i.e., weights update methods in the active selection, we make the following observations. First, if we consider actively balanced EBBag, *margin and component error* weights update method, thus (*mce*-EBBag), has the best average rank. Nevertheless, *margin, weight and component error* weights update method, thus (*mwce*-EBBag), has the best value of median calculated for all G-mean in Table 1. These observations are, not statistically significant according to the critical difference and results of Wilcoxon test for a selected pair of classifiers. On the other hand, in all of experiments with actively balanced EBBag we note that one of the two weights update methods: *mce* or *mwce* yields the best results.

Similar observations are valid for actively balanced NBBag. In this case, the best weights update method according to average rank is *margin, weight and component error*, and thus (*mwce*-NBBag). On the other hand, *margin and component error* weights update method, and thus (*mce*-NBBag), has the best value of median. Again these observations are not statistically significant. However, Wilcoxon paired test between *mwce*-NBBag and *mw*-NBBag resulted in p-value close to 0.045, and *mwce*-NBBag has the higher value of median. We can take it as another indication that inclusion of *component error* inside the weights update method yields better G-mean results.

To sum up, we can interpret results obtained with all four different weights update methods applied in the active selection in the following way. First, they show that all considered elements of weights update strategy: margin of classifiers in ensemble, weight of example, and component error are important for improving ABBag performance. Second, two combinations of considered elements tend to give better results than the others. These are: *margin and component error* (*mce*), and *margin, weight and component error* (*mwce*).

5   Conclusions

In this paper we examined the following research question: is it possible to improve classification performance of an under-sampling bagging ensemble with an active learning strategy? To address this question we introduced a new approach, called Actively Balanced Bagging (ABBag). The proposed active selection of examples involves iterative updating of bootstraps with batches composed of examples selected from the training set. These examples were sampled according to the distribution of weights, which expressed: a decision margin of ensemble votes, balancing of example class distribution in the training set and/or in its neighbourhood, and prediction errors of component classifiers.

The results of experiments have clearly shown that the active selection of examples has improved G-mean performance of considered under-sampling bagging extensions, which are known to be very good classifiers for imbalanced data. Another important observation resulting from experiments is that the active selection strategy performs best when it integrates the ensemble disagreement factor (typical for active learning) with information on class distribution in imbalanced data and prediction error of component classifiers.

We hope that this study may open future research lines on adapting active learning strategies to improve ensemble classifiers.

References

1. Abe, N., Mamitsuka, H.: Query learning strategies using boosting and bagging. In: Proceedings of 15th International Conference on Machine Learning, pp. 1–10 (2004)
2. Aggarwal, C., Kong, X., Gu, Q., Han, J., Yu, P.: Active learning: a survey. In: Data Classification: Algorithms and Applications, pp. 571–606. CRC Press (2015)
3. Błaszczyński, J., Stefanowski, J., Idkowiak, L.: Extending bagging for imbalanced data. In: Burduk, R., Jackowski, K., Kurzynski, M., Wozniak, M., Zolnierek, A. (eds.) Proceedings of the 8th CORES 2013. AISC, vol. 226, pp. 269–278. Springer, Heidelberg (2013)
4. Błaszczyński, J., Stefanowski, J.: Neighbourhood sampling in bagging for imbalanced data. Neurocomputing **150A**, 184–203 (2015)
5. Borisov, A., Tuv, E., Runger, G.: Active batch learning with Stochastic Query-by-Forest (SQBF). In: JMLR Workshop on Active Learning and Experimental Design 2011, vol. 16, pp. 59–69 (2011)
6. Branco, P., Torgo, L., Ribeiro, R.: A survey of predictive modeling on imbalanced domains. ACM Comput. Surv. (CSUR) **49**(2), 31 (2016)
7. Chang, E.: Statistical learning for effective visual information retrieval. In: Proceedings of ICIP 2003, pp. 609–612 (2003)
8. Ertekin, S., Huang, J., Bottou, L., Giles, L.: Learning on the border: active learning in imbalanced data classification. In: Proceedings of the 16th ACM Conference on Information and Knowledge Management, pp. 127–136 (2007)

9. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **99**, 1–22 (2011)
10. He, H., Yungian, M. (eds.): Imbalanced Learning. Foundations, Algorithms and Applications. IEEE - Wiley, Hoboken (2013)
11. Hido, S., Kashima, H.: Roughly balanced bagging for imbalance data. In: Proceedings of the SIAM International Conference on Data Mining, pp. 143–152 (2008) - An extended version in Stat. Anal. Data Mining **2**(5–6), 412–426 (2009)
12. Khoshgoftaar, T., Van Hulse, J., Napolitano, A.: Comparing boosting and bagging techniques with noisy and imbalanced data. IEEE Trans. Syst. Man Cybern.-Part A **41**(3), 552–568 (2011)
13. Napierala, K., Stefanowski, J.: The influence of minority class distribution on learning from imbalance data. In: Corchado, E., Snášel, V., Abraham, A., Woźniak, M., Graña, M., Cho, S.-B. (eds.) HAIS 2012. LNCS, vol. 7209, pp. 139–150. Springer, Heidelberg (2012)
14. Napierala, K., Stefanowski, J.: Types of minority class examples and their influence on learning classifiers from imbalanced data. J. Intell. Inf. Syst. **46**(3), 563–597 (2016)
15. Settles, B.: Active learning literature survey. Technical report, Computer Sciences Technical Report (2009)
16. Yang, Y., Ma, G.: Ensemble-based active learning for class imbalance problem. J. Biomed. Sci. Eng. **3**(10), 1022–1029 (2010)
17. Wojciechowski, S., Wilk, S.: The generator of synthetic multi-dimensional data. Poznan University of Technology Report RB-16/14 (2014)