
Dealing with Overlap and Imbalance: A New Metric and Approach

Zalán Borsos · Camelia Lemnaru · Rodica Potolea

Abstract This paper addresses learning in complex scenarios involving imbalance and overlap. We propose a novel measure, the Augmented R-value, for estimating the level of overlap in the data. It improves an existing model-based measure, by including the data imbalance in the estimation process. We provide both a theoretical demonstration as well as empirical validations of the new metric's efficacy in estimating the overlap level. Another contribution of the present paper is to propose a collection of meta-features to be used in conjunction with a meta-learning strategy for predicting the most suitable classifier for a given problem. The evaluations performed on a well-known collection of benchmark problems have shown that the meta-learning approach achieves superior results to the manual classifier selection process normally carried out by data scientists. The analysis of the results obtained by the meta-feature selection step has confirmed the power of the Augmented R-value in predicting the expected performance of classifiers in such complex classification scenarios. Also, we found that the overlap is a more serious factor affecting the performance of classifiers than imbalance.

Keywords imbalance · overlap · Augmented R-value · meta-classification

Technical University of Cluj-Napoca
Dept. of Computer Science, 26 Baritiu Street, Room C9, 400027,
Cluj-Napoca, Romania
Tel.: +40 264 402389
Fax: +40 264 594491
E-mail: zalan.borsos@student.utcluj.ro

Published in Pattern Analysis and Applications
The final publication is available at Springer via
<http://dx.doi.org/10.1007/s10044-016-0583-6>

1 Introduction

One of the current important challenges in data mining research is classification under an imbalanced data distribution. This issue appears when a classifier has to identify a rare, but important case. Domains in which class imbalance is prevalent include fraud or intrusion detection, medical diagnosis, risk management, text classification and information retrieval [9]. In such domains, all traditional classifiers fail to achieve a satisfactory performance level, due to several causes, such as the use of an inappropriate optimization criterion, which favors the identification of the majority cases, or the co-occurrence of other data-related factors/phenomena, which in conjunction with the data imbalance accelerate the performance drop beyond levels which could be reached by their combination, were these phenomena independent. One such data-related factor is the overlap of the class boundaries. Recent studies [11, 13] attempt to characterize the joint expected effect of data imbalance and overlap. Their findings suggest that, in isolation, overlap degrades performance more severely than imbalance. However, when the two co-occur, their joint impact on performance is more serious than expected. If the imbalance problem has been extensively studied within the scientific community, the overlap problem has received comparatively less attention.

This paper focuses on providing a meta-learning based solution to the challenging classification scenario involving imbalance and overlap. We propose a minimal set of meta-features which capture important dataset characteristics, such as imbalance, overlap and complexity, and enhance the correct choice of the best classifier for the specific problem. We propose a novel overlap metric, which adapts a previous general metric to imbalanced scenarios. We show that the newly proposed metric captures the severity of the problem better than the initial formulation. We perform an extensive experimental evaluation of the proposed meta-learning approach, both with and without a prominent preprocessing strategy for imbalanced learning problems - SMOTE [9]. Our results indicate that the meta-learner is able to correctly identify the most appropriate classifier. Also, we attempt to validate the proposed feature set empirically, and find that the overlap and the complexity-related measures are the most important, while the imbalance ratio is the least significant.

The rest of the paper is organized as follows: the next section briefly presents the problem we address in the paper: the joint occurrence of the data imbalance and overlap. Section 3 introduces a new metric for estimating data overlap in imbalanced problems, which extends and improves an existing overlap measure. Section 4 presents a meta-learning based strategy we consider for handling complex real scenarios involving data imbalance and overlap. Section 5 de-

scribes the experimental evaluations performed on the new overlap metric and the meta-learning strategy and discusses the results obtained. Section 6 presents several important aspects related to imbalanced classification and briefly presents the meta-learning-driven classifier selection problem; it is intended as an overview of the domain for readers less acquainted with it. Section 7 presents the concluding remarks.

2 Imbalance and Overlap

A classification problem is imbalanced if, in the available data, a certain class is represented by a very small number of instances compared to the other classes [18]. In practice, the problem is generally addressed with two-class problems, multi-class problems being transformed into binary. As the minority instances are of greater interest, they are referred to as positive instances (positive class); the majority class is referred to as the negative class. Imbalanced problems constitute a challenge due to the fact that most traditional classifiers are affected by the class imbalance problem to some extent [18, 34]. Moreover, since classifiers possess separate biases, they respond differently to different data imbalance-related factors.

Initial efforts to study the loss of performance in imbalance scenarios focused on characterizing the skewed data distribution, via the imbalance ratio (IR), defined as the ratio between the number of cases of the majority and the minority of cases. Also, the role of training set size and concept complexity in imbalance scenarios has been relatively early acknowledged [19], and a meta-feature which attempts to estimate their joint occurrence has been proposed in [20]. More recently, the focus has started to shift towards the analysis of several data intrinsic characteristics, which although do not form a canonical set of data-related issues, have been shown to bear an important role in the level of performance which can be achieved by the classifiers [24].

Overlapping of the class separation boundaries is such a data characteristic. It appears when regions of the data contain similar quantities of training data from every class. Consequently, classifiers have difficulties distinguishing between the two classes in such areas. Experiments performed on artificial datasets have indicated that the imbalance ratio in the overlapping area has a greater influence on performance than the size of the overlapping area [13]. Also, in [11] the authors analyze the SVM behavior in scenarios considering imbalance, small sample size and overlap. Their results reveal that overlap is a more serious problem than the imbalance. However, when the two co-occur, the SVM performance degrades significantly (more than the accumulated effect of the individual factors).

We believe the co-occurrence of imbalance and overlap to be an important problem for several reasons: first, most

real world problems possess a certain level of class boundary overlap and imbalance; secondly, their co-occurrence seriously affects classification performance, as revealed by numerous research papers; third, we believe this phenomenon can be characterized quantitatively, and thus the behavior of classifiers can be improved in such scenarios. A broader discussion on the imbalance problem and other intrinsic data characteristics is presented in section 6.

3 Augmented R-value

One of the important results of the current research is the proposal of an overlap measure which characterizes the level of overlap present in an imbalanced problem. Our proposed measure - the *Augmented R-value* - adapts the existing R-value overlap measure, introduced in [26]. We present a formal proof, as well as an intuitive motivation of its efficacy. The experiments performed in Section 5 on both artificially generated and benchmark datasets further support the validity of the metric.

The original R-value is based on the intuition that :

Definition 1. An instance from class c belongs to an overlapping region if out of its k nearest neighbors, at least $\theta + 1$ belong to a class other than c .

The R-value of a class is estimated as the portion of its instances belonging to overlapping regions, an example of which is represented in Figure 1 by the gray band. The evaluations performed in [26] indicate that the R-value correlates quite well with classifier *accuracy*.

