Evaluating and Comparing Classifiers: Review, Some Recommendations and Limitations

Katarzyna Stąpor(✉)

Institute of Computer Science, Silesian Technical University, Gliwice, Poland
katarzyna.stapor@polsl.pl

Abstract. Performance evaluation of supervised classification learning method related to its prediction ability on independent data is very important in machine learning. It is also almost unthinkable to carry out any research work without the comparison of the new, proposed classifier with other already existing ones. This paper aims to review the most important aspects of the classifier evaluation process including the choice of evaluating metrics (scores) as well as the statistical comparison of classifiers. Critical view, recommendations and limitations of the reviewed methods are presented. The article provides a quick guide to understand the complexity of the classifier evaluation process and tries to warn the reader about the wrong habits.

Keywords: Supervised classification · Classifier evaluation · Performance metrics · Statistical classifier comparison

1 Introduction

In a supervised classification problem one aims to learn a classifier from a dataset $U = \{(x^{(1)}, t^{(1)}), \ldots, (x^{(n)}, t^{(n)})\}$ of n labeled data instances and each instance $x^{(i)}$ is characterized by d predictive variables/features, $X = (X_1, \ldots, X_d)$, and a class T to which it belongs. This dataset is obtained from a physical process described by an unknown probability distribution $f(X, T)$. Then, the learned classifier, after *evaluating* its *quality* (usually on test dataset), can be used to classify new samples, i.e. to obtain their unknown class labels. We do not make here a distinction between a *classifier* (being a function that maps an input feature space to a set of class labels) and a *classification learning algorithm* which is a general methodology that can be used, given a specific dataset, to learn a specific classifier. Theoretical background on supervised classification problem as well as the whole description of classifier construction process can be found in many books on machine learning and pattern recognition (see for example [2,8,31,33,34,44,47,49]). Usually, the problem of evaluating a new classifier is tackled by using the *score* that try to summarize the specific conditions of interest. *Classification error* and *accuracy* are widely used scores in the classification problems. In practice, classification error must be *estimated* from all the available samples.

M. Kurzynski et al. (eds.), *Proceedings of the 10th International Conference on Computer Recognition Systems CORES 2017*, Advances in Intelligent Systems and Computing 578, DOI 10.1007/978-3-319-59162-9_2

The *k-fold cross-validation*, for example, is on of the most frequently used such estimation methods. Then, questions are whether such a new, proposed classifier (or enhancement of the existing one) yields an improved score over the competitor classifier (or classifiers) or the state of the art. It is almost impossible now to do any research work without an experimental section where the score of a new classifier is tested and compared with the scores of the existing ones. This last step also requires the *selection of datasets* on which the compared classifiers are learned and evaluated. The purpose of dataset selection step should not be to demonstrate classifiers superiority to another in all cases, but rather to identify its areas of strengths with respect to domain characteristics. This paper is focused only on a supervised classification problem as defined in the beginning. Other types of classification such as *classification from data streams* or *multi-label classification* are not addressed here, since they may impose specific conditions to the calculation of the score (for the most important reference in evaluating (static) data streams, see for example [15]). The whole evaluation process of a classifier should include the following steps [41]:

1. choosing an evaluation metric (i.e. a score) according to the properties of a classifier,
2. deciding the score estimation method to be used,
3. checking whether the assumptions made by (1) and (2) are fulfilled,
4. running the evaluation method and interpret the results with respect to the domain,
5. compare a new classifier with the existing ones selected according to the different criteria, for example problem dependent; this step requires *selection of datasets*.

The main purpose of this paper is to provide the reader with a better understanding about the overall classifier evaluation process. As there is no fixed, concrete recipe for the classifier evaluation procedure, we believe that this paper will facilitate the researcher in the machine learning area to decide which alternative to choose for each specific case. The paper is set up as follows. In Sect. 2 we describe measures of classifier quality while in Sect. 3, a short overview of their estimation methods. Section 4 focuses on statistical methods for classifier quality comparison. Finally, in Sect. 5 we conclude giving some recommendations.