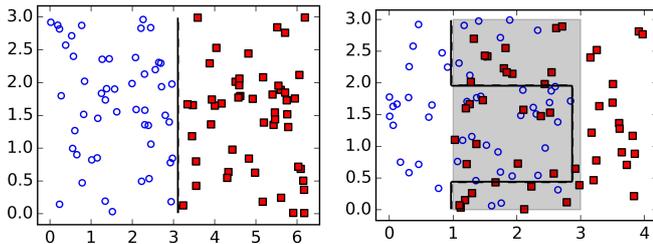


Fig. 1 Non-overlapping and overlapping areas, together with decision boundaries of a DTREE

In order to introduce the formal definitions for R-value and the Augmented R-value, we introduce the following notations:

- n - the number of classes
- C_i - the set of instances belonging to class i
- U - the set of all instances, $U = C_1 \cup C_2 \cup \dots \cup C_n$
- $P_{i,m}$ - the m -th instance of class i

- $\lambda(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases}$
- $kNN(P, S)$ - the subset of k nearest neighbors of instance P that belong to the set of instances S
- θ - threshold value on the number of different class neighbors for considering an instance as belonging to an overlap region

The R-value of a class i is defined in Eq. 1 and can be interpreted as the portion of instances belonging to class i which fulfill the condition in 1.

$$R(C_i) = \frac{1}{|C_i|} \sum_{m=1}^{|C_i|} \lambda(|kNN(P_{i,m}, U - C_i)| - \theta) \quad (1)$$

The R-value of a dataset f is defined in Eq. 2 and captures the portion of instances belonging to all classes, which fulfill the condition presented in 1.

$$R(f) = \frac{1}{|U|} \sum_{i=1}^n \sum_{m=1}^{|C_i|} \lambda(|kNN(P_{i,m}, U - C_i)| - \theta) \quad (2)$$

It follows that, for estimating the level of overlap using the R-value, two main parameters have to be set: k , the number of the nearest neighbors to consider and θ , defined above. The authors of [26] recommend to set the value for θ within the range $[0, k/2]$. According to [26], in all our further experiments, we employ $k = 7$ and $\theta = 3$, i.e. an instance be considered to belong to an overlap region if at least 4 out of its 7 nearest neighbors belong to another class.

We have calculated the R-value for several imbalanced binary classification problems from the KEEL repository [2]. Table 1 presents the values obtained for four datasets. We have also recorded the AUC value of a decision tree classifier and the imbalance ratio of the dataset. One can observe that $1 - R(f)$ is almost constant for these imbalanced datasets, whereas the performance drops as the IR increases.

Table 1 AUC, Imbalance Ratio, 1-R for 4 imbalanced datasets

	AUC	IR	1-R
ecoli1	0.860	3.363	0.913
ecoli3	0.727	8.6	0.931
glass4	0.792	15.461	0.953
yeast4	0.595	28.098	0.964

We assert that the imbalance should also be considered when estimating the degree of overlap. Thus, if we consider that the positive class is the minority class and $U = C_{neg} \cup C_{pos}$ and $C_{neg} \cap C_{pos} = \emptyset$, the R-value of a dataset becomes:

$$R(f) = \frac{1}{|C_{neg}| + |C_{pos}|} \cdot \left(\sum_{m=1}^{|C_{pos}|} \lambda(|kNN(P_{pos,m}, C_{neg})| - \theta) + \sum_{m=1}^{|C_{neg}|} \lambda(|kNN(P_{neg,m}, C_{pos})| - \theta) \right) \quad (3)$$

which is equivalent to:

$$R(f) = \frac{1}{|C_{neg}| + |C_{pos}|} (|C_{neg}| R(C_{neg}) + |C_{pos}| R(C_{pos})) \quad (4)$$

Provided that $|C_{pos}| \neq 0$, we can simplify the equation by $|C_{pos}|$ and use the definition of the imbalance ratio, $IR = \frac{|C_{neg}|}{|C_{pos}|}$:

$$R(f) = \frac{1}{IR + 1} (IR \cdot R(C_{neg}) + R(C_{pos})) \quad (5)$$

As the imbalance increases, the R-value of the majority class possesses an increasingly larger weight than the R-value of the minority class (Equation 5). Thus, for large imbalance ratios, the R-value of a dataset remains almost constant, since the few minority cases contribute very little to its value:

$$\lim_{IR \rightarrow \infty} R(f) = R(C_{neg}) \quad (6)$$

This phenomenon is captured in the results presented in Table 1: as IR increases, the R-value changes very little, whereas the performance drops significantly. Intuitively, for binary classification, the contribution of the majority class overlap to the overall overlap *should not be directly proportional* to the number of negative instances, since the majority of its instances are not located in the overlap region with a high probability. An analogous reasoning can be applied to the contribution of the minority class overlap. Consequently, we introduce the Augmented R-value of a dataset, by weighting the R-value of a class by $|U - C_i|$ instead of $|C_i|$:

$$R_{aug}(f) = \frac{1}{|C_{neg}| + |C_{pos}|} \cdot (|C_{pos}| R(C_{neg}) + |C_{neg}| R(C_{pos})) \quad (7)$$

which simplified by $|C_{pos}|$ results in:

$$R_{aug}(f) = \frac{1}{IR + 1} (R(C_{neg}) + IR \cdot R(C_{pos})) \quad (8)$$

For $IR = 1$, the Augmented R-value is equal to the R-value of a dataset. For large imbalance datasets, the Augmented R-value gets close to the R value of the positive class:

$$\lim_{IR \rightarrow \infty} R_{aug}(f) = R(C_{pos}) \quad (9)$$

In Figure 2 several artificially generated datasets have been plotted together with their R-value, Augmented R-value and Imbalance Ratio. The datasets on the first row try to capture the behavior for changing values of IR, whereas the datasets on the second row have the same IR and varying levels of overlap.

In the generation process of the datasets, only two numerical features were considered: X (horizontal axis) and Y (vertical axis). For generating the datasets on the first row, X and Y were randomly drawn from a uniform distribution of $[0, 10)$ and the imbalance ratio was gradually reduced from 10 to 1. For the datasets on the second row, the imbalance ratio was kept constant at 6. Let $\mathcal{N}(\mu, \sigma^2)$ be the normal distribution with mean μ and scale σ . The feature values of the majority class were drawn from $\mathcal{N}(5, 4)$. The X values of the minority class were drawn from $\mathcal{N}(\mu, 1)$, with μ gradually increasing, while the values of Y were drawn from $\mathcal{N}(5, 1)$.

If we look at the scatter plots in Figure 2 we can observe that the R-value exhibits little variation to the different generation scenarios, whereas the Augmented R-value changes appropriately, according to the “intuitive” degree of overlap. It can also be observed that for higher imbalance values the Augmented R-value places a larger weight on false negative errors, but also takes into consideration the false positive rate.

Since it is based on kNN, the Augmented R-value of a dataset is directly proportional to the portion of the minority instances that would be incorrectly classified by the kNN classifier. Thus, it is expected that $1 - R_{aug}(f)$ has a strong positive correlation with the performance of the kNN classifier.

4 Proposed Meta-learning Approach

Considering that there is no best-suited preprocessing strategy or best imbalance-specific classifier, which achieves good performance on any imbalanced problem, we believe that extracting a relevant set of meta-features and employing them within a meta-learning framework could provide a more valuable solution to issues arising from the imbalance and other data related factors. Consequently, this section presents the meta-features we propose, which focus specifically on capturing the imbalance, the overlap and the complexity of the problem and we briefly describe the overall

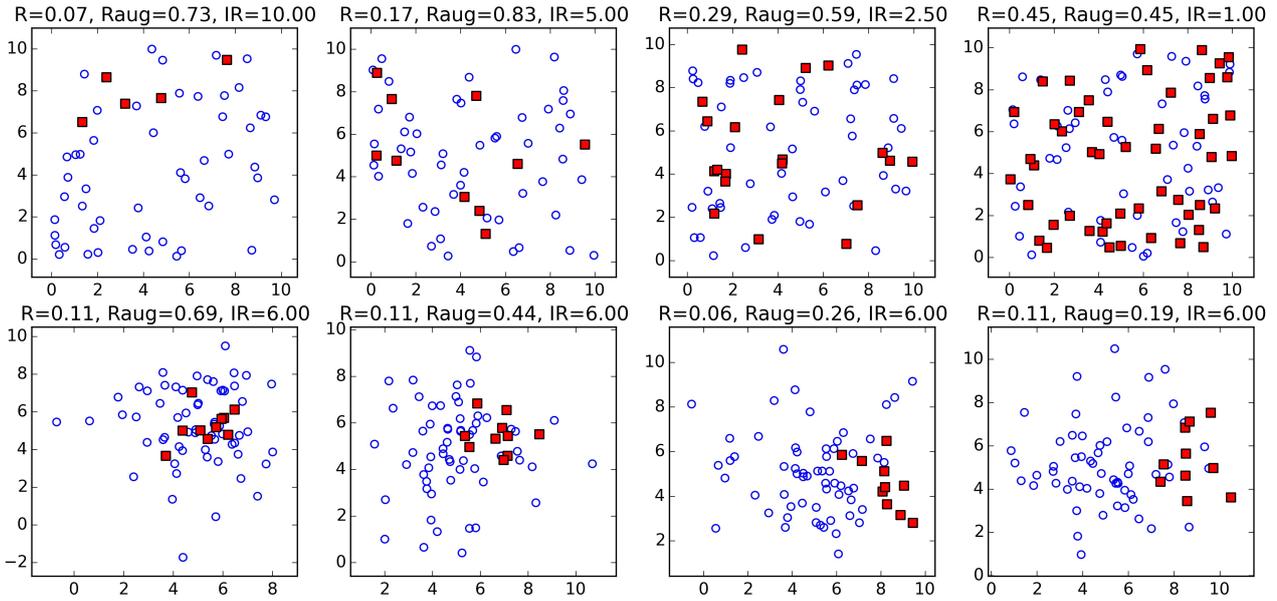


Fig. 2 Plots of datasets with R, R_{aug} and IR

meta-learning approach. Last, we present the feature selection strategy we employed in the next section to validate our feature set empirically. However, we begin with a brief overview of the selected base classifiers, since several meta-features are based them.

4.1 Base classifiers

1. **Support Vector Machine** [10] with a polynomial kernel of degree 1, hereafter referred to as SVM1.
2. **Support Vector Machine** with a polynomial kernel of degree 3, hereafter referred to as SVM3.
3. **C4.5 Decision tree** [29] pruned, with a confidence level of 0.25, hereafter referred to as DTREE.

The choice of SVM1 and DTREE can be motivated by the fact that they represent two of the most utilized traditional classification methods, their behavior being extensively studied in imbalance scenarios as well. The SVM3 classifier earned its place among the selected classifiers, because it can identify more complex decision surfaces than SVM1.

4.2 Meta-features

This section describes the pool of proposed meta-features. They can be divided in three classes, each focusing on a specific problem characteristic:

- **imbalance**: Imbalance Ratio

- **overlap**: Augmented R-value, Fisher’s Maximum Discriminant Ratio
- **complexity**: Instances per Attributes Ratio, Number of Support Vectors generated by SVM1, Number of Support Vectors generated by SVM3, Number of Leaves of DTREE

The *Imbalance Ratio* (IR) is defined as the ratio between the number of instances of the majority class and the number of instances of the minority class. If the positive class is the minority class, then:

$$IR = \frac{|C_{neg}|}{|C_{pos}|} \quad (10)$$

Even if previous studies suggest that there is little correlation between the imbalance ratio and the expected classifier performance [24, 25], we have included it in the meta-features pool, as the representative metric for quantifying the level of imbalance in a dataset.

Fisher’s Discriminant Ratio [25] for feature i is defined as:

$$f_i = \frac{(\mu_{i,1} - \mu_{i,2})^2}{\sigma_{i,1}^2 + \sigma_{i,2}^2} \quad (11)$$

where $\mu_{i,1}$, $\mu_{i,2}$, $\sigma_{i,1}$, $\sigma_{i,2}$ are the means and variances of feature i belonging to class 1 and 2, respectively.

The authors in [25] justify that it is enough to consider Fisher’s Maximum Discriminant Ratio:

$$F1 = \max(f_1, f_2, \dots, f_k) \quad (12)$$

where k is the number of features, since in multiple dimensions one discriminating feature is enough to increase the separability of classes.

The *Instances per Attributes Ratio* (IAR) is defined as the ratio between the number of instances (N) and the number of features ($|A|$) [20]:

$$IAR = \frac{N}{|A|} \quad (13)$$

It tries to provide a simple, straightforward measure for the complexity of a dataset. The evaluations performed in [20] indicate that classifier performance improves at larger IAR values.

The *number of support vectors* is another estimate we consider for the complexity of the dataset. Intuitively, if their number is high, i.e. we have many data points close to the decision surface, the classes are hard to separate. We compute this feature for both SVM1 and SVM3.

A third measure for the complexity of a dataset is the *number of leaves* of a decision tree model. This meta-feature is frequently employed in literature, as a model-based meta-feature [30]. In [18] the authors estimate dataset complexity as $\log L$, where L is the number of leaves generated by an unpruned decision tree. In order to keep the magnitude of this feature comparable to the number of support vectors and since the decision tree algorithm we used performs pruning (as default setting), we used the actual number of leaves as meta-feature.

4.3 Meta-learning strategy

The objective of the meta-classification strategy is twofold:

- to demonstrate that it is possible to select best classifiers based on meta-features for datasets and the average performance competes with the usual strategy of classifier selection based on cross-validation
- to prove the efficacy of the Augmented R-value as a meta-feature

Arguably, the goal of a meta-learning strategy is generally to indicate the most appropriate classifier for a new problem without having to run computationally intensive cross-validation experiments. Although this objective is realistic, our primary goal is to increase classification performance on any given problem which exhibits both imbalance and overlap. We employ several meta-features which imply

training the classifiers on the datasets, such as the number of support vectors and number of nodes in the decision tree, and the two metrics for estimating overlap - the R-value and the Augmented R-value.

The different phases of the process are presented in Figure 3. In the meta-training set generation phase, the collection of available training datasets is evaluated. For each dataset, the meta-features and the classifier achieving the highest AUC score among the classifiers presented in Section 4.1 is retained. The collected meta-features together with the best classifier as the nominal class label form the training set for the meta-classifier. The meta-classifier is a **Logistic Regression Classifier** model. The motivation behind selecting this classifier is that it is simple and robust against overfitting.

During the meta-model building phase, we introduced a feature selection step. We employed Classifier Subset Evaluation [17], which evaluates the feature subset on the training data and utilizes Logistic Regression to estimate the merit of a feature subset in conjunction with Linear Forward Selection search method [16]. This process is repeated, varying the threshold for the maximal number of selected features between 2 and 7. During each iteration, the merit of a feature subset is estimated as the average AUC value obtained via 10-fold cross validation. The threshold value achieving the highest AUC is then used for training the final classifier. The feature selection process is illustrated in listing 1.

Listing 1 Feature Selection

```

classifier = AttributeSelectedClassifier(
    classifier=LogisticRegression(),
    evaluator=ClassifierSubsetEval(),
    searcher=LinearForwardSearch())

for i in 2..7:
    classifier.searcher.max_features = i
    score[i] = crossValidate(classifier,
                            data, folds=10).AUC

classifier.searcher.max_features = argmax(score)
classifier.fit(data)

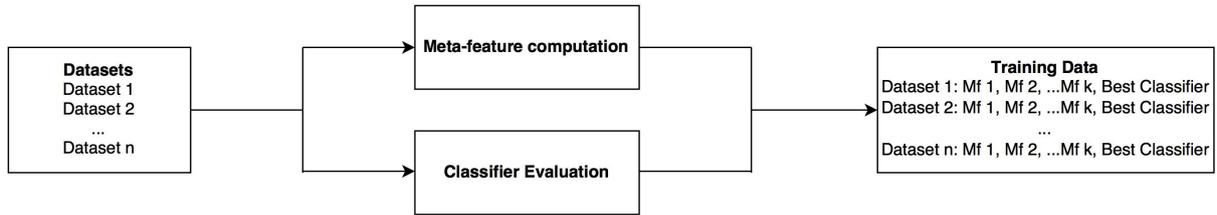
```

Even though this attribute selection method is slow in general, we employ a maximum of 7 meta-features; the meta-classification strategy achieves good performance even with a small number of meta-features. Also, it can be reasonably assumed that the number of meta-instances is small. Therefore, the running time of the meta-classifier is dominated by training the Logistic Regression on small dataset multiple times. The training time of Logistic Regression is actually independent of the sample sizes of the original datasets.