2 Measures of Classifier Quality

Usually the problem of evaluating a new classifier (i.e. measuring its quality) is tackled by using the *score* that try to summarize the specific conditions of interest when evaluating a classifier. There may be many scores according to how we aim to quantify classifiers behavior. In this section, we only present some of the most extended scores. Typical scores for measuring the performance of a classifier are *accuracy* and *classification error*, which for a two-class problem can be easily derived from a 2×2 confusion matrix as that given in Table 1. These scores can be computed as:

$$Acc = (TP + TN)/(TP + FN + TN + FP)$$

$$Err = (FP + FN)/(TP + FN + TN + FP)$$

Sometimes, accuracy and classification error are selected without considering in depth whether it is the most appropriate score to measure the quality of a classifier for the classification problem at hand. When both class labels are relevant and the proportion of data samples for each class is very similar, these scores are a good choice. Unfortunately, equally class proportions are quite rare in real problems. This situation is known as the *imbalance problem* [29,45]. Empirical evidence shows that accuracy and error rate are biased with respect to data imbalance: the use of these scores might produce misleading conclusions since they do not take into account misclassification costs, the results are strongly biased to favor the majority class, and are sensitive to class skews. In some application domains, we may be interested in how our classifier classifies only a part of the data. Examples of such measures are: *True positive rate (Recall or Sensitivity):* TPrate = TP/(TP+FN), *True negative rate (Specificity):* TNrate = TN/(TN + FP), *False positive rate:* FPrate = FP/(TN + FP), *False negative rate:* FNrate = FN/(TP + FN), Precision = TP/(TP + FP). Shortcomings of the accuracy or error rate have motivated search for new measures which aim to obtain a trade-off between the evaluation of the classification ability on both positive and negative data samples. Some straightforward examples of such alternative scores are: the *harmonic mean* between *Recall* and *Precision* values: *F-measure* = 2 × TPrate × Precision/(TPrate + Precision), and the *geometric mean* of accuracies measured separately on each class: *G-mean* = $\sqrt{\text{TPrate} \times \text{TNrate}}$ [3]. Harmonic and geometric means are symmetric functions that give the same relevance to both components. There are other proposals that try to enhance one of the two components of the mean. For instance, index of balaned accuracy [18], the *adjusted geometric mean* [1], the *optimized precision OP* from [37] computed as: OP = Acc − (|TNrate − TPrate|/(TNrate + TPrate)), and F-score [30]:

$$F\text{-}score = \frac{(\beta^2 + 1)\text{Precision} \times \text{TPrate}}{\beta^2 \times \text{Precision} + \text{TPrate}}$$

A parameter β can be tuned to obtain different trade-offs between both components. When a classifier classifies an instance into a wrong class group, a loss is incurred. *Cost-sensitive learning* [10] aims to minimize this loss incurred by the classifier. The above introduced scores use the 0/1 *loss function*, i.e. they treat all the different types of misclassification as equally severe. The *cost matrix* can be used if the severity of misclassifications can be quantified in terms of costs. Unfortunately, in real applications, specific costs are difficult to obtain. In such situations, however, the described above scores may be useful since they may also be used to set more relevance into the costliest misclassification: minimizing the cost may be equivalent to optimal trade-off between *Recall* and *Specificity* [7]. When the classification costs cannot be accessed, another most widely-used techniques for the evaluation of classifiers is the *ROC curve* [4,11], which is a

Table 1. Confusion matrix for a two-class problem

	Predicted positive	Predicted negative
Positive class	True Positive (TP)	False Negative (FN)
Negative class	False Positive (FP)	True Negative (TN)

graphical representation of Recall versus *FPrate (1-Specificity)*. The information about classification performance in the ROC curve can be summarized into a score known as AUC (Area under the ROC curve) which is more insensitive to skewness in class distribution since it is a trade-off between Recall and Specificity [43]. However, recent studies have shown that AUC is a fundamentally incoherent measure since it treats the costs of misclassification differently for each classifier. This is undesirable because the cost must be a property of the problem, not of the classification method. In [21,22], the *H measure* is proposed as an alternative to AUC. While all of the scores described above in this section are appropriate for two-class imbalanced learning problems, some of them can be modified to accommodate the multi-class imbalanced learning problems [23]. For example [46] extends the G-mean definition to the geometric mean of Recall values of every class. Similarly, in [12] they defined *mean F-measure* for multi-class imbalance problem. The major advantage of this measure is that it is insensitive to class distribution and error costs. However, it is now an open question if such extended scores for multi-class classification problem are appropriate on scenarios where there exist multiple minority and multiple majority classes [40]. In ([20] they proposed the *M measure*, a generalization approach that aggregates all pairs of classes based on the inherent characteristics of the AUC. In this paper, we focus on the scores since they are popular way to measure classification quality. But these measures do not capture all the information about the quality of classification methods some graphical methods may do. However, the use of quantitative measures of quality makes the comparison among the classifiers easier (for more information on graphical methods see for example [9,30,36]). The presented list of scores is by no means exhaustive. The described scores are focused only on the evaluating the performance of a classifier. However, there are other important aspects of classification such as robustness to noise, scalability, stability under data shifts, etc. which are not addressed here.