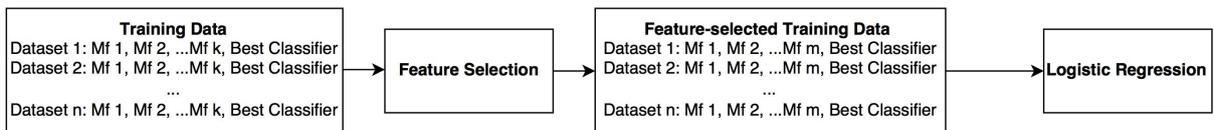
5 Experimental Evaluation

The experimental evaluation considers two different objectives. The first one is to study comparatively the R-value

Meta-training Set Generation



Building the Meta-model



Classifier Prediction



Fig. 3 The phases of Meta-classifier

and the Augmented R-value metrics, on both benchmark and artificially generated data. We take this opportunity to study also the influence of IR on the performance of classifiers. The second objective is to evaluate the performance of the meta-learning strategy for recommending the most appropriate classifier, given the proposed collection of meta-features. This evaluation is conducted on benchmark data. We also perform an analysis of the importance of each meta-feature as resulted from the feature selection step applied in the meta-learning strategy. The results indicate the importance of the Augmented R-value in the performance of the meta-learning strategy.

Throughout our evaluation, we use the Area Under the ROC Curve [5] for measuring classifier performance. Besides the motivation presented in Section 5 for the appropriateness of AUC for imbalanced classification, by measuring classifier performance with AUC we obtain results that are comparable to the findings presented in [24], where the same metric is used. The benchmark data consists of a relatively well known collection of 66 datasets for imbalanced classification, obtained from the KEEL repository. This collection

has been more recently employed in [24], and we wish to be able to compare our results with the results presented there. Each dataset is prepared for 5-fold cross-validation - 5 train and test pairs - maintaining the original class distribution. The datasets are presented in Table 2.

5.1 Evaluation of the Augmented R-value

The first experiment was conducted to evaluate comparatively the R_{aug} , R and IR metrics in a controlled manner, using artificially generated datasets with various levels of imbalance and overlap. The datasets represent binary problems having two numeric attributes each (X and Y). In the Y dimension, the values for both classes are sampled uniformly from $[0, 1)$. In the X dimension, the values for negative class instances are sampled from $[0, 0.5)$, whereas the values for the positive class are sampled from $[0.5 - x, 1 - x)$, both uniformly. This generation process allows us to define an absolute overlap measure, which is exactly $200x$, since x controls the overlap percentage of the two bands along the X direction.

Table 2 Datasets from the KEEL repository

Name	#Inst.	#Attrs	IR	Name	#Inst.	#Atts.	IR
Glass1	214	9	1.82	Glass04vs5	92	9	9.22
Ecoli0vs1	220	7	1.86	Ecoli0346vs5	205	7	9.25
Wisconsin	683	9	1.86	Ecoli0347vs56	257	7	9.28
Pima	768	8	1.9	Yeast05679vs4	528	8	9.35
Iris0	150	4	2	Ecoli067vs5	220	6	10
Glass0	214	9	2.06	Vowel0	988	13	10.1
Yeast1	1484	8	2.46	Glass016vs2	192	9	10.29
Vehicle1	846	18	2.52	Glass2	214	9	10.39
Vehicle2	846	18	2.52	Ecoli0147vs2356	336	7	10.59
Vehicle3	846	18	2.52	Led7digit02456789vs1	443	7	10.97
Haberman	306	3	2.68	Glass06vs5	108	9	11
Glass0123vs456	214	9	3.19	Ecoli01vs5	240	6	11
Vehicle0	846	18	3.23	Glass0146vs2	205	9	11.06
Ecoli1	336	7	3.36	Ecoli0147vs56	332	6	12.28
New-thyroid2	215	5	4.92	Cleveland0vs4	177	13	12.62
New-thyroid1	215	5	5.14	Ecoli0146vs5	280	6	13
Ecoli2	336	7	5.46	Ecoli4	336	7	13.84
Segment0	2308	19	6.01	Yeast1vs7	459	8	13.87
Glass6	214	9	6.38	Shuttle0vs4	1829	9	13.87
Yeast3	1484	8	8.11	Glass4	214	9	15.47
Ecoli3	336	7	8.19	Page-blocks13vs2	472	10	15.85
Page-blocks0	5472	10	8.77	Abalone9vs18	731	8	16.68
Ecoli034vs5	200	7	9	Glass016vs5	184	9	19.44
Yeast2vs4	514	8	9.08	Shuttle2vs4	129	9	20.5
Ecoli067vs35	222	7	9.09	Yeast1458vs7	693	8	22.1
Ecoli0234vs5	202	7	9.1	Glass5	214	9	22.81
Glass015vs2	172	9	9.12	Yeast2vs8	482	8	23.1
Yeast0359vs78	506	8	9.12	Yeast4	1484	8	28.41
Yeast02579vs368	1004	8	9.14	Yeast1289vs7	947	8	30.56
Yeast0256vs3789	1004	8	9.14	Yeast5	1484	8	32.78
Ecoli046vs5	203	6	9.15	Ecoli0137vs26	281	7	39.15
Ecoli01vs235	244	7	9.17	Yeast6	1484	8	39.15
Ecoli0267vs35	224	7	9.18	Abalone19	4174	8	128.87

The set of artificial datasets was obtained by varying both the overlap and imbalance. The overlap percentage generator function is $f_O(n) = 2n$ for $n = 0, 1, \dots, 49$, whereas the imbalance ratio generator function is $f_{IR}(n) = n^{3/2}$ for $n = 1, 2, \dots, 20$. During the generation process the number of instances was kept constant. Thus 1000 datasets were obtained, three of them being illustrated in Figure 4. As evaluation metric we have employed the Pearson correlation coefficient between the values of each of these metrics and the performance of the classifiers measured with AUC.

The results of this first experiment can be found in Table 3. We included the absolute overlap metric also as a column, to conclude that R_{aug} has a stronger correlation with the absolute overlap than R, but even so, the correlation moderate. This actually expected, since both R and R_{aug} are model based metrics, not data based. However, we expected that the R_{aug} possesses strong correlation with performance classifier, better than R, which is confirmed by the results. Even more, R_{aug} is better correlated with the performance of the SVM classifiers than the absolute overlap value; for the DTREE classifier the situation is the opposite. The motivation for this behavior could be found in the data generation process, which varies the overlap level, while keeping other factors which might affect classifier learning constant (e.g. data distribution).

Table 3 The correlations between overlap, IR, R measures and classifier AUC on synthetic data

	Abs. Overlap	SVM1	SVM3	DTREE
Abs. Overlap		-0.490	-0.752	-0.870
IR		-0.489	-0.321	-0.166
R	0.27	0.149	0.019	-0.138
R_{aug}	0.462	-0.903	-0.782	-0.527

We have also performed an analysis on the efficacy of the Augmented R-value in comparison with R-value and the other meta-features on the 66 KEEL benchmark datasets. We employed the same evaluation strategy as before. The results are presented in Table 4. R_{aug} possesses the highest correlation out of the three meta-features for all three classifiers considered. Its absolute value indicates a moderate to strong negative correlation with the performance of classifiers. The correlation of IR with classifier performance can be labeled as weak negative, confirming that the imbalance is not necessarily the only factor affecting performance. We believe the results validate that R_{aug} is more appropriate as overlap metric in imbalanced scenarios than R.

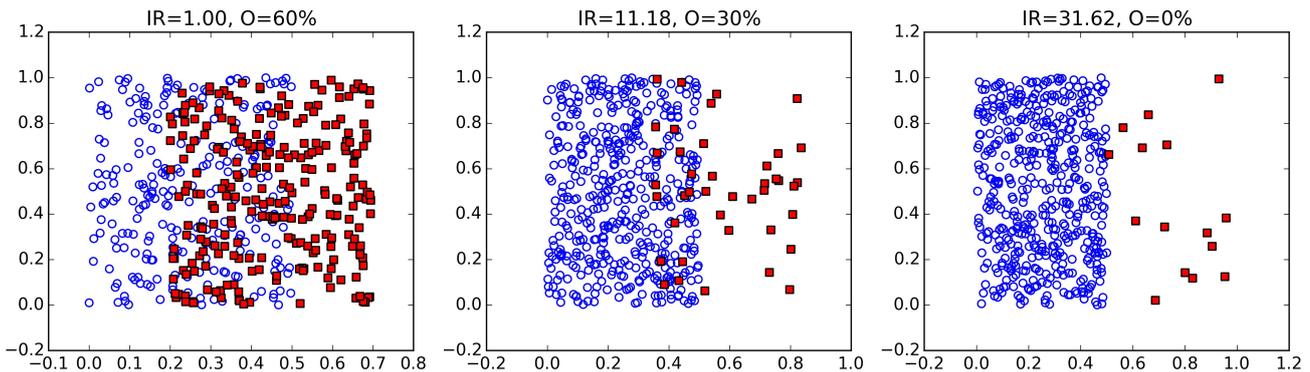


Fig. 4 Plots of synthetic datasets with various degree of overlap and imbalance

Table 4 The correlations between metrics and AUCs on benchmark data

	SVM1	SVM3	DTREE
IR	-0.259	-0.338	-0.3133
R	-0.360	-0.330	-0.289
R_{aug}	-0.663	-0.697	-0.540

Table 5 Weighted AUC, Precision and Recall for the 3-class meta-classification problem

	AUC	Precision	Recall
Train	0.850	0.721	0.719
11-fold CV	0.765	0.634	0.636

5.2 Evaluation of the Meta-classification Strategy

The second set of experiments focuses on assessing the performance of the meta-classification strategy and provides an analysis of the efficacy of the proposed meta-features, highlighting the importance of the Augmented R-value. The evaluation of the meta-classifier was performed in two-steps. First, we assessed the performance of the meta-classifier in predicting the best base classifier (i.e. SVM1, SVM3 or DTREE). We included SMOTE in these evaluations as well, to investigate whether the meta-learning strategy is affected by the application of preprocessing methods. Therefore we generated balanced distribution versions for all the datasets. Thus, we ended up with a collection of 132 datasets, each having 5 train-test cross-validation pairs. Each dataset was then represented by the meta-features as an instance in the meta-classifier’s training set having the best performing classifier as class label.

We performed a 11-fold cross validation on this set and we also reported the performance on the training set, to check for potential overfitting behavior. Table 5 presents the results obtained for this evaluation. The results indicate there seems to be no overfitting, and the meta-classifier achieves a 0.765 average AUC value on the three-class problem of predicting the best classifier.

However, in practical situations, the goal is to have a good performance on a given dataset, and reduce the risk of selecting an inappropriate classifier for the new problem. Therefore, in the second step, we evaluated the actual average performance achieved by the base classifiers predicted by the meta-classifier and reported the average of their AUC score. To perform a fair evaluation at this step, we defined 11 dataset folds. When predicting for a dataset in fold f_{test} , the meta-classifier was trained on training data formed by all meta-instances of the datasets belonging to other folds than f_{test} . We have compared this performance to the average performance of the base classifiers, the baseline performance and the maximum achievable performance. The maximum achievable performance is the average of the AUC scores of the best classifiers for each dataset, which is equivalent to the predictions of a perfect meta-classifier. The baseline performance is obtained by selecting the classifier which achieves the highest average AUC score in the 5-fold cross validation process performed on the training sets.

The performance of the meta-classifier should be better than the one achieved by the baseline recommender. This would validate the idea that making an informed decision on which classifier to use for a new dataset, by inspecting the dataset characteristics, is better than selecting the best average performer indicated by a cross validation process performed on the dataset.

Upon collecting the predictions for each dataset, we evaluated the classifiers indicated by the predictions and measured their average AUC. We have done this both on the "raw" datasets and their modified versions obtained by ap-

plying the SMOTE oversampling method. Table 6 presents these results. The average AUC values achieved by SVM1 and DTREE are comparable with the values reported in [24]. These results indicate that the meta-classification method is indeed successful at predicting the most appropriate classifier for a specific problem; it achieves superior performance to all the individual classifiers it considers and to the baseline meta-classifier - both on non-processed datasets and when SMOTE is applied.

We have also analyzed the suitability of the meta-features considered, by looking at the subsets generated in the feature selection step. For each feature, we computed a score based on the number of times it was selected as being part of the best resulting subset during the experiments. The maximum achievable score is 132, i.e. 11 runs, 6 test datasets in each run, with 2 versions of dataset - without preprocessing and oversampled with SMOTE. The Augmented R-value is the only feature to reach the maximal value. The second most selected feature was F1 (score=124) and the third was the number of support vectors for SVM3 (score=122). IR achieved the lowest score value (44). This confirms yet again the superiority of the Augmented R-value over R-value in capturing the overlap level in imbalanced scenarios, and the fact that the imbalance is actually a poor indicator of expected classifier performance.

Table 6 Average AUC values obtained via the different classification and meta-classification schemes

	SVM1	SVM3	C4.5	Max	Baseline	Meta
No Preproc.	0.689	0.759	0.788	0.820	0.790	0.796
SMOTE	0.850	0.866	0.835	0.878	0.854	0.868

We repeat the latter experiment with three base meta-feature sets. The first feature set consists of all the features defined in 4.2, except for the Augmented R-value; we denote this set by S_1 . In the second set we add the original R-value to the base set, and in the third, the Augmented R-value, respectively. Thus $S_2 = S_1 \cup \{Rval\}$ and $S_3 = S_1 \cup \{AugRval\}$. The results are shown in Table 7; they indicate that the addition of the Augmented R-value to the base feature set produces a performance boost, significantly higher than that produced by the R-value. These results in conjunction with Table 6 also show that the Augmented R-value produces significant performance improvements to the meta-learning strategy.

Table 7 Average AUC values for meta-classification using different meta-feature sets

	Meta S_1	Meta S_2	Meta S_3
No Preproc.	0.791	0.758	0.796
SMOTE	0.843	0.865	0.868

Since the meta-classifier uses logistic regression, it is possible to rank the classifiers based on posterior class distributions. The ideal ranking consists of ranking the classifiers based on their best AUC scores, as defined in the previous section. The Normalized Discounted Cumulative Gain (NDCG) metric is the ratio between the ranking generated by a classifier (DCG) to the ideal ranking (IDCG):

$$DCG = \sum_{i=1}^k \frac{2^{rel_i} - 1}{\log_2(i + 1)} \quad (14)$$

$$NDCG = \frac{DCG}{IDCG} \quad (15)$$

where rel_i is the relevance measure value for classifier at position i . In our case $k = 3$, since we recommend three classifiers and the relevance measure is the AUC score. If we calculate this measure for each dataset and take the average, we get an empirical estimate of the ranking performance of our proposed method. Table 8 presents the NDCG values achieved by the previously defined feature sets S_1 , S_2 and S_3 . The results indicate once again that the Augmented R-value contributes to achieving an almost perfect ranking.