3 Quality Estimation Methods

Various methods are commonly used to estimate classification error and the other described classifier scores (the review of estimation methods can also be found in the mentioned literature on machine learning). *Holdout method* of estimation of classification error divides randomly the available dataset into independent training and testing subsets which are then used for learning and evaluating a classifier. This method gives a pessimistically biased error estimate (calculated as a ratio of misclassified test samples to a size of test subset), moreover it depends

on a particular partitioning of a dataset. These limitations are overcome with a family of *resampling methods*: cross validation (random sub-sampling, k-fold cross-validation, leave-one-out) and bootstrap. *Random subsampling* performs k random data splits of the entire dataset into training and testing subsets. For each data split, we retrain a classifier and then estimate error with test samples. The true error estimate is the average of separate errors obtained from k splits. The *k-fold cross-validation* creates a k fold partition of the entire dataset once: Then, for each of k experiments, it uses $(k-1)$ folds for training and a different fold for testing. The classification error is estimated as the average of separate errors obtained from k experiments. It is approximately unbiased, although at the expense of an increase in the variance of the estimate. *Leave-one-out* is the degenerate case of k-fold cross-validation where k is chosen as the total number of samples. This results in the unbiased error estimate, but have large variance. In the *bootstrap* estimation, we randomly select with replacement the samples and use this set for training. The remaining samples that were not selected for training are used for testing. We repeat this procedure k times. The error is estimated as the average error on test samples from k procedures. The benefit of this method is its ability to obtain accurate measures of both bias and variance of classification error estimate.

4 Statistical Comparison of Classifiers

The comparison of the scores obtained by two or more classifiers in a set of problems is a central task in machine learning, so it is almost impossible to do any research work without an experimental section where the score of a new classifier is tested and compared with the scores of the existing ones. When the differences are very clear (e.g., when the classifier is the best in all the problems considered), the direct comparison of the scores may be enough. But in most situations, a direct comparison may be misleading and not enough to draw sound conclusions. In such situations, the statistical assessment of the scores such as hypothesis testing is required. Statistical tests arise with the aim of giving answers to the above mentioned questions, providing more precise assessments of the obtained scores by analyzing them to decide whether the observed differences between the classifiers are real or random. However, although the statistical tests have been established as a basic part of classifier comparison task, they are not a definitive tool, we have to be aware about their limitations and misuses. The statistical tests for comparing classifiers are usually bound to a specific estimation method of classifier score. Therefore, the selection of a statistical test is also conditioned by this estimation method. For the comparison of two classifiers on one dataset, the situation which is very common in machine learning problems, the *corrected resampled t test* has been suggested in the literature [35]. This test is associated with a repeated estimation method (for example holdout): in i-th of the m iterations, a random data partition is conducted and the values for the scores $A_{k1}^{(i)}$ and $A_{k2}^{(i)}$ of compared classifiers $k1$ and $k2$, are obtained. The statistic is:

$$t = \frac{\overline{A}}{\sqrt{\left(\frac{1}{m} + \frac{N_{\text{test}}}{N_{\text{train}}}\right) \cdot \sum_{i=1}^{m} \frac{(A^{(i)} - \overline{A})^2}{m-1}}}$$

where $\overline{A} = \frac{1}{m}\sum_{i=1}^{m} A^{(i)}$, $A^{(i)} = (A_{k1}^{(i)} - A_{k2}^{(i)})$, N_{test}, N_{train} are the number of samples in the test and train partitions. The second parametric test that can be used in this scenario whose behavior, however, has not been studied as for previous, is the *corrected t test for repeated cross-validation* [3]. These tests assume the data follow the normal distribution which should be first checked using the suitable normality test. A non-parametric alternative for comparing two classifiers that is suggested in the literature is *McNemars test* [26]. For the comparison of two classifiers on multiple datasets the *Wilcoxon signed-ranks test* [26] is widely recommended. It ranks the differences $d_i = A_{k1}^{(i)} - A_{k2}^{(i)}$ between scores of two classifiers $k1$ and $k2$ obtained on i-th of N datasets, ignoring the signs. The test statistic of this test is:

$$T = \min(R^+, R^-)$$

where:

$$R^+ = \sum_{d_i>0} \text{rank}(d_i) + \frac{1}{2}\sum_{d_i=0} \text{rank}(d_i), R^- = \sum_{d_i<0} \text{rank}(d_i) + \frac{1}{2}\sum_{d_i=0} \text{rank}(d_i)$$

are the sums of ranks on which the $k2$ classifier outperforms $k1$, respectively. Ranks $d_i = 0$ are split evenly among the sums. Other test that can be used is the sign test, but it is much weaker than the Wilcoxon signed-ranks test. Comparison among multiple classifiers on multiple datasets arise in machine learning when a new proposed classifier is compared with the state of the art. For this situation, the general recommended methodology is as follows [5,6,16,39,41]. First, we apply an omnibus test to detect if at least one of the classifiers performs different than the others. *Friedman nonparametric test* [14] with *Iman-Davenport extension* [28] is probably the most popular omnibus test. It is a good choice when comparing more than five different classifiers. Let R_{ij} be the rank of the j-th of K classifiers on the i-th of N data sets and

$$R_j = \frac{1}{N}\sum_{i=1}^{N} R_{ij}$$

is the mean rank of j-th classifier. The test compares the mean ranks of the classifiers and is based on the test statistic:

$$F_F = \frac{(N-1)\chi_F^2}{N(K-1) - \chi_F^2} \quad \chi_F^2 = \frac{12N}{K(K+1)}\left[\sum_{j=1}^{K} R_j^2 - \frac{K(K+1)^2}{4}\right]$$

which follows a F distribution with $(K-1)$ and $(K-1)(N-1)$ degrees of freedom. For the comparison of five or less different classifiers, *Friedman aligned*

ranks [17] or the *Quade* test [25,38] are the more powerful alternatives. Second, if we find such a significant difference, then we apply a pair-wise test with the corresponding post-hoc correction for multiple comparisons. For the described above Friedman test, comparing the r-th and s-th classifiers is based on the mean ranks and has the form:

$$z = \frac{R_r - R_s}{\sqrt{\frac{K(K+1)}{6N}}}$$

The z value is used to find the corresponding probability from the table of normal distribution, which is then compared with an appropriate significance level α. As performing pair-wise comparisons is associated with a set or family of hypotheses, the value of α must be adjusted for controlling the family-wise error [42]. There are multiple proposals in the literature to adjust the significance level α: Holm [27], Hochberg [24], Finner [13]. The results of pair-wise comparisons, often, give not disjoint groups of classifiers. In order to identify disjoint, homogenous groups, in [19] they apply special cluster analysis approach. Their method results in dividing K classifiers into groups in such a way that classifiers belonging to the same group do not significantly differ with respect to the chosen distance.

5 Recommendations and Conclusions

This paper covers the basic steps of classifier evaluation process, focusing mainly on the evaluation metrics and conditions for their proper usage as well as the statistical comparison of classifiers. The evaluation of classification performance is very important to the construction and selection of classifiers. The vast majority of the published articles use the *accuracy* (or *classification error*) as the score in the classifier evaluation process. But these two scores may be appropriate only when the datasets are balanced and the misclassification costs are the same for false positives and false negatives. In the case of skew datasets, which is rather typical situation, the accuracy/error rate is questionable and other scores such as *Recall*, *Specificity*, *Precision*, *Optimized Precision*, *F-score*, *geometric* or *harmonic means*, H or M measures are more appropriate. The comparison of two classifiers on a single dataset is generally unsafe due to the lack of independence between the obtained score values. Thus, the *corrected* versions of the *resampled t test* or *t test* for *repeated cross-validation* are more appropriate. *McNemars test*, being non-parametric, does not make the assumption about distribution of the scores (like the two previous tests) but it does not directly measure the variability due to the choice of the training set nor the internal randomness of the learning algorithm. When comparing two classifiers on multiple datasets (especially from different sources), the measured scores are hardly commensurable. Therefore, the *Wilcoxon signed-rank test* is more appropriate. Regarding the comparison of multiple classifiers on multiple datasets, if the number of classifiers involved is higher than five, the use of the *Friedman test with Iman and Davenport extension* is recommended. When this number is low, four or five, *Friedman aligned ranks*