Table 8 NDCG values for the meta-classifier using different base feature sets

	Meta S_1	Meta S_2	Meta S_3
No Preproc.	0.990	0.971	0.991
SMOTE	0.989	0.996	0.996

6 Current state in imbalanced classification

This section reviews the most relevant aspects related to the class imbalance problem - classifier evaluation, strategies for alleviating the imbalance and other data characteristics which affect classifier performance in conjunction with the imbalance. Also, the last sub-section briefly presents the main idea behind meta-learning.

6.1 Evaluating performance

Establishing how to assess performance of classifiers is a sensitive task in imbalanced problems. The selection of

an inappropriate evaluation measure may lead to unexpected predictions, which are not in agreement with the actual problem goals. Consider, for example a classifier constructed on a training set consisting of a positive example and 99 negative examples. If it classifies all examples as negative, it will have an accuracy of 99% on that set, however, such a model is actually useless. Even if they do not explicitly consider accuracy as optimization criterion, most classifiers employ loss functions which generalize relatively well to accuracy.

Moreover, data imbalance problems usually come with an associated requirement: the recognition of minority cases is more important than that of majority cases. For example, in cancer diagnostic problems, positive cases are relatively less common than negative cases. As a consequence, in the available data, the number of patients diagnosed with cancer is smaller than negative diagnosis cases. However, failing to identify a positive case is significantly more serious than misdiagnosing a negative as positive (arguably, both errors are serious, but the former possesses more severe implications on human life).

Table 9 depicts the confusion matrix for a two-class problem. The accuracy is defined as:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (16)$$

It is widely acknowledged within the scientific community that it is an improper metric for imbalanced problems, since it considers the total number of correctly classified instances and it is less sensitive to the recognition errors of the minority class [24?].

Table 9 Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	True Positives (TP)	False Negative (FN)
Actual Negative	False Positives (FP)	True Negatives (TN)

A widely accepted metric for imbalanced scenarios is the Area Under the ROC Curve [5], which captures the trade-off between the True Positive Rate and the False Positive Rate (17) into one single measure. ROC curves are generated by varying the score threshold for the classifiers' prediction probability and obtaining pairs of (FP_{rate}, TP_{rate}) points. An ideal classifier would have $FP_{rate} = 0, TP_{rate} = 1$ thus $AUC = 1$, whereas $AUC = 0.5$ is the expected value of a random classifier. Since it is insensitive to the ratio between positive and negative instances, the AUC is not affected by the class skew, and therefore it is an objective performance criterion for imbalanced problems.

$$TP_{rate} = \frac{TP}{TP + FN}; FP_{rate} = \frac{FP}{FP + TN} \quad (17)$$

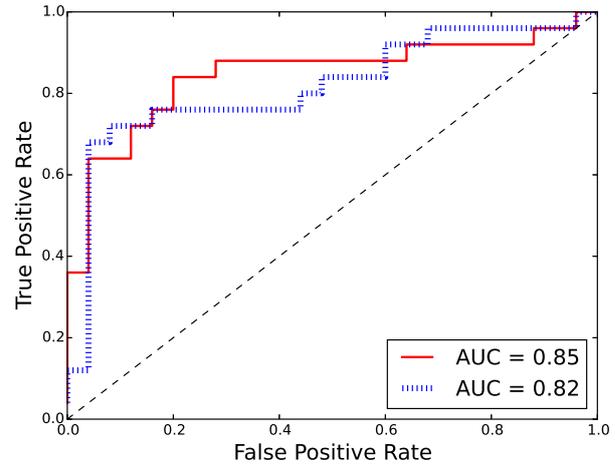


Fig. 5 ROC Curves

The first bisector in Figure 5 represents the ROC curve of the random classifier, whereas the other two dashed and solid curves represent the ROC Curves of SVM classifiers, with polynomial kernel of degree 1 and 2, respectively. The area under the solid ROC curve is larger, which indicates that the latter classifier has a better performance.

The scientific community has also suggested other composite metrics, also derived from the confusion matrix, for evaluating the performance in imbalanced problems: the geometric mean (GM) [4], the balanced accuracy (BAcc) [6], the F-measure [15, 7] and its generalization – the F_{β} -measure – which provides a trade-off between the correct identification of the positive class and the cost of false alarms (in number of false positive errors).

6.2 Factors affecting the performance of classifiers

Another category of factors affecting the performance of classifiers in imbalanced problems encompass a series of data related characteristics co-occurring with the imbalance, or as independent phenomena, but which in conjunction with the data imbalance produce a significantly larger drop in performance than taken individually. Besides the overlapping of the class boundaries, which was discussed in section 2, the authors of [24] identify the following data intrinsic characteristics as being relevant to the expected performance of classifiers in imbalanced scenarios:

- The *lack of density*, or the *small sample size* issue, is related to the insufficient data quantity to allow learning algorithms to generalize separation boundaries correctly. It is known that as the number of features increases, the number of training samples needed to achieve the same performance grows exponentially. When the training data is also imbalanced, classifier overfitting becomes

even more severe [24]. In high dimensional feature spaces feature selection has been shown to alleviate this effect [35].

- The *small disjuncts* problem occurs when sub-concepts are represented in small clusters in the training data, which makes it difficult for a classifier to separate between actual information and noise. This issue may appear as a consequence of the lack of density problem, but it may also arise independently. [36] proposes several strategies to deal with the small disjuncts problem. An important conclusion of the study is related to the fact that there exists a best marginal distribution for learning, which is not necessarily the balanced or the naturally occurring distribution (even though the two achieve reasonable results on average).
- *Dataset shift* refers to the difference between train and test distributions. Classifiers are generally able to handle mild distribution shifts, which is inherent in most real-world applications. In imbalanced classification scenarios, this issue becomes accentuated due to the increased importance of the poorly represented minority instances [24].
- The existence of *noise* in the available training data possesses a stronger effect on learning performance than the imbalance [31]. However, as the imbalance becomes more severe, it plays an increasingly significant role in classifier performance. Naïve Bayes and Support Vector Machines seem to be the most robust to noise, while the performance of the C4.5 classifier degrades more rapidly with increasing noise levels.

6.3 Strategies for alleviating the effect of the imbalance on classifier performance

Several different strategies for improving the behavior of classifiers in imbalanced domains have been reported in the scientific community. Broadly, the approaches for dealing with imbalanced problems can be split into data-centered, algorithm-centered and hybrid solutions.

- **Data-centered techniques** focus on altering the distribution of the training data: either randomly, or by making an informed decision on which instances to eliminate or add (by multiplying existing ones, or artificially generating new cases). Under this category we find random over- and under-sampling, or more elaborated approaches, the prominent approach in this category being Synthetic Minority Over-sampling Technique [9]. SMOTE performs oversampling on the minority class, by randomly generating synthetic new instances on the vectors connecting two original instances lying in the kNN neighborhood of each other. The process of generating synthetic samples is briefly described below:

- let P be an instance from the minority class and \mathbf{P} its feature vector
- let Q be another instance from the minority class, being in the kNN neighborhood of P , and \mathbf{Q} its feature vector
- the new synthetic instance is M described by its feature vector $\mathbf{M} = \mathbf{P} + k(\mathbf{Q} - \mathbf{P})$, where k is a random number, $k \in (0, 1)$

According to the results presented in [24], SMOTE is the *de facto* method to apply in imbalanced scenarios, due to its inherent simplicity and efficiency in reducing the effect of the imbalance.

Sampling methods can be employed as preprocessing techniques. This may come as an advantage, since the computational effort to prepare the data is needed only once. However, most methods require the analyst to set the amount of re-sampling needed, and this is not always easy to establish.

- **Algorithm-centered techniques**, also known as internal approaches, refer to strategies which adapt the inductive bias of classifiers, or specific strategies to adapt the general methodology for tackling the imbalance. Such strategies have been devised for decision trees [28, 39], classification rule learners [14, 22], instance-based learners [23], logistic regression [37] or SVMs [21, 38]. Their main disadvantage is the fact that they are restricted to the specific learning algorithm.