and the *Quade test* are more useful. If the null hypothesis has been rejected, we should proceed with a *post-hoc test* to check the statistical differences between pairs of classifiers. The last but not least conclusion follows from no free lunch theorem [48] which states that for any two classifiers, there are as many classification problems for which the first classifier performs better than the second as vice versa. Thus, it does not make sense to demonstrate that one classifier is, on average, better than the others. Instead, we should focus our attention on exploring the conditions of the classification problems which make our classifier to perform better or worse than others. We must carefully choose the datasets to be included in the evaluation process to reflect the specific conditions, for example class imbalance, classification cost, dataset size, application domain, etc. In other words, the choice of the datasets should be guided in order to identify specific conditions that make a classifier to perform better than others. Summarizing, this review tries to provide the reader with a better understanding about the overall process of comparison in order to decide which alternative to choose for each specific case. We believe, that this review can improve the way in which researchers and practitioners in machine learning contrast the results achieved in their experimental studies using statistical methods.

References

1. Batuvita, R., Palade, V.: A new performance measure for class imbalance learning: application to bioinformatics problem. In: Proceedings of 26th International Conference Machine Learning and Applications, pp. 545–550 (2009)
2. Bishop, C.: Pattern Recognition and Machine Learning. Springer, New York (2006)
3. Bouckaert, R.: Estimating replicability of classifier learning experiments. In: Proceedings of the 21st Conference on ICML. AAAI Press (2004)
4. Bradley, P.: The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recogn. **30**, 1145–1159 (1997)
5. Dietterich, T.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput. **10**, 1895–1924 (1998)
6. Demsar, J.: Statistical comparison of classifiers over multiple data sets. J. Mach. Learn. Res. **7**, 1–30 (2006)
7. Dmochowski, J., et al.: Maximum likelihood in cost-sensitive learning: model specification, approximation and upper bounds. J. Mach. Learn. Res. **11**, 3313–3332 (2010)
8. Duda, R., Hart, P., Stork, D.: Pattern Classification and Scene Analysis. Wiley, New York (2000)
9. Drummond, C., Holte, R.: Cost curves: an improved method for visualizing classifier performance. Mach. Learn. **65**(1), 95–130 (2006)
10. Elkan, C.: The foundation of cost-sensitive learning. In: Proceedings of 4th International Conference Artificial Intelligence, vol. 17, pp. 973–978 (2001)
11. Fawcett, T.: An introduction to ROC analysis. Pattern Recogn. Lett. **27**(8), 861–874 (2006)
12. Ferri, C., et al.: An experimental comparison of performance measures for classification. Pattern Recogn. Lett. **30**(1), 27–38 (2009)
13. Finner, H.: On a monotonicity problem in step-down multiple test procedures. J. Am. Stat. Assoc. **88**, 920–923 (1993)