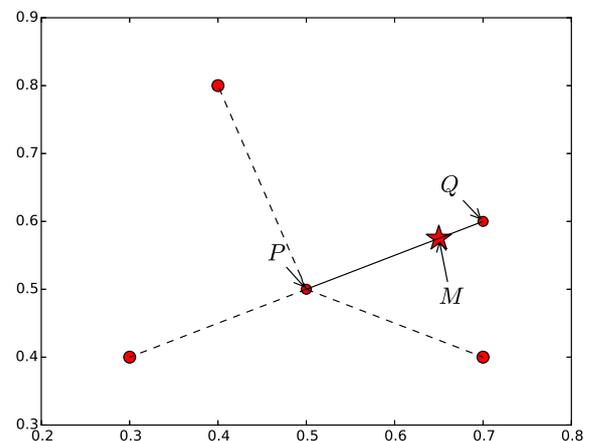


Fig. 6 Synthetic instance generation in SMOTE

- **Hybrid approaches** combine data- and algorithm-centered strategies. A number of approaches in this category consist of ensembles built via boosting, which also employ replication on minority class instances to second the weight update mechanism, in the attempt to focus on the hard examples. Also, the base classifiers may be modified to

tackle imbalanced data. Such approaches include SMOTE-Boost [8], DataBoost-IM [15], and a complex SVM ensemble [33]. Another hybrid strategy which may prove beneficial in imbalanced problems is the one employed in cost-sensitive problems, to bias the learning process according to the different costs of the errors involved [12, 40, 32].

Since classifiers have different biases due to their diverse learning strategies, they are affected differently by the imbalance and the associated characteristics of the training data. Preprocessing strategies have been shown to generally alleviate the performance drop related to the presence of several data-related factors within the context of imbalanced problems, with SMOTE being seemingly the best solution on average. However, to maximize their effect in a specific imbalanced problem, sampling methods need to be paired with the most appropriate learning algorithm – activity which requires time and an experienced analyst. Algorithm-based strategies on the other hand are restricted to a specific algorithm category. If the learning bias of the algorithm does not match the problem characteristics, it will not provide the best solution for the specific imbalanced problem. Hybrid techniques are more general, but they come with additional complexity issues (e.g. time, model interpretability, setting the cost matrix).

Thus, when faced with a new imbalanced problem, with different specific characteristics (e.g. overlap level, complexity or sample size) one cannot establish which learning strategy will prove to be the most robust. Consequently, provided that the analyst possesses an appropriate set of meta-features which can capture the different data-related aspects, a meta-learning approach could solve the difficult task of achieving a good performance level for any imbalanced problem.

6.4 Meta-learning

Automatic selection of a suitable classifier for a given problem has been investigated for some time now, early approaches focusing on deriving interpretable selection rules [1, 3]:

```
if std <= 9970.047 then
  best := NNClassifier;
```

[25] presents a thorough analysis of data complexity measures and their effect on classification, providing also a set of empirically determined rules for classifier behavior. Instead of generating empirical rules, the best classifier for a problem can also be determined by relying on a set of data characteristics, or meta-features. [30] presents such a method and proposes various types of meta-features, including simple, statistical, information-theoretic, model-based and landmarking ones. A collection of problems is established, the

meta-features are calculated and the classifiers are evaluated on each of these datasets. Feature selection is performed to reduce the number of meta-features. In the prediction step, the meta-features are computed for the new problem, and based on their values the meta-classifier recommends the most suitable base classifier for the new problem. The authors of [27] propose an instance based meta-learning strategy for generating ranked classifier predictions. They rely on data based meta-characteristics and explore several alternatives for distance computation, neighbor selection and prediction combination.

7 Conclusions

This paper presented a meta-learning based approach for dealing with complex scenarios involving imbalance and overlap. We proposed a new overlap metric, the *Augmented R-value*, by extending an existing measure, R-value. We provide a theoretical proof as well as qualitative and quantitative evaluations to demonstrate the superiority of the new metric over the initial R-value. Also, confirming previous results, we found that the influence of imbalance alone on the performance of classifiers is limited. However, the level of overlap influences the performance of all classifiers considered and the newly proposed Augmented R-value measure presents a stronger correlation with the performance of classifiers than the original R-value.

Another contribution of the current work is that it proposes a collection of model based meta-features which capture several data characteristics and to provide a meta-learning strategy for predicting the most suitable classifier for a given dataset. The approach was evaluated on a well known collection of benchmark datasets for imbalanced problems, yielding superior results to all the base classifiers considered and to the baseline performance, which reflects the "manual" classifier selection process normally performed by data scientists. The analysis performed on the results of the feature selection process considered in the meta-training flow suggests that overlap measures are the best indicators for expected classifier performance, followed by complexity measures, while the imbalance is the weakest predictor meta-feature.

References

1. Aha DW (1992) Generalizing from case studies: A case study. In: In Proceedings of the Ninth International Conference on Machine Learning, Morgan Kaufmann, pp 1–10
2. Alcalá-Fdez J, Fernández A, Luengo J, Derrac J, García S (2011) KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Multiple-Valued Logic and Soft Computing* 17(2-3):255–287, URL <http://www.oldcitypublishing.com/>

- MVLSVC/MVLSCabstracts/MVLSVC17.2-3abstracts/MVLSVCv17n2-3p255-287Alcala.html
3. Ali S, Smith KA (2006) On learning algorithm selection for classification. *Appl Soft Comput* 6(2):119–138, DOI 10.1016/j.asoc.2004.12.002, URL <http://dx.doi.org/10.1016/j.asoc.2004.12.002>
 4. Barandela R, Sánchez JS, García V, Rangel E (2003) Strategies for learning in class imbalance problems. *Pattern Recognition* 36(3):849–851
 5. Bradley AP (1997) The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition* 30:1145–1159
 6. Brodersen K, Ong CS, Stephan K, Buhmann J (2010) The balanced accuracy and its posterior distribution. In: *Pattern Recognition (ICPR), 2010 20th International Conference on*, pp 3121–3124, DOI 10.1109/ICPR.2010.764
 7. Chawla N (2005) Data mining for imbalanced datasets: An overview. In: Maimon O, Rokach L (eds) *Data Mining and Knowledge Discovery Handbook*, Springer US, pp 853–867, DOI 10.1007/0-387-25465-X_40, URL http://dx.doi.org/10.1007/0-387-25465-X_40
 8. Chawla N, Lazarevic A, Hall L, Bowyer K (2003) Smoteboost: Improving prediction of the minority class in boosting. In: Lavrač N, Gamberger D, Todorovski L, Blockeel H (eds) *Knowledge Discovery in Databases: PKDD 2003, Lecture Notes in Computer Science*, vol 2838, Springer Berlin Heidelberg, pp 107–119, DOI 10.1007/978-3-540-39804-2_12, URL http://dx.doi.org/10.1007/978-3-540-39804-2_12
 9. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) Smote: Synthetic minority over-sampling technique. *J Artif Int Res* 16(1):321–357, URL <http://dl.acm.org/citation.cfm?id=1622407.1622416>
 10. Cortes C, Vapnik V (1995) Support-vector networks. *Mach Learn* 20(3):273–297, DOI 10.1023/A:1022627411411, URL <http://dx.doi.org/10.1023/A:1022627411411>
 11. Denil M, Trappenberg T (2010) Overlap versus imbalance. In: *Proceedings of the 23rd Canadian Conference on Advances in Artificial Intelligence*, Springer-Verlag, Berlin, Heidelberg, AI'10, pp 220–231, DOI 10.1007/978-3-642-13059-5_22, URL http://dx.doi.org/10.1007/978-3-642-13059-5_22
 12. Domingos P (1999) Metacost: A general method for making classifiers cost-sensitive. In: *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, New York, NY, USA, KDD '99, pp 155–164, DOI 10.1145/312129.312220, URL <http://doi.acm.org/10.1145/312129.312220>
 13. García V, Mollineda R, Sánchez J (2008) On the k-nn performance in a challenging scenario of imbalance and overlapping. *Pattern Analysis and Applications* 11(3-4):269–280, DOI 10.1007/s10044-007-0087-5, URL <http://dx.doi.org/10.1007/s10044-007-0087-5>
 14. Grzymala-Busse J, Stefanowski J, Wilk S (2004) A comparison of two approaches to data mining from imbalanced data. In: Negroita M, Howlett R, Jain L (eds) *Knowledge-Based Intelligent Information and Engineering Systems, Lecture Notes in Computer Science*, vol 3213, Springer Berlin Heidelberg, pp 757–763, DOI 10.1007/978-3-540-30132-5_103, URL http://dx.doi.org/10.1007/978-3-540-30132-5_103
 15. Guo H, Viktor HL (2004) Learning from imbalanced data sets with boosting and data generation: The databoost-im approach. *SIGKDD Explor Newsl* 6(1):30–39, DOI 10.1145/1007730.1007736, URL <http://doi.acm.org/10.1145/1007730.1007736>
 16. Gutlein M, Frank E, Hall M, Karwath A (2009) Large-scale attribute selection using wrappers. In: *Computational Intelligence and Data Mining, 2009. CIDM '09. IEEE Symposium on*, pp 332–339, DOI 10.1109/CIDM.2009.4938668
 17. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009) The weka data mining software: An update. *SIGKDD Explor Newsl* 11(1):10–18, DOI 10.1145/1656274.1656278, URL <http://doi.acm.org/10.1145/1656274.1656278>
 18. Japkowicz N, Stephen S (2002) The class imbalance problem: A systematic study. *Intell Data Anal* 6(5):429–449, URL <http://dl.acm.org/citation.cfm?id=1293951.1293954>
 19. Japkowicz N, Stephen S (2002) The class imbalance problem: A systematic study. *Intell Data Anal* 6(5):429–449, URL <http://dl.acm.org/citation.cfm?id=1293951.1293954>
 20. Lemnar C, Potolea R (2012) Imbalanced classification problems: Systematic study, issues and best practices. In: Zhang R, Zhang J, Zhang Z, Filipe J, Cordeiro J (eds) *Enterprise Information Systems, Lecture Notes in Business Information Processing*, vol 102, Springer Berlin Heidelberg, pp 35–50, DOI 10.1007/978-3-642-29958-2_3, URL http://dx.doi.org/10.1007/978-3-642-29958-2_3
 21. Lin Y, Lee Y, Wahba G (2002) Support vector machines for classification in nonstandard situations. *Machine Learning* 46(1-3):191–202, DOI 10.1023/A:1012406528296, URL <http://dx.doi.org/10.1023/A:1012406528296>
 22. Liu B, Ma Y, Wong C (2000) Improving an association rule based classifier. In: Zighed D, Komorowski J, Żytkow J (eds) *Principles of Data Mining and Knowledge Discovery, Lecture Notes in Computer Science*, vol 1910, Springer Berlin Heidelberg, pp 504–509, DOI 10.1007/3-540-45372-5_58, URL http://dx.doi.org/10.1007/3-540-45372-5_58
 23. Liu W, Chawla S (2011) Class confidence weighted knn algorithms for imbalanced data sets. In: *Proceedings of the 15th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining - Volume Part II*, Springer-Verlag, Berlin, Heidelberg, PAKDD'11, pp 345–356, URL <http://dl.acm.org/citation.cfm?id=2022850.2022879>
 24. López V, Fernández A, García S, Palade V, Herrera F (2013) An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Inf Sci* 250:113–141, DOI 10.1016/j.ins.2013.07.007, URL <http://dx.doi.org/10.1016/j.ins.2013.07.007>
 25. Luengo J, Fernández A, García S, Herrera F (2011) Addressing data complexity for imbalanced data sets: analysis of smote-based oversampling and evolutionary undersampling. *Soft Computing* 15(10):1909–1936, DOI 10.1007/s00500-010-0625-8, URL <http://dx.doi.org/10.1007/s00500-010-0625-8>
 26. Oh S (2011) A new dataset evaluation method based on category overlap. *Comput Biol Med* 41(2):115–122, DOI 10.1016/j.compbiomed.2010.12.006, URL <http://dx.doi.org/10.1016/j.compbiomed.2010.12.006>
 27. Potolea R, Cacoveanu S, Lemnar C (2011) Meta-learning framework for prediction strategy evaluation. In: Filipe J, Cordeiro J (eds) *Enterprise Information Systems, Lecture Notes in Business Information Processing*, vol 73, Springer Berlin Heidelberg, pp 280–295, DOI 10.1007/978-3-642-19802-1_20, URL http://dx.doi.org/10.1007/978-3-642-19802-1_20
 28. Quinlan J (1991) Improved estimates for the accuracy of small disjuncts. *Machine Learning* 6(1):93–98, DOI 10.1007/BF00153762, URL <http://dx.doi.org/10.1007/BF00153762>
 29. Quinlan JR (1993) *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA
 30. Reif M, Shafait F, Goldstein M, Breuel T, Dengel A (2014) Automatic classifier selection for non-experts. *Pattern Analysis and Applications* 17(1):83–96, DOI 10.1007/s10044-012-0280-z, URL <http://dx.doi.org/10.1007/s10044-012-0280-z>
 31. Seiffert C, Khoshgoftaar T, Van Hulse J, Follco A (2007) An empirical study of the classification performance of learners on im-

- balanced and noisy software quality data. In: Information Reuse and Integration, 2007. IRI 2007. IEEE International Conference on, pp 651–658, DOI 10.1109/IRI.2007.4296694
32. Sun Y, Kamel MS, Wong AKC, Wang Y (2007) Cost-sensitive boosting for classification of imbalanced data
 33. Tian J, Gu H, Liu W (2011) Imbalanced classification using support vector machine ensemble. *Neural Computing and Applications* 20(2):203–209, DOI 10.1007/s00521-010-0349-9, URL <http://dx.doi.org/10.1007/s00521-010-0349-9>
 34. Visa S (2005) Issues in mining imbalanced data sets - a review paper. In: in Proceedings of the Sixteen Midwest Artificial Intelligence and Cognitive Science Conference, 2005, pp 67–73
 35. Wasikowski M, wen Chen X (2010) Combating the small sample class imbalance problem using feature selection. *Knowledge and Data Engineering, IEEE Transactions on* 22(10):1388–1400, DOI 10.1109/TKDE.2009.187
 36. Weiss GM (2003) The effect of small disjuncts and class distribution on decision tree learning. PhD thesis, New Brunswick, NJ, USA, aAI3093004
 37. Williams D, Myers V, Silvious M (2009) Mine classification with imbalanced data. *Geoscience and Remote Sensing Letters, IEEE* 6(3):528–532, DOI 10.1109/LGRS.2009.2021964
 38. Wu G, Chang EY (2003) Class-boundary alignment for imbalanced dataset learning. In: In ICML 2003 Workshop on Learning from Imbalanced Data Sets, pp 49–56
 39. Zadrozny B, Elkan C (2001) Learning and making decisions when costs and probabilities are both unknown. In: In Proceedings of the Seventh International Conference on Knowledge Discovery and Data Mining, ACM Press, pp 204–213
 40. Zhou ZH, Liu XY (2006) Training cost-sensitive neural networks with methods addressing the class imbalance problem. *Knowledge and Data Engineering, IEEE Transactions on* 18(1):63–77, DOI 10.1109/TKDE.2006.17