14. Friedman, M.: A comparison of alternative tests of significance for the problem of m rankings. Ann. Math. Stat. **11**, 86–92 (1940)
15. Gama J., et. al.: On evaluating stream learning algorithms. Mach. Learn., pp. 1–30 (2013)
16. Garcia, S., Herrera, F.: An extension on statistical comparison of classifiers over multiple datasets for all pair-wise comparisons. J. Mach. Learn. Res. **9**(12), 2677–2694 (2008)
17. Garcia, S., Fernandez, A., Lutengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in the computational intelligence and data mining: experimental analysis of power. Inf. Sci. **180**(10), 2044–2064 (2010)
18. García, V., Mollineda, R.A., Sánchez, J.S.: Index of balanced accuracy: a performance measure for skewed class distributions. In: Araujo, H., Mendonça, A.M., Pinho, A.J., Torres, M.I. (eds.) IbPRIA 2009. LNCS, vol. 5524, pp. 441–448. Springer, Heidelberg (2009). doi:10.1007/978-3-642-02172-5_57
19. Górecki, T., Krzyśko, M.: Regression methods for combining multiple classifiers. Commun. Stat. Simul. Comput. **44**, 739–755 (2015)
20. Hand, D., Till, R.: A simple generalization of the area under the ROC curve for multiple class classification problems. Mach. Learn. **45**, 171–186 (2001)
21. Hand, D.: Measuring classifier performance: a coherent alternative to the area under the ROC curve. Mach. Learn. **77**, 103–123 (2009)
22. Hand, D., Anagnostopoulos, C.: A better beta for the H measure of classification performance. Pattern Recogn. Lett. **40**, 41–46 (2014)
23. He, H., Garcia, E.: Learning from imbalanced data. IEEE Trans Data Knowl. Eng. **21**(9), 1263–1284 (2009)
24. Hochberg, Y.: A sharper Bonferroni procedure for multiple tests of significance. Biometrika **75**, 800–802 (1988)
25. Hodges, J.L., Lehmann, E.L.: Ranks methods for combination of independent experiments in analysis of variance. Ann. Math. Stat. **33**, 482–487 (1962)
26. Hollander, M., Wolfe, D.: Nonparametric Statistical Methods. Wiley, New York (2013)
27. Holm, S.: A simple sequentially rejective multiple test procedure. Scand. J. Stat. **6**, 65–70 (1979)
28. Iman, R., Davenport, J.: Approximations of the critical region of the Friedman statistic. Comput. Stat. **9**(6), 571–595 (1980)
29. Japkowicz, N., Stephen, N.: The class imbalance problem: a systematic study. Intell. Data Anal. **6**(5), 40–49 (2002)
30. Japkowicz, N., Shah, M.: Evaluating learning algorithms: a classification perspective. Cambridge University Press, Cambridge (2011)
31. Krzyśko, M., Wołyński, W., Górecki, T., Skorzybut, M.: Learning Systems. In: WNT, Warszawa (2008) (in Polish)
32. Kubat, M., Matwin, S.: Adressing the curse of imbalanced training sets: one-sided selection. In: Proceedings of the 14th ICML, pp. 179–186 (1997)
33. Kurzyński, M.: Pattern Recognition. Statistical Approach. Wrocław University Technology Press, Wrocław (1997) (in Polish)
34. Malina, W., Śmiatacz, M.: Pattern Recognition. EXIT Press, Warszawa (2010) (in Polish)
35. Nadeau, C., Bengio, Y.: Inference for the generalization error. Mach. Learn. **52**(3), 239–281 (2003)
36. Prati, R., et al.: A survey on graphical methods for classification predictive performance evaluation. IEEE Trans. Knowl. Data Eng. **23**(11), 1601–1618 (2011)

37. Ranavana, R., Palade, V.: Optimized precision: a new measure for classifier performance evaluation. In: Proceedings of the 23rd IEEE International Conference on Evolutionary Computation, pp. 2254–2261 (2006)
38. Quade, D.: Using weighted rankings in the analysis of complete blocks with additive block effects. J. Am. Stat. Assoc. **74**, 680–683 (1979)
39. Salzberg, S.: On comparing classifiers: pitfalls to avoid and recommended approach. Data Min. Knowl. Disc. **1**, 317–328 (1997)
40. Sánchez-Crisostomo, J.P., Alejo, R., López-González, E., Valdovinos, R.M., Pacheco-Sánchez, J.H.: Empirical analysis of assessments metrics for multi-class imbalance learning on the back-propagation context. In: Tan, Y., Shi, Y., Coello, C.A.C. (eds.) ICSI 2014. LNCS, vol. 8795, pp. 17–23. Springer, Cham (2014). doi:10.1007/978-3-319-11897-0_3
41. Santafe, G., et al.: Dealing with the evaluation of supervised classification algorithms. Artif. Intell. Rev. **44**, 467–508 (2015)
42. Shaffer, J.P.: Multiple hypothesis testing. Annu. Rev. Psychol. **46**, 561–584 (1995)
43. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Inf. Proc. Manag. **45**, 427–437 (2009)
44. Stąpor, K.: Classification methods in computer vision. In: PWN, Warszawa (2011) (in Polish)
45. Sun, Y., et al.: Classification of imbalanced data: a review. Int. J. Pattern Recogn. Artif. Intell. **23**(4), 687–719 (2009)
46. Sun, Y., et. al.: Boosting for learning multiple classes with imbalanced class distribution. In: Proceedings of International Conference on Data Mining, pp. 592–602 (2006)
47. Tadeusiewicz, R., Flasiński, M.: Pattern recognition. In: PWN, Warszawa (1991) (in Polish)
48. Wolpert, D.: The lack of a priori distinctions between learning algorithms. Neural Comput. **8**(7), 1341–1390 (1996)
49. Woźniak, M.: Hybrid classifiers. Methods of Data, Knowledge and Classifier Combination. SCI, vol. 519, Springer, Heidelberg (2014